March 3rd, 2023 | University of Twente

On the right track

Developing a multi-objective routing optimization model for high-speed logistics companies

MSc Thesis

Author: V. M. Minghetti, BSc

Supervisor: Dr. ir. E. A. Lalla-Ruiz

Supervisor: Dr. D. R. J. Prak

Supervisor: R. Schoenzetter, MSc

On the right track

Developing a multi-objective routing optimization model for high-speed logistics companies

University of Twente, Department of Industrial Engineering & Business Information Systems

PO Box 217, 7500 AE Enschede, tel. +31(0)534899111

Author: V.M. Minghetti, BSc

Lead Supervisor: E. A. Lalla-Ruiz, Dr. ir.

Second Supervisor: D.R.J. Prak, Dr.

Company Lead Supervisor: R. Schoenzetter, MSc.

This report is written as part of an MSc graduation project in the program Industrial Engineering and Management, performed at a Special Speed Logistics Company (which we refer to as SSLC). Its content is primarily intended for the company, as input material for improving the discussed processes, and for the examination committee.

Schiphol, March 3rd 2023

Management summary

We conduct this research at the Customer Service (CS) department within a Special Speed Logistics Company (SSLC), a freight-forwarding company specialized in urgent deliveries. This department is responsible for handling incoming requests for transportation jobs from its customers. The requests come in daily, mostly by phone or email. Agents of the customer service handle them by first offering a quotation to the customer. This offer contains information about the proposed route to transport the parcel, including an estimated time of pickup and arrival at destination, and a price. If the customer agrees with the quotation, the agents proceed with the booking. Once the booking is complete, logistic partners involved are notified, and thus proceed on executing the transportation job.

Crucial to the entire process is the route calculation. The latter is done by entering information about the shipment's pickup and delivery locations, availability time, nature of transported goods and dimensions into a route calculator, which then returns a set of feasible routes. The current system has several limitations, requiring agents to regularly use their expertise and make manual amendments to the routes. As SSLC is moving towards more standardized and data-driven operations, they wish to improve the route calculation of their express airfreight network. We approach this assignment from a mathematical perspective, and thus formulate the main goal of this research:

Design and validate a mathematical route optimization model that minimizes transit time, expected delay and costs, such that it cumulatively improves the current Express Airfreight routes at SSLC

We start by mapping and analyzing the routing process. By conducting interviews with company stakeholders, we learn that the route calculation is actually performed by an external entity, which also owns the flight data. SSLC's system interacts with this entity sending an API[1] request each time a new route needs to be calculated, upon which the latter returns up to 150 routes. Next, SSLC's system validates those options, filtering out inoperable routes; finally, it displays a selection of the fastest ones to the CS agent. Both routing process and system have limitations:

- <u>Inefficient process setup</u>: since the validation step is only done at the end, an estimated 75% of the generated routes is discarded. This waste is computationally demanding and overall inefficient.
- <u>Selection of airports</u>: the system generating the routes selects the departing and arriving airports solely based on proximity to the pickup and delivery points. This potentially limits the number of flight options, especially when regional airports are selected.
- <u>Comparison of alternatives</u>: the system can only generate routes for the selected origin-destination pair and does not compare other alternatives; this again limits the quality of results.

[1]An Application Programming Interface (API) is a software interface that allows two computer programs to communicate with each other.

- <u>Only time is considered</u>: the routes are solely selected based on the earliest arrival at destination. Crucial factors like the risk of incurring delays and the costs of a route are neglected throughout the calculation

By consulting a panel of company experts we define a framework to measure a route's quality. We hence distinguish three primary objectives, consisting of the minimization of a route's transit time, delay and operating costs. By applying the AHP method we translate the stakeholders' relative preferences of these objectives into weights. We use these weights to cumulatively measure and compare the performance of Express routes from the years 2020-2022.

After consulting literature, we decide to model the problem at hand as a multi-objective Multimodal Route Choice Problem, altering the original formulation by Lei et al. (2014). The problem consists of finding the optimal combination of airports and carriers to minimize collectively its transit time, risk of delay and total costs. To account for the three objectives, we aggregate them linearly using weights. To provide SSLC with alternative solutions, we solve the models with five different weights configurations: three which consider only either of the objectives (hence greedy), one which attributes them equal importance (we call this balanced model) and one which uses the preferences we elicit from the panel of experts (we call this weighted model). To solve the problem, we use the tabu search algorithm. Tabu search is a metaheuristic which solves complex optimization problems by altering an initial solution; the peculiarity of this method is the usage of memory to prevent exploring the same solution spaces recursively (Glover and Taillard 1993). To generate the initial solution, we follow the same steps used by the current system. We then use two tabu search variants: a classic one (TS), which changes the initial solution's route by swapping airports in the sequence, and a hybrid one (MNTS), which additionally removes or adds airports to the existing route. To account for the problem's stochastic nature, we incorporate the tabu search methods into a simheuristic. Simheuristic procedures evaluate solutions solved deterministically by means of a simulation (Juan, et al. 2015).

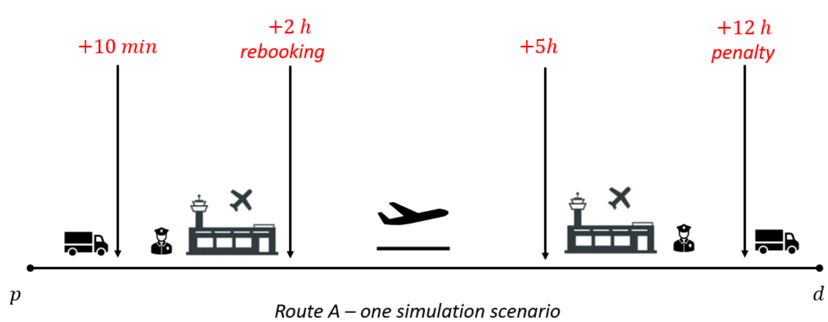


Figure 1: Illustrative example of one simulation scenario on route A

Figure 1 shows an illustrative example on how we apply simulation to a route. Using probability distributions we directly estimate from the data, we start by randomly drawing a time realization for the first-mile leg; we then calculate the time difference with the deterministically calculated arrival time at the origin airport and eventually store the delay. In our example, this results in a delay of 10 minutes. We proceed with randomly drawing time values for each subsequent leg. If, following a delay, we miss our next flight, we store the delay value and add a rebooking fee to the original total costs. Ultimately, if we arrive later (or earlier) than 12 hours from the deterministically calculated delivery time, we add penalty costs. We start at a 50% penalty of the total (deterministic) costs, with an accumulation up to a 100% penalty for 24 hours deviation and beyond. The penalty system is not used in reality by SSLC but is rather a model design choice to enforce timeliness. By averaging the simulation outcomes, we evaluate the solution's robustness.

We test our solution on a single Express Airfreight lane. A lane consists of a group of orders sharing the same origin and destination. Using historical data, we estimate parameters for the model and probability distributions for the simheuristic. We then proceed with solving 49 problem instances which correspond to the lane's historical orders. Hence, for each order, we calculate routes using each combination of model configuration and solution type. Table 1 provides a summary of each solution's average improvement (in green) or deterioration (in red) with respect to the lane's current performance (benchmark) on transit time, delivery delay and costs. We also attach a summary of the routing policy per solution type, with their respective advantages and disadvantages.

Table 1: Solutions' improvements compared to the test lane's current performance (left) and general routing policies for each solution type (right) with corresponding advantages and disadvantages.

Benchmark			
Solution type	Transit time	Delivery delay	Total cost
Current	41.5	3.2	647.9
Time-greedy			
Solution type	Transit time	Delivery delay	Total cost
TS	-44%	107%	120%
MNTS	-45%	102%	143%
Risk-greedy			
Solution type	Transit time	Delivery delay	Total cost
TS	37%	116%	394%
MNTS	66%	63%	264%
Cost-greedy			
Solution type	Transit time	Delivery delay	Total cost
TS	22%	75%	-17%
MNTS	49%	62%	-22%
Balanced			
Solution type	Transit time	Delivery delay	Total cost
TS	-2%	51%	-15%
MNTS	-3%	25%	-23%
Weighted			
Solution type	Transit time	Delivery delay	Total cost
TS	-30%	4%	-5%
MNTS	-29%	-6%	-20%

Solution	Routing policy	Advantages	Disadvantages
Time-greedy	• Use more regional and medium-sized hubs and direct options • Use vehicles to avoid long waiting times for the next flight	• Significantly faster routes	• High risk of delay • Frequent rebookings and penalties • Overall high operating costs
Risk-greedy	• Choose combination of least risky carriers and airports	• Partly reduces carrier- and partner-bound delay	• Worst performance on all KPIs • Practically inoperable routes
Cost-greedy	• Use major hubs with frequent and reliable flights • Use only cost-efficient airlines	• Reduces transit delay as to avoid extra rebooking costs • Best cost reduction	• Slower routes
Balanced	• Use major hubs at departure • Choose destination airport to reduce last-mile leg	• Overall good KPI performance	• Deterioration of transit time at the expense of cheaper routes
Weighted	• Choose minor or medium hub as starting airport • Transit at a well-connected major hub • Combine multiple different airlines on same route	• More emphasis on transit time and reliability	• More expensive than balanced and cost-greedy routes

Time-greedy

The time-greedy solutions improve the transit time KPI and hence yield faster routes. This comes however at the cost of significant delay and costs deteriorations. Overall, these routes are sensitive to delay and thus instable. Most of them are operated through minor airports. The advantage of doing so is that minor airports have generally lower service times as they are less burdened by high cargo volumes; moreover they are closer to most pickup points. However, they have also less (frequent) flight options, meaning that a parcel missing its flight needs to wait longer until the next one and thus incurs significant delays. To mitigate this partly, the solutions choose to use vehicles for transportation. As those are not bound to scheduled departure times, it takes generally less time to drive the shipment than to wait for the next flight. Vehicles are however significantly more expensive than aircraft. This, in combination with the high number of rebookings and penalties incurred, makes those routes more expensive than others.

Risk-greedy

Risk-greedy solutions perform the worst, showing a significant deterioration in almost all KPIs. From a practical point of view, the generated routes do not make sense. The reason for this is that the model only accounts for routes having the least risky airports and carriers, neglecting any other aspect. Whereas we observe that accounting for those risks helps lowering the average delay KPIs, one third source of risk is overlooked in the model design, namely too tightly planned transits. This explains why the risk-greedy routes do not yield the lowest delay KPIs of the proposed solutions.

Cost-greedy

The cost-greedy routes yield overall the best improvement on the cost objective. These routes predilect the usage of major hubs, as those are overall cheaper. Moreover, the hybrid solution variant tends to add transits (at major hubs) to the initial route. Despite being counterintuitive, this move is done to catch more reliable flights: by doing so the solution avoids missing transits, which implicitly avoids rebookings and penalty fees. For the same reason, this solution predilects routes with more waiting time between flights: this partly mitigates the fact we do not consider tight connections as a risk. Both developments deteriorate the transit time, making routes on average 35% slower than the original historical ones.

Balanced

The balanced solutions tendentially improve the routes' transit times and costs and deteriorate the delivery delays. The routes are similar to the cost-greedy ones, as they predilect the use of major hubs. They are, however, faster, as the balanced options show more variation in the destination airports. Where the cost-greedy ones prefer major-to-major routes, the balanced routes tendentially fly on major-to-medium and major-to-minor airports. This ultimately shortens the transit times and slightly increases the operating costs.

Weighted

The weighted solutions are the best performing ones, with the MNTS variant showing the best results. With respect to the others, these solutions show the most variation in routes utilized and consequently carriers as well: we distinguish fourteen different routes and a total of five airlines used. Furthermore, different airline groups are used in combination for a single route, which is not permitted in the current system. Interestingly, a considerable 26% of routes uses a medium or minor airport as origin. In contrast with the time-greedy solutions, direct options are avoided, as those routes operate using one transit at a major airport and again a major airport as destination. This solution type finds a good balance in the speed/reliability tradeoff. By staring at a minor (or medium) hub the travel time of the first-mile leg and the initial service time are reduced. Typically, this hub is better connected to its destination via a major hub than directly, as for both flight legs more regular options are available. This means that if the parcel misses its original first flight it can still recover the delay by taking the next one.

Although we obtain encouraging results, we acknowledge a still notable effect of randomness on the average outcomes. In fact, abnormally high generated time values increase the average results, which differ largely from the simulation's most occurring values. This means that the average time and delay outcomes are overall pessimistic. If we validate the outcomes using real order data, we see that latter measures tend to be overestimated, whereas the costs are underestimated. This is however not surprising, since we explicitly avoid using the real costing data, as this is considered sensitive information.

In conclusion, we provide SSLC with different solution alternatives which overcome most of the identified limitations of the current system. Firstly, our solutions select the origin and destination airports in a smarter way and can compare multiple combinations, thereby returning the best route. Most importantly, we demonstrate that accounting for multiple objectives yields overall better routes, with our best-performing solution improving the current routes' transit time by 29%, delivery delay by 6% and total operating cost by 20%. Additionally, we show how shifting their relative importance influences our solutions' routing behavior. Based on the obtained results, we formulate recommendations for the short and medium-long term.

Short term recommendations

- Usage of minor and medium hubs: using these types of airports can be beneficial to a route's overall speed. We recommend exploring these benefits by trying to use these hubs more regularly.
- Combine different airlines: the current system does not allow to use different airline groups on a single route, whereas our solution demonstrates this can be beneficial. Therefore, we recommend investigating further whether dropping this constraint could be beneficial.
- Relax the departure constraint: similarly, the current system generates only routes with flights departing from the origin within 24 hours from the pickup time. Our solution relaxes this constraint obtaining better results; therefore we recommend dropping it.
- Explore the possibilities of using trucks: the time-greedy solutions use (fictive) trucks for delay recovery, yielding positive outcomes on the delay and transit time objectives. It is questionable whether such options are operable in practice; therefore, we recommend to research this.

Medium-long term recommendations

- Improve data quality: the overestimations in the simheuristic are the consequence of outliers in the used data. These are hard to be filtered out, as the data is not documented sufficiently. Therefore, we recommend researching ways to improve the overall data collection process.
- Acquisition of flight data: by owning the schedule data, SSLC would be able to implement an in-house routing system, with the benefits of a more efficient process and better routes. It is up to the company to examine whether the costs are worth the presented benefits.
- Test the solution on larger network: if SSLC wishes to implement our solution, we recommend first testing it on a larger sample, to verify the computational burden can be sustained as to calculate routes for the entire Express network.

Foreword

I hereby present you with my Master Thesis, which marks the end of my master Industrial Engineering and Management at the University of Twente and thus, my time as a student. As I write this preface, I cannot help thinking of when I started as a freshman back in 2017. I had then taken the tough decision to put one of my biggest dreams, becoming an airline pilot, in stand-by. Times back then were hard for newcomers to the aviation industry and my parents (sagely) advised me to first pursue an academic degree. Ironically, what started as a backup plan for me turned out to be the right decision for my career. Throughout the years, I have developed a strong interest for many subjects taught within the Industrial Engineering and Management program, which ultimately lead me to drop the pilot dream and continue with a master. The passion for aviation, however, has never vanished. SSLC gave me the opportunity to enter the industry and learn that, indeed, I belong there.

I would like to start thanking Eduardo Lalla and Dennis Prak, for their contribution as supervisors from the University of Twente. Besides sharing my enthusiasm for the research topic, they always supported me with clear, constructive, and honest feedback that ultimately helped lift my work to a higher level. But most especially, they always gave me the feeling of great trust in my capabilities and judgement.

I would also like to thank all the wonderful people at SSLC whom I had the pleasure to work with for almost a year. In particular, I want to thank Remy Schoenzetter for giving me the opportunity to join the company and entrusting me with such a unique assignment. I would also like to thank Christoph Lansink, who encouraged me to initiate this research and always believed in its relevance.

I also want to thank my friends, in particular Thom, Martijn, Daniël, Steve and Thijmen, who accompanied me during both my bachelor and master years, and to whom I owe countless hours of procrastination and perhaps even a couple of study points.

Additionally, I would like to thank my girlfriend for her support and patience and last but not least, my parents, to which I owe everything.

As a final word, I would like to share a quote which best puts this research, but also the fast world we live in, into a different perspective:

“Hurry up Big Panda, we’re going to be late!” Tiny Dragon said.

Big Panda sat down. “I’d like to think I’m creating anticipation.”

Big Panda and Tiny Dragon, by James Norbury

I hope you enjoy reading my thesis.

Vittorio Minghetti

Contents

Management summary ... i
Foreword ... vi
Contents ... vii
Tables and figures ... ix
Abbreviations and Terminology ... xi
Chapter 1 - Introduction ... 1
1.1 Background ... 1
1.2 Problem identification ... 3
1.3 Research approach ... 5
Chapter 2 - Problem context ... 11
2.1 Routing system ... 11
2.2 Route performance metrics ... 18
2.3 Current routes performance ... 21
2.4 Conclusion ... 31
Chapter 3 - Literature review ... 32
3.1 Problem identification ... 32
3.2 Solution approach ... 41
3.3 Conclusion ... 47
Chapter 4 – Solution design ... 49
4.1 Problem formulation ... 49
4.2 Mathematical model ... 54
4.3 Tabu search ... 58
4.4 Pilot study setup ... 64
4.5 Simheuristics procedure ... 67
4.6 Conclusion ... 69
Chapter 5 – Solution results ... 70
5.1 Experimental design ... 70
5.2 Experimental results ... 74
5.3 Conclusion ... 90
Chapter 6 – Conclusions and recommendations ... 91

6.1 Conclusions 91
6.2 Recommendations 95
6.3 Limitations and further research 96
References 98
Appendices 103
A. Product portfolio 103
B. Problem cluster 105
C. KPI selection list 108
D. AIJ method calculations 110
E. Auxiliary procedures pseudocode 112
F. Example plots of visual goodness-of-fit tests 114
G. Experimental outcomes of the tuning experiments 115
H. Experimental outcomes of the deterministic analysis 117
I. Experimental outcomes of the simheuristic 122

Tables and figures

Tables

Table 1: Solutions' improvements compared to the test lane's current performance (left) and general routing policies for each solution type (right) with corresponding advantages and disadvantages. iii
Table 2: Abbreviations used in this report xi
Table 3: Terminology used in this report xi
Table 4: Main research stakeholders 3
Table 5: Entry data for the current routing system. 12
Table 6: Fictive door2door route from the Netherlands to Germany. 13
Table 7: Weighted KPIs 21
Table 8: Volume-performance comparison. 28
Table 9: Performance of the on-time orders. 30
Table 10: Comparison current best/worst lanes per KPI. 30
Table 11: MRCP literature summary. 39
Table 12: Description of the sets and indices used in the mathematical model. 54
Table 13: Description of the parameters used in the mathematical model. 55
Table 14: Model variables, distinguished between decision and auxiliary variables. 55
Table 15: Expected delay per km. 67
Table 16: Overview of experiments with explanation of the goal of each experiment 70
Table 17: Objective weights configurations of the five tested models. 71
Table 18: Overview of the tuning experiments, with the respective parameter, the experimental range, the value by which we increment the parameter and the measured metrics. 72
Table 19: Summary of the experimental settings per solution type, found after the tuning experiments. 75
Table 20: Comparison of results for the MIP model and the TS and MNTS metaheuristics. 77
Table 21: Overview of the average performance comparison per solved model type. 77
Table 22: Overview of the solutions' routing policies for solving the problem deterministically. 78
Table 23: Performance comparison between the deterministic and stochastic solutions. 81
Table 24: Comparison of the routing decisions of the deterministic and stochastic solutions and historical routes. 83
Table 25: KPI comparison between the simheuristic solutions and the benchmark. 84
Table 26: For instance 1 (left) and 2 (right), we compare the percentual difference between each KPI's average value and mode. On the bottom part of the table, we report the average gap per KPI type. Instance 2 shows the highest gap and thus the most skewness of results. 88
Table 27: Experimental results of the sensitivity analysis. 89
Table 28: Summary of the main experimental outcomes. 92
Table 29: Summary of the routing policies for each of the five solved model types. 93
Table 30: Solvable core problems (Blue= already being solved, Green = yet to be solved) 106
Table 31: KPI selection on route quality 108
Table 32:Tuning outcomes for the simheuristic parameters $iteri$ and $iterf$. 115
Table 33: Outcomes of the tuning experiments for the Classic Tabu search 1. 116
Table 34: Outcomes of the tuning experiments for the Classic Tabu search 2. 116
Table 35: Outcomes of the tuning experiments for the Hybrid Tabu search. 116
Table 36: Experimental results of deterministically solving the test instances with the time-greedy ($\alpha = 1$) model. 117
Table 37:Experimental results of deterministically solving the test instances with the risk-greedy ($\beta = 1$) model. 118
Table 38: Experimental results of deterministically solving the test instances with the cost-greedy ($\gamma = 1$) model. 119
Table 39: Experimental results of deterministically solving the test instances with the balanced model. 120
Table 40: Experimental results of deterministically solving the test instances with the weighted model. 121
Table 41: Simheuristic results for the time-greedy ($\alpha = 1$) model. 123
Table 42: Simheuristic results for the risk-greedy ($\beta = 1$) model. 124
Table 43: Simheuristic results for the cost-greedy ($\gamma = 1$) model. 125

Table 44: Simheuristic results for the balanced model. 126
Table 45: Simheuristic results for the weighted model. 127

Figures

Figure 1: Illustrative example of one simulation scenario on route A ii
Figure 2: Summary of the assignment, core problem, goal and main deliverable 7
Figure 3: Research Design (Grey = input resources, Blue = research questions, Gold = outputs) 10
Figure 4: System interaction in route calculation. 14
Figure 5: Current (As-Is) versus ideal (To-Be) validation process. 18
Figure 6: AHP judgement scale (Nydick and Hill 1992) 20
Figure 7: Initial datasets 22
Figure 8: Datasets combined 23
Figure 9: Pick-up and delivery points, global overview. 23
Figure 10: Locations sample of pick-up and delivery points, European view. 24
Figure 11: ABC classification of lanes according to cumulative volume. 26
Figure 12: Identification of outliers using a box plot. 27
Figure 13: ABC lane classification from worst to best weighted performance 28
Figure 14: Classification scheme for the Multimodal Route Choice Problem based on network attributes. 38
Figure 15: Schematic representation of time in the proposed model. 51
Figure 16: Pseudocode of the initial solution generation. 60
Figure 17: Pseudocode of the utilized tabu search algorithm. 61
Figure 18: Pseudocode for generating the neighborhood in the classic tabu search. 61
Figure 19: Neighborhood generation for the MNTS. 63
Figure 20: Schematical representation of neighborhood operators used in the hybrid tabu search. 63
Figure 21: Example plot of the probability density function for service times at airport a. 65
Figure 22: Simheuristic solving process. 68
Figure 23: Comparison of the solutions' aggregated scores per model type.. 85
Figure 24: Performance gap between the current situation and the proposed solution. 85
Figure 25: Boxplots of the solutions' stochastic evaluation for instance 1. 86
Figure 26: Boxplots of the solutions' stochastic evaluation for instance 2. 87
Figure 27: Global overview of Express Air hubs 104
Figure 28: Problem cluster. 106
Figure 29: Individual expert judgments 111
Figure 30: Aggregated matrix and consistency check 111
Figure 31: Pseudocode for calculating the ready time at the origin airport o. 112
Figure 32: Pseudocode for calculating the ready time at a transiting airport a. 112
Figure 33: Pseudocode for calculating the ready time at the ending airport e. 112
Figure 34: Pseudocode for finding the optimal carrier combination given a sequence of input airports 113
Figure 35: Pseudocode for selecting the best neighbor solution from a generated neighborhood. 113
Figure 37: Example plot of the service times distributions at airport a. 114
Figure 38: P-P (left) and Q-Q plot for visually inspecting the goodness-of-fit of the Weibull, Gamma exponential and lognormal distributions on service times data. 114

Abbreviations and Terminology

Abbreviations

Table 2: Abbreviations used in this report

Abbreviation	Description
CC&M	Customer Care & Monitoring
CS	Customer Service
ETA	Expected Time of Arrival
GOP	Global Offer Project
LAT	Latest Acceptance Time
MA	Major hub
ME	Medium hub
MI	Minor hub
NFO	Next Flying Option
NS&P	Network Solutions & Procurement
EA	Express Airfreight
STA	Scheduled Time of Arrival
STD	Scheduled Time of Departure
SSLC	Special Speed Logistics Company

Terminology

Table 3: Terminology used in this report

Term	Explanation
Airport2airport	Shipment from origin to destination airports
Compartment five	Cargo compartment for smaller sized packages; located at the tail of an aircraft
Consignee	The entity receiving the shipment on behalf of the end customer
Customer	The entity requesting an order
Destination	Ending airport
Door2door	Shipment from shipper's to consignee's address
Lane	Unique combination of origin and destination airport, which groups all routes sharing that same combination.
Origin	Starting airport
Route	Unique combination of pick-up and delivery points, visited airports and logistic partners involved
Shipper	The entity shipping the goods on behalf of the end customer
Tail-to-tail	Direct transit of a shipment from one aircraft to the other

Chapter 1 - Introduction

This chapter introduces the company and the research addressed within this master thesis. Section 1.1 provides background to SSLC, introducing the company in general and their motivation to initiate this research. Section 1.2 details the addressed core problem and motivates its relevance. It introduces the stakeholders involved throughout the research and discusses the examined problem and the implications of its choice. Finally, Section 1.3 demarcates the research boundaries. First, it describes the research scope and main goal; next, it states the main research question and auxiliary sub questions that we answer as to solve the core problem. The chapter concludes with an outline of the remainder of this report, specifying the order and method by which we address the research questions.

1.1 Background

1.1.1 The company

Company history

SSLC is an international, all-round freight forwarder specialized in urgent transport and complex logistics. Spare parts, medical samples, important documentation, and other types of urgently needed goods are transported upon daily customer requests, some of them incoming just hours before the next available option. Performing a transportation job often includes diverse logistic operations, like picking up goods at the shipper's location, performing customs clearance and ensuring cargo security, and delivering to the consignee. Besides being part of a major European airline, SSLC has six offices worldwide, employs over 330 employees and owns a vast logistic network, allowing them to provide their customers with highly tailored, express solutions. SSLC Netherlands, at which we conduct this research, is part of the Western Europe office, with home base in Amsterdam.

Operations

SSLC Netherlands offers a wide range of products to their customers (more detail in Appendix A). Requests come in daily at the Customer Service (CS) department, where we conduct this research. This department is responsible for handling them: this entails tasks spanning from processing a transport job to ensuring the shipment has been delivered and the invoice is complete. The processes executed within the department can be broken down to the following:

- Quoting: this process starts with an incoming request for a transportation job. Typically, the request is received via email or phone; information provided includes the type of shipment to be transported, its pickup and delivery addresses and the time by which the transportation needs to be arranged (which in most of the cases, is as soon as possible). A CS agent enters the data into their main operative system to search for viable options and offers an Estimated Time of Arrival (ETA), routing (i.e., transit stations and time slots at which each intermediary transport job is executed) and price to the customer.
- Booking: if a customer accepts the offer (i.e., sends a confirmation, either by email or phone), then the agent books it in the system and informs partners involved. If the partners do not accept

the job, the agent manually looks for alternatives and rebooks the order. Once the booking is complete the shipment gets tracking status.

- Tracking and monitoring: agents follow shipments throughout their transit and ensure the status is regularly updated. This is done with a monitoring system, where all shipments in transit are shown together with their status. Shipments with green status do not require action, when these turn red an irregularity in the schedule is taking place. This might be a delay, flight cancellation, loss of package, or anything that could prevent the shipment from reaching its destination on time. When this happens, agents assess the situation and eventually intervene to solve the irregularity. This may involve coordinating the involved logistics partners, and eventually cancel and rebook the shipment's route.
- Billing: when tracking is concluded, agents check the shipment's tracking info for delays. It might be the case that customers cause delays during transport by providing SSLC with incorrect information (like wrong contact details or addresses). In this case, additional charges are added to the invoice and the order archived.

Within the department, the operations team oversees the first two processes, while the Customer Care & Monitoring (CC&M) oversees the latter two. Besides that, customers might request to either change or cancel bookings; in that case, depending on how far the shipment status is, either of the two teams takes care of it. Finally, dedicated teams that handle either contracted business or highly challenging transport requests (Specials and Industry desks) perform all important processes for their customer segments.

1.1.2 Research motivation

This section sheds light on recent developments within the company and how these led to initiating this research. SSLC originates as a customer-tailored forward freighter. Key selling points are high speed, reliability and dedicated logistic experts for the individual customer. In its twenty years of existence, SSLC has grown into having a large network that allows them to offer much more standardized solutions and requires less effort and knowledge to arrange transports. While the company's size and market share have been growing at high pace, operations remain rather manual and often too customer centric[2].

Treacy and Wiersma (1997) distinguish three competitive strategies that define companies:

- Operational excellence: this is focused on cost leadership, where firms manage to align their resources as to optimize their cost/offering ratio.
- Customer intimacy: tailor-made solutions for customers; resources at hand are dedicated to serve customer needs.

[2] Customers that do not require challenging transportation jobs (e.g., between European hubs) can easily search and book options via the SSLC online portal. Most, however, still prefer to have a dedicated agent to take care of the booking for them, since they are used to this type of service.

- Product leadership: the product offered is either superior or unique in the company's market.

Following this definition, the company's upper management argues SSLC is currently transitioning from customer intimacy towards more product leadership, as the assets withing their network allow them to offer an increasingly unique value proposition. Therefore, one of the prime challenges the company faces is to modernize their current operations, introducing standardization and automation for high-volume, low complexity customer requests, while keeping their trademark dedication and high service levels.

1.2 Problem identification

1.2.1 Stakeholders

This section presents the entities or persons that are closely involved within this research and briefly describes their contribution to it. Table 4 shows an overview of the individual stakeholders.

Table 4: Main research stakeholders

Stakeholder	Description
Dr. ir. E. A. Lalla-Ruiz	Lead supervisor from the University of Twente
Dr. D. R. J. Prak	Second supervisor from the University of Twente
MSc. R. Schoenzetter	Company supervisor, Head of Operations at SSLC Netherlands
CS supervisors	Supervisors of the Customer Service team at SSLC Netherlands
CS team	Agents within the Customer Service team at SSLC Netherlands
NS&P team	Members of the Network Solutions & Procurement team at SSLC Netherlands
GOP team	Team members of the Global Offer Tool project at SSLC Global
IT team	Team members of the IT department at SSLC Global

The University of Twente supervisors are primarily concerned with the research's academic importance. Therefore, their main contribution is to verify the research provides with sufficient innovation within the fields of Logistics Management and Operations Research. The lead company supervisor on the other hand, wishes to see practical results that are of use for SSLC Netherlands. His role is to validate the practicability of the proposed solution, thus ensuring the research's impact for companies operating in high-speed logistics. Throughout the research's lifetime, we analyze several business processes involving different company's departments. In particular, members of the CS, NS&P, GOP and IT departments play an important role in understanding how these processes currently work and how they feel towards the improvements proposed within this research.

1.2.2 Core problem

This section presents the problem addressed by our research and discusses its relevance. In order to identify the right problem to tackle in a managerial setting, Heerkens and van Winden (2017) propose the Managerial Problem-Solving Method. We choose to follow this method, as it helps us distinguishing the various types of problems present in the current operational landscape, understand their relationships, and prioritize them. Appendix B discusses these steps in more detail.

Current routing process and limitations

We choose to focus on improving the route calculation process, which is initiated any time a CS agent looks for possible options given a new transportation request by a customer. To handle each request, the CS agent enters information about the shipment in SSLC'S operative system, which on its turn returns a set of possible routes. A route describes the sequence of visited transit points between origin and destination, transit times between them and the responsible partner for each transit job. The calculation needs to be done for all products listed in Appendix A, except for the Spare Parts & Service Logistics and SSLC Warehouse products. Routing is part of the quotation process (booking involves deploying assets needed to operate the selected route) and is currently done in either three systems: systems A, B or C. If a specific product is required, a corresponding system is used to query options. Most customer requests, however, do not explicitly require a product. For example, a shipment that needs to be transported from Paris to Munich and is within the 300kg weight can be transported via the Express Train product, tailored service (direct truck or private flight), or flight express options (air freight). In this case, system A is used most of the times. Depending on the selected product, the system calculates the route with a different type of logic.

Given the latter, we make a distinction between tailored, train and air freight routes. The former two are rather trivial: there is a limited set of partners to select, tracks are either non-stop or via a fixed set of stations and there is minimal variability in terms of transportation schedules. On the other hand, air freight implies a lot more variables to deal with. Besides the bigger number of combinations of possible partners and flight connections, airport handling and customs operations come into play. Choosing the wrong partner or airport can have significant consequences for the success of a transportation job. For instance, when customs are involved, most agents will prefer a route using a bigger hub rather than a regional airport, as bigger hubs are usually better equipped and have more flexible office hours. Similarly, bigger hubs generally have more outbound flight options, which helps mitigating potential travel delays. In fact, if a shipment misses its flight, it may still arrive on time at destination if there are good alternative options. A drawback of major hubs, however, is that they are burdened by higher cargo volumes, and may not be always near the appointed pickup and/or delivery points. As a consequence, choosing to depart/arrive at a smaller airport might be beneficial in reducing the overall transportation time. Besides that, speed of delivery is not the only relevant metric that determines a route's quality. Overall, SSLC strives to provide the fastest transportation options available to their customers, but also seeks to offer a reliable service at a competitive price. Therefore, choosing the right set of airports, airlines and handling partners also affects other relevant aspects like the risk of incurring delays, and the costs of operating a route.

Currently, these considerations are neglected by the routing system. In fact, it is capable of solely comparing flight options for one couple of departing and arriving airports at the time, and does not consider any aspect other than which combination of flights, given known flight schedules, yields the earliest arrival time at the final delivery point. Furthermore, the origin and destination airports are currently selected based on geographical proximity to the shipment's pickup and delivery point, which can significantly limit the number and quality of available transportation options. Because of the latter, agents mostly force the system to find options between airports they manually input as origin and destination. On their turn, the agents base their selection on experience. This leads them to systematically overlook potentially better options and makes the process more error prone in general. Ultimately, poor routes are more likely to cause irregularities (e.g. delays, missing parcels, miscommunication errors, etc.) which add workload to the CC&M department.

Solution objectives

Given its upstream position within the quotation business process, the route calculation significantly affects the subsequent processes and is therefore of considerable importance to SSLC's operations. By engaging this topic, we seek to achieve multiple objectives:

- Improvement of the customer service
  Firstly, customers should benefit from improved routes for their shipments. Given the urgency of the shipments, the primary objective would be reducing transit time, followed by costs as second important factor.
- Reduction of irregularities
  Besides enhancing speed and reducing costs, choosing better routes would also mean avoiding disruptions throughout transportation. These include delays, lost packages and rebookings.
- Reduction of agents' workload
  Finally, a smarter way of routing should also free agents from unnecessary work. Ideally, agents should spend less time to come up with routes during the quoting process and to solve irregularities during the tracking & monitoring process.

Besides its practical relevance, the problem also poses an academically interesting challenge. Indeed, the complexity of the logistic network at hand combined with the fact that multiple objectives must be considered when calculating the optimal route, make this a rather unique and complex problem type within the field of Operations Research. By studying it we therefore also contribute to brining novel insights into this field of study.

1.3 Research approach

To improve the route calculation process at SSLC, we approach the topic with an analytical mindset. First, we analyze the current situation both qualitatively as quantitatively, as to get a complete benchmark. From there, we explore different types of candidate solutions within the field of mathematical modelling and optimization, select one and adapt it to our context. Finally, we setup experiments to test the solution's effectiveness, upon which we draw our conclusions. In the coming section we break down those steps in detail.

1.3.1 Research scope and goal

We perform this research at the Customer Service department of SSLC Netherlands. Therefore, the problem setting is limited to the air transportation routes that are booked by its agents. From this set, we make a selection of routes, which serves as sample to design and implement a solution. This is a necessary step, as taking all combinations of airports served yearly by SSLC would result in a mathematical problem of disproportionate size. Consequently, practical recommendations regarding, for example, which transit airports or carriers should be operated are limited solely to those selected routes.

In general, the mathematical problem and solution approach serve as starting point for SSLC to implement a new routing system. A first step in this process would mean extending the model to all lanes at SSLC Netherlands, and then eventually to lanes used by other offices. It should be noted however, that the model scale-up is not the focus of the research and therefore out of scope. Additionally, the technical implementation of the model into the existing information systems at SSLC falls also outside the research scope. Overall, we aim to provide the company with a product- and system-agnostic solution. This design choice seeks to ensure the presented results are generalizable and flexible enough to be shaped for future developments, given SSLC'S complex and dynamic nature.

Another scope limitation regards the resources considered in the model formulation. The solution approach is designed to provide optimal routes given the current resources at hand (partners, available flights, etc.). Besides providing routes, the model has also the potential of exposing eventual bottlenecks within SSLC'S logistic network. We could well observe, for example, a systematic avoidance of certain airports or airlines, implying the need to search for alternatives for the provision of better options. Although we acknowledge this would be a very interesting extension of the model, we argue this is not directly related to the core problem we define in Section 1.2.2 . Consequently, we leave this outside of the scope of this research.

In terms of goals, the first one is to provide with results that are potentially generalizable for other forward freighters operating in high-speed logistics. Although we answer the research questions presented in Section 1.3.2 within the context provided by the Customer Service department at SSLC Netherlands, we seek to design a mathematical model that can be adjusted to other similar contexts, thereby increasing the relevance of this research. Furthermore, this project distinguishes two other types of expected contributions. The practical contribution is to provide SSLC Netherlands with concrete suggestions to improve their current routes. Besides a practical goal, this research also seeks to pursue an academic goal, this being to contribute to the existing body of knowledge in the domain of routing optimization and mathematical programming. Having outlined the research boundaries, Figure 2 summarizes its general assignment, core problem, goal and main deliverable.

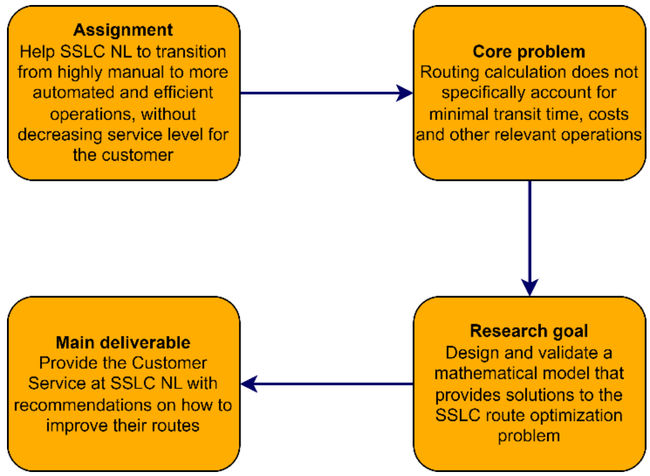


Figure 2: Summary of the assignment, core problem, goal and main deliverable

1.3.2 Research questions

In order to solve the core problem and to achieve the set goals, we formulate and answer the main research question:

How can mathematical programming support route calculation at SSLC as to improve the performance of their routes?

To address it properly, several supportive questions need to be answered first. This section presents them in sequence.

Analysis of the current situation

The first step we take is to examine how the airfreight routes are currently calculated by the system and how this affects their overall performance. To understand the former, we conduct interviews with company stakeholders from the CS, NS&P and IT teams. Here we acquire information on the process' current design, its requirements and supposed limitations. Hence, we answer the question:

1.1 How does the current system in use calculate a route?

a) Which input data is required?
b) How is this data processed?
c) What makes a route feasible?
d) What are the limitations to the current approach?

The next step is to measure the performance of the current routes. To do so, we need to determine which KPIs should be used for the measurement and their relative importance. On one hand, using knowledge by the company stakeholders involved closely with the routing process helps us in understanding why certain KPIs are more important than others and, in general, prevents us from potentially misjudging the company's perspective on routes' quality. On the other hand, literature complements their perspectives, providing us with novel insights that can be potentially become relevant to our problem setting. Therefore, by interviewing stakeholders from the CS, NS&P, GOP and IT departments and consulting literature, we answer the following question:

1.2 How can we measure routing performance?

a) What metrics does SSLC use to measure routing performance?
b) What additional metrics can be found in literature?
c) Which metrics are relevant to this context?
d) How can we assign weights to the selected performance metrics?

Finally, we seek to analyze the current performance of the air freight routes. We use the found metrics to conduct a quantitative analysis of SSLC's historical orders, which answers the question:

1.3 How do the air freight routes at SSLC Netherlands perform?

a) What are the most used routes at SSLC Netherlands and how are they performing?
b) What are the best performing routes at SSLC Netherlands and why?
c) What are the worst performing routes at SSLC Netherlands and why?

Literature review

Once the current situation is analyzed, we need to explore which routing problems and corresponding solution approaches potentially fit to our context, and ultimately compare them to find the one best-tailored to our needs. By conducting a literature review and analyzing its outcomes, we thus provide a solid foundation for the to-be solution approach. In sequence, we answer the following two questions:

2.1 Which routing optimization problem fits best to the context of SSLC?

a) What are the problem specifics? (Objective, constraints, sets, variables, parameters).
b) How does the problem setting differ to the one of SSLC?
c) What changes must be made in order to adapt the found model to the problem context?

2.2 What is proposed in literature to solve the SSLC routing problem?

a) What methods are proposed?
b) How do these methods differ?

Design of the solution approach

The third phase is concerned with the design of a solution approach. We design a model and select optimization techniques able to solve it in reasonable time, while yielding the best results possible. To accomplish the latter, we answer the question:

3.1 How should the solution approach be designed?

a) What is the scope of the solution?
b) What are the requirements of the solution?
c) What are the assumptions of the solution?

Experimentation and evaluation

Once we designed the solution approach, we proceed on evaluating its performance. The first step we take is to prepare the experimental setup, where we define the experiments to execute and their intended purpose. Next, we carry on executing them and analyzing the outcomes.

4.1 How does the proposed solution perform compared to the current routing algorithm?

a) What experimental setup should be used?
b) How do the generated routes differ from the routes analyzed in 1.3?

Conclusions and recommendations

Based on the observed results, we can make inferences about how the proposed solution performs compared to the current situation. Based on that, we make recommendations to the company with suggestions for eventual future work:

5.1 What are the conclusions and recommendations for SSLC?

a) What can be concluded on the proposed solution in terms of routing performance, compared to the status quo?
b) What recommendations and suggestions for future research can be done as to enable SSLC to improve their most used routes?

1.3.3 Research design and methodology

Based on the research questions, we divide our research in five phases:

- **Problem context analysis:** Qualitative and quantitative analysis of the current situation, addressed in Chapter 2.
- **Solution generation and selection:** Exploration, comparison and selection of known solution approaches in literature, addressed in Chapter 3.
- **Solution design:** Adaptation of the selected solution method to the studied context, addressed in Chapter 4.
- **Experimentation and results analysis:** Analysis of the solution's effectiveness by means of experiments, addressed in Chapter 5.
- **Evaluation:** Evaluation of the research's finding and the implications for SSLC; discussed in Chapter 6.

Altogether, these phases form the research design. Figure 3 gives a schematic representation of the research design and methods to address the research questions throughout those phases.

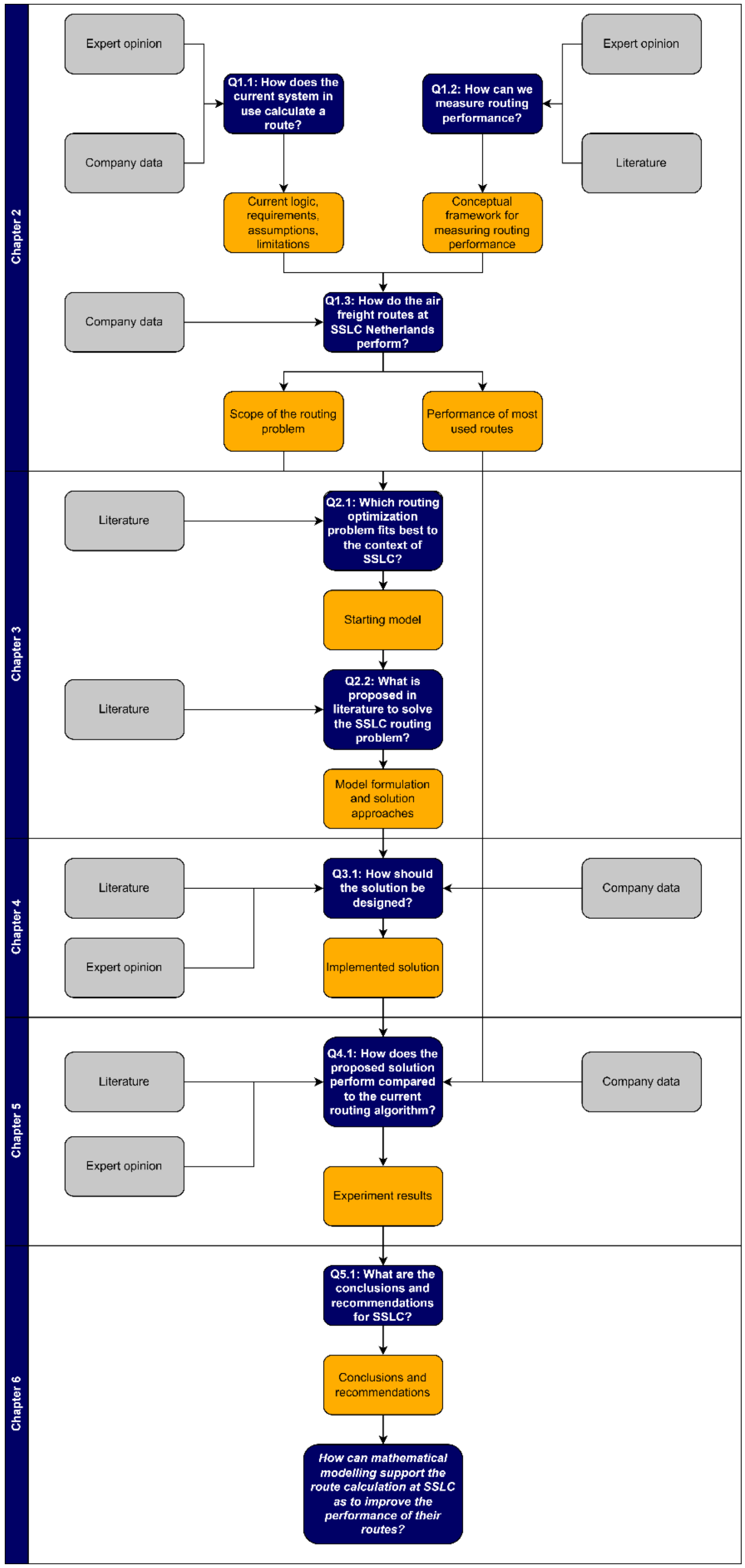


Figure 3: Research Design (Grey = input resources, Blue = research questions, Gold = outputs)

Chapter 2 - Problem context

This chapter addresses the routing status quo at SSLC. Section 2.1 answers the question: “How does the current system in use calculate a route?”. To do so, we examine which data is required by the system and how these data are processed. As explained further in Section 2.1.1 , SSLC’s routing system processes the input filled in by an agent and sends an API request to an external flight data provider. This party queries a list of possible flight combinations and returns it to SSLC; ultimately, their system validates flight options given a set of operational requirements. Section 2.1.2 addresses this validation step in more detail and thereby clarifies when a route is deemed feasible. After the validation, agents select either of the filtered results for the transportation job. This approach has limitations: we address them in Section 2.1.3 .

Section 2.2 presents the metrics we use to evaluate the quality of the actual flight routes, and thereby answers the question: “How can we measure routing performance?”. Section 2.2.1 introduces the main objectives of interest when it comes to route selection and how these can be measured by means of Key Performance Indicators (KPIs). Section 2.2.2 shows how we assigned weights to the selected KPIs and thereby their relative importance.

Section 2.3 concludes the context analysis by showing how air freight routes perform according to the framework defined in Section 2.2 . We thus answer the research question: “How do the air freight routes at SSLC Netherlands perform?”. First, Section 2.3.1 presents which data are used for the analysis and roughly how it is (pre-) processed; Section 2.3.2 proceeds by displaying the main analysis results. Section 2.3.3 looks further at select group of routes, their performance and reflects on the implications of those findings.

2.1 Routing system

2.1.1 Route selection process

Input and output

Section 1.2.2 globally describes the process of offering a transportation route upon incoming customer requests. Here we provide a helicopter view of the steps in sequence and interactions between systems that ultimately generate flight options, which form a route that can be selected by an agent. Please note that technical details regarding the algorithms in use, like detailed algorithm steps, are omitted due to data sensitivity.

The first step in the process involves an agent entering information from a customer request in their operating system. Table 5 lists the data required by the system to calculate a route.

Table 5: Entry data for the current routing system. This is the information SSLC booking agents input to the system to generate a route.

Data	Type	Description
Pick-up point	Location	Starting point of the shipment, can be either a postal code or an airport code.
Delivery point	Location	Ending point of the shipment, can be either a postal code or an airport code.
Transport date	Date	Date at which the shipment leaves the pickup point.
Available time	Time	Time at which the shipment is ready for pickup.
Package dimensions	Long	Package length, width, height, weight and number.
Good description	Category/text	Type of product being shipped.
Origin (optional)	Location	Origin airport code.
Overnight (optional)	Location	Mandatory transit airport code.
Destination (optional)	Location	Destination airport code.
Lithium batteries (optional)	Category	Only relevant when shipment contains either type of lithium battery.

As shown in Table 5, entering origin and destination airports is not required. In fact, when the fields are empty, the system uses the pickup and delivery coordinates to find their nearest airports and flight connections between them. Whenever agents manually amend them, the system automatically calculates the distance and travel time from/to the indicated airports. Agents also have the option to enter an overnight station: in this case, the system checks combinations of flight options that transit at this airport. This can be done for several reasons: for example, one of them would be to reduce risk of missing a transit flight. To do so, an agent might choose to manually enter a major hub as transit point, since those hubs generally provide more frequent flight options than regular airports. Finally, if the shipment contains certain types of lithium batteries, the number of flight options are limited, since transportation can be done only by cargo aircraft, following SSLC's airline policy. For more detail on this, we refer to the documentation provided by the IATA (2021).

When all required data is entered, the system provides at most seven, already validated, flight options. Each route has a number of legs, depending on how many transits are included. Table 6 shows a possible route for a fictive door2door shipment from the Netherlands to Germany. In this case, we have a direct flight plus the first- and last- leg miles, hence three legs. Each leg has a set of times: the Latest Acceptance Time (LAT) is the latest by which the shipment needs to be at the origin location, ready for transport. The Scheduled Time of Departure (STD) is the time at which the transport between leg origin and destination starts; the Scheduled Time of Arrival (STA) is the time at which the shipment is scheduled to arrive at the leg destination point. There is a distinction between the latter and the Time Of Arrival (TOA). Whereas the STA marks the scheduled time of arrival of the shipment at a location, the TOA is an indication of the scheduled time at which the shipment is ready for the next leg. For example, an STA may indicate the time of arrival of a flight at the airport, while the TOA indicates the time by which the shipment is offloaded and ready for collection. This distinction is made to account for handling time at stations. Furthermore, each leg contains information regarding the operating partner and the travelled distance. The partner selection per leg is also an output from the routing system, as choosing the right courier, handler or airline has a significant impact on the shipment's transit times. Finally, both the costs per leg as the total costs for the route are displayed.

Table 6: Fictive door2door route from the Netherlands to Germany. The route consists of transportation legs, each having a start and end point, planned cut-off and transportation times, the partner responsible for the leg and the travelled distance.

Leg	Start	End	LAT	STD	STA	TOA	Partner	Distance (km)	Cost
1	Pickup	AMS	13:45	13:45	15:00	15:00	Courier X	47	X €
2	AMS	FRA	16:20	18:35	19:40	20:40	Airline Y	-	Y €
3	FRA	Delivery	20:40	20:40	22:00	22:00	Courier Z	17	Z €

The routes are sorted by earliest TOA at delivery point. This ordering does therefore not consider risk (for example, a route through major hubs or with less transits generally implies less delay risk) or costs. There are, however, some warnings the routing system displays to the agent. For example, if either transportation leg is outside office hours of the involved stations, the system warns the agent to check this since there might be complications. Also, capacity warnings are displayed (if there is no capacity guarantee onboard, the agent needs to double-check this). If equipment is needed for loading/offloading of heavy shipments, this is indicated as well. Finally, flights that historical data show to be systematically late are marked. Ultimately, the agent still has the possibility to manually amend some fields; this includes choosing a different handling partner if available and changing the pickup times of driving legs. Depending on the manual changes, the system automatically adjusts dependent time and cost estimates.

System interaction overview

As anticipated at the start of the chapter, an external party is involved in the process of providing SSLC agents with potential routes. Specifically, this entity queries flight schedules based on the parameters sent by SSLC, combines the schedules, and returns them. Figure 4 gives an overview of the steps. The process starts with the location of an origin and destination airport, given either the pickup and delivery locations or manual agent specifications. Next, a courier partner is selected for the pickup and delivery legs. This is done by choosing the closest partner to either pickup/origin airport and delivery/destination airport. Based on the transportation date and availability time of the shipment, a pickup leg time is calculated, hence the TOA of the shipment at the origin airport is computed. By adding 30 minutes to this time (which is a fixed buffer SSLC wields), the LAT for the first flying leg is obtained. With this, all information is in place to send an API request to the flight scheduling entity. Section 2.1.2 explains in more details the contents of such request. The entity then responds by sending up to 150 flight options to SSLC's operative system. This is a predefined number, as SSLC observed that returning more options would further increase the system's response time without providing significantly better options. If an insufficient number of options is received, SSLC sends another request with less tight requirements as to broaden the flight search (see Section 2.1.2). If three requests do not yield enough options, the system returns an error to the agent. This might be a suggestion to insert a different combination of origin/destination airports or different time slots as to generate flight options. When however, the flight scheduler provides sufficient results, SSLC's system uses operational data to validate them (i.e., filter out infeasible options). After the validation, up to seven fastest flight options are displayed containing information about the transportation legs and the costs. The agent finally decides whether to re-query for options with different settings (e.g., airports, dates and times) or to select either option.

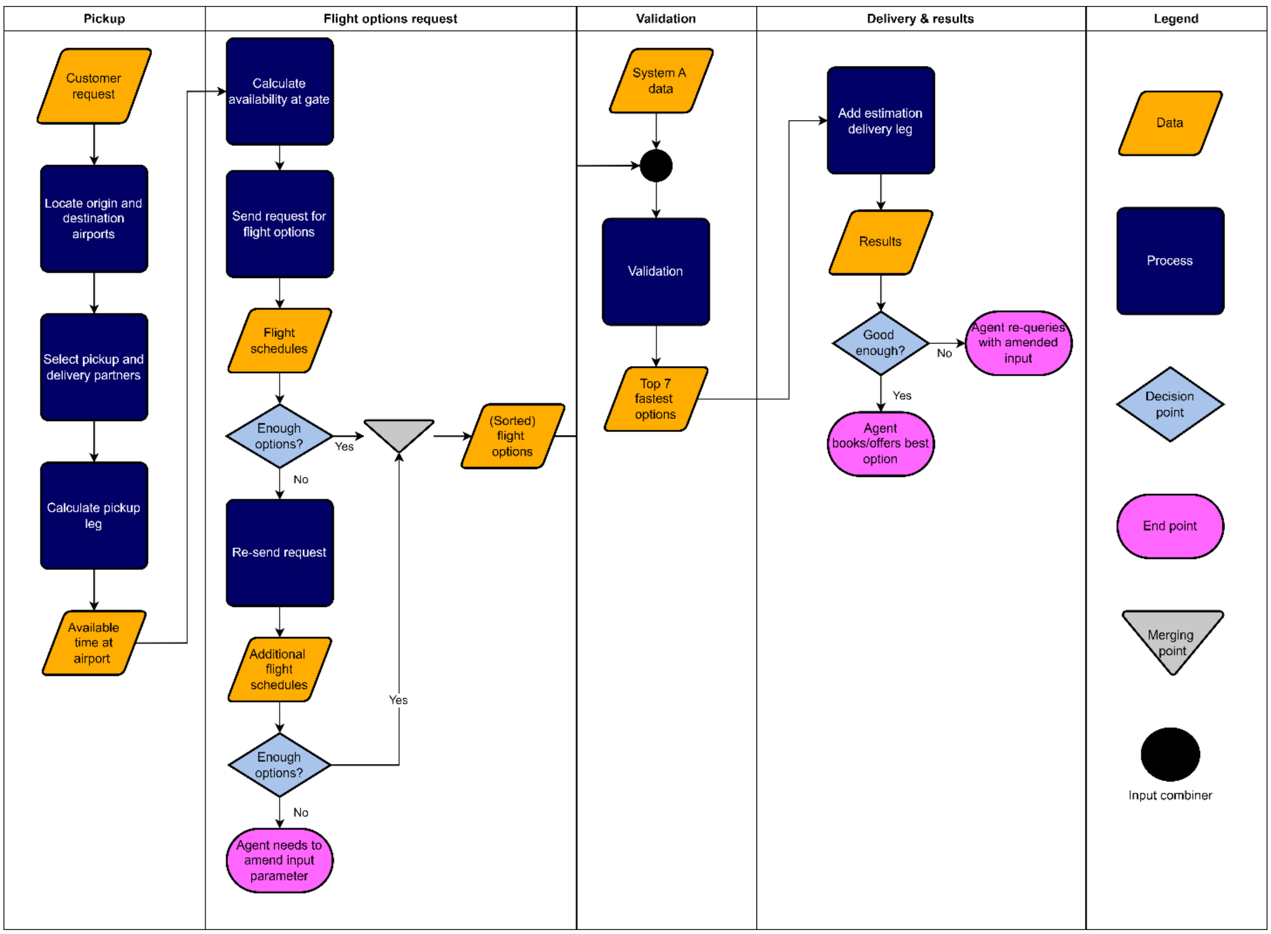


Figure 4: System interaction in route calculation. The main steps to generate a route consist of calculating the specifics to send within the API request, receiving the results and validating them against internal operating parameters.

2.1.2 Validation

This section explains the validation step in detail. The reason this is a necessary step in the process is that the flight data is owned and maintained by a separate company, which is not aware of the (constantly changing) business specifications SSLC enforces upon its routes. In fact, when combining the flight schedules, this external party does not take into consideration limitations imposed by, for example, contractual agreements between SSLC and its logistic partners. Therefore, the validation step is necessary as to ensure the presented options match internal commercial requirements. Following the process depicted in Figure 4, those requirements are used during two phases: a first set of requirements is sent within the API call to the scheduling entity when requesting flights; the remaining ones are used after receiving the flight options to filter out the infeasible ones.

API call parameters

Together with the information listed in Table 5, the following constraints are sent when requesting flight options (not all parameters are included in this overview, as some are not particularly relevant).

- Earliest departure time: this is the time at which the shipment is available at the origin airport's export station plus a time buffer of 30 minutes.
- Include/exclude airlines: this parameter specifies which airlines to include/exclude in the query, as to limit results only to SSLC partners.
- Number of hours: here the time window for departing options from the origin station is specified. This is set to 24 hours, which means options departing later than a day from the availability time of the package at the origin airport are excluded. In the proposed solution, we show this constraint can be relaxed, with beneficial effects on the quality of presented options.
- Max connections: specifies the maximum number of allowed transits. SSLC does not allow more than three connections, as each extra connection is associated with risk of transporting disruption. Also here we show that relaxing this constraint can be beneficial.
- Truck deployment: this parameter specifies that only aircraft can be used to transport goods between airports. In other words: either goods are transported between the pickup and delivery points directly by truck, or by transiting at airports, between which only flights are used. In our proposed solution we relax this constraint, as we are interested in observing whether using driving options between airports can be beneficial in any way to a route's quality.
- Payload type: this specifies the type of flight to be used (passenger or freighter). In almost all cases SSLC uses both options, except from when particular good types (e.g. lithium batteries) are being shipped.
- Include multiple carriers: this parameter specifies whether different airlines (i.e. belonging to different groups) can be used to operate on the same route, given there are more than one flight leg. This Boolean is set to false, as SSLC believes switching carrier increases the risk of irregularities. In our solution approach, we relax this constraint and show the benefits of combining multiple airline carriers on one route.
- Minimum connecting time: this is a static indication of the minimal time required for tail2tail transits. This value depends on the transiting station: for example, Frankfurt airport yields the shortest possible tail2tail transit (45 minutes), whereas at other airports, this value is significantly higher. This is a static parameter, meaning that based on its values, options with flight transits exceeding the threshold are not returned to agents. To avoid throwing away potential good routes, SSLC always sets it to the known minimum (i.e., 45 minutes). After receiving the flight options, SSLC'S system validates the results, as it needs to double-check whether the transit times

per leg are operable in practice, based on the known service times of each hub. This is a rather inefficient way of working, as we address in the next section.

Validation of results

After the request is sent, the external party fetches and matches its known flight schedules and returns up to 150 options. SSLC uses its own systems as to check the individual options' feasibility. On a high level, the following aspects are considered (actions that do not contribute to the overall understanding of the validation process are omitted):

- Check product airline: the carrier airlines are retrieved, and an internal check is performed to ensure the selected airlines can actually perform the transport. SSLC has different types of contracts with their partners, which grossly specify the type of shipments that can be transported, the route operated, etc. Please note that this is different from the preliminary selection of partner airlines, as a particular partner may still not be able to operate a route given contractual agreements. Flight options containing airlines that do not meet the contractual requirements are discarded.
- Check customs capabilities: based on shipment specifications and the origin/destination country, the system checks if customs clearance is needed. If affirmative, then another check is performed to see if the airports involved in the flight route have the capability to perform customs (i.e., it checks whether SSLC has a customs broker at the airport of interest).
- Check overall stations capabilities: this step ensures the stations involved in each transportation leg can sustain the route; in other words, whether there is at least one partner which can oversee the transportation leg given contractual agreements.
- Check for daily closed flights: additionally, a route cannot be utilized when it contains so-called closed flight numbers. The NS&P team has daily contact with the airline partners and can decide based on recent information (for example about capacity or delays) to exclude several flight options for a particular day. They do so by entering these flights in the system, which then takes them into account for this validation step.
- Check involved goods: this step ensures the goods can be transported by the carriers within each option, taking again contractual agreements between SSLC and partner airlines into account.
- Check aircraft capacity: given the shipment dimensions weight, the system checks whether it can be loaded on board of the selected aircraft type.
- Check stations opening times: per option, for each transit leg, this step ensures the leg takes place within the opening times of involved stations.
- Check transit times: this step checks whether a route's transits all have sufficient time, which depends on the airport visited for the transit. The system checks this using its internal database, where a list of required times per hub is kept.

2.1.3 Limitations of the routing system

While collecting the information described in Sections 2.1.1 and 2.1.2 we were able to identify and discuss, together with the involved stakeholders, several limitations to the way the current routing system works. This section briefly reflects upon these limitations and potential solutions.

Origin and destination airports

The first limitation is bound to the selection of the origin and destination airports, in case of door2door shipments. The system is currently able to compare just one single combination of airports at the time. Moreover, if the input fields are left blank, the airports are selected based on geographical distance from the pickup and delivery points. This can be problematic when for example, two minor (regional) airports are the closest to the designated pickup and delivery points. Minor airports present several disadvantages, like fewer flight connections, less facilities (e.g., customs brokers), etc. To overcome this, most agents rely on experience and manually amend the combination of airports to search for better (faster, more reliable) options. Although bigger hubs generally yield better results, the risk here is to overlook potentially superior flight connections from minor airports by holding this rule of thumb for truth. A way to overcome this issue would be to try out different combinations of origin and destination airports manually; this, however, can be time consuming and is therefore far from optimal. Hence, the problem is twofold: the algorithm only compares one single airport combination at the time and automatic airport selection is unaware of relevant aspects like hub size and required facilities.

To overcome these issues, an intuitive solution would be to send multiple API requests with different origin/destination combinations to the flight scheduling entity and to base airport selection on a set of logical rules rather than solely geographical distance. There are however still several issues that would need to be addressed within this approach; for example, how many separate airport combinations would be needed to provide with a better set of flight options, while keeping the computational burden within sustainable boundaries. Another one would be which aspects to consider when selecting airports to assign to pickup and delivery points and how to prioritize them in order to yield the best combinations. Therefore, we identify this as a deliverable for a new solution: a supporting system which selects an origin and destination airport using relevant criteria rather than solely distance from pickup/delivery.

Excessive waste

Another major drawback identified by the SSLC IT team is that flight route validation cannot be done entirely upfront. The consequence is that when checking flight options returned by the scheduler, the majority of them is discarded because of infeasibility. As an estimate, about 75% of the received schedules from the external planner are thrown away after the validation. This is problematic for two reasons: firstly, this is inefficient from a computational point of view. Secondly, this limits the number of potential options presented to the agent and thereby lowers the quality of transport solutions offered. If instead the validation could be performed upfront, hence by specifying all aspects listed in Section 2.1.2 into an API request, there would be potential of receiving back 150 valid flight solutions instead of, say, 40. Figure 5 illustrates this concept, where the upper diagram illustrates the process As-Is while the lower one suggests an improving To-Be scenario, where less data is wasted and computational effort is saved. This consideration brings us to the final point of discussion of this section.

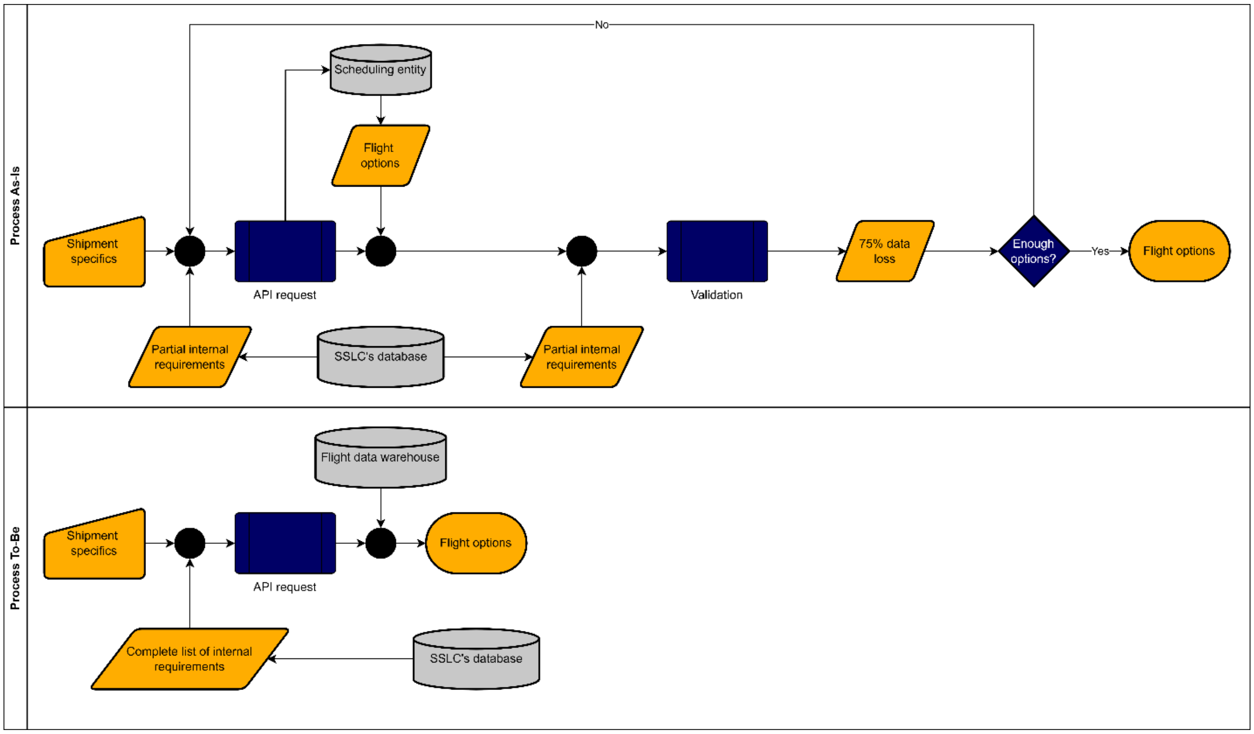


Figure 5: Current (As-Is) versus ideal (To-Be) validation process. As can be seen, validating data upstream would potentially save computational effort (repeating the request several times) and reduce the amount of data being discarded.

Data and algorithm ownership

The above-mentioned problem is bound to the fact that SSLC does not own neither of the flight schedules nor the algorithm combining the schedules. Therefore, it has only partial control on how to steer the routing process towards a more efficient workflow. For example, SSLC could ask their partner to change their technical implementation as to be able to send all their parameters listed in Section 2.1.2 upfront. However, there is no guarantee the partner will agree to it. On top of that, we know the scheduling partner combines flights as to find the fastest option. Nevertheless, this research argues that transit speed is not the only important criterion for selecting a flight route. At this point, the tradeoff SSLC would then need to make is between the cost of building an inhouse scheduling algorithm, acquiring and owning the necessary flight data, and the benefits this insourcing would yield for their business overall. Although the former aspect is not in scope for this research, we can partly enlighten the company on the latter by showing the degree by which they can improve their flight routes.

2.2 Route performance metrics

2.2.1 KPI selection

To design a measurement framework to analyze the quality of current routes, we first consult company experts on the matter. We do so to identify what objectives are important when it comes to route selection and, secondly, learn how these are measured currently at the company. First, we outline the panel of experts. As discussed in Chapter 1, this topic affects multiple teams within the organization. Among the ones discussed, we identify the Customer Service, NS&P and IT teams as being the ones most directly involved. We choose to pick at least two stakeholders for each of these groups, as to distinguish further team-bound perspectives from individual judgement. Within the CS team, we differentiate further between regular operations, CC&M and supervising agents, therefore we select at least one expert per subgroup. In total, we invite eight employees to contribute to this part of the research.

Participants are asked to fill in an online form; here, we ask them to state what metrics are used or they think should be adopted at SSLC as to measure a route's quality. From the respondents' answers, we learn three main objectives should be considered: speed, defined as how quickly a shipment can be transported from origin to destination; delay, defined as the likelihood to incur delays by operating a certain route; and finally operating costs. Although being mentioned in a few responses, we choose to exclude the environmental objective from the analysis as we understand that it is still not considered as determinant for a route selection as the previous three objectives.

After collection of the results, we perform a complementary literature research to ensure relevant KPIs are not overlooked. Appendix C lists the complete selection of KPIs gathered from the panel of experts and found literature. After careful analysis, we select five KPIs:

- Transit time: difference between actual arrival time at delivery and actual departure time at the pickup.
- Transit time quotient: ratio between route's transit time and longest transit time registered on that particular route.
- Delivery delay: difference between actual and planned times of final delivery.
- Total transit delay: sum of all occurred delays throughout the route.
- Total operating costs: total actual costs incurred to operate the route.

Before moving on, we need to make two clarifications. Firstly, there is a distinction in what transit time and transit time quotient measure. The first measures purely how long it takes to operate a particular route, from pickup to delivery points. The other one expresses the variability of a flight route, as it is a way to compare the registered transit time of a particular shipment with the worst transit time ever registered on that same route. In other words, a route with values closer to zero implies less variable (and thus supposedly stabler) flight transits than routes with values closer to one. The second distinction concerns the delay metrics. Whereas the delivery delay measures only the difference between planned and actual times of delivery (hence for the last leg), the total transit delay takes into consideration all delays occurred throughout the route, meaning it consists of the sum of delays of all transportation legs throughout the shipment's route. The reason to use the two metrics is that they also provide different perspectives on the stability of a route. In fact, the delivery delay mainly affects the end customer, and thereby is indicative for the quality of service provided. Nevertheless, a route which does not have (significant) delivery delays could still be problematic if it shows a notably high total delay value. In fact, a shipment might experience delay during transit and still arrive on time at the end customer. This affects the Customer Service: intermediate delays can require action by the monitoring agents, who need to ensure the delay is minimized and sometimes might need to intervene, by re-routing the shipment as to get it on time to the customer. The latter implicates additional operational costs for SSLC and workforce deployment; it is therefore relevant to account for when wanting to improve operations in general.

2.2.2 KPI prioritization

After having selected the KPIs, we prioritize them to understand their relative importance. This way, we seek ultimately to get an all-round measurement of a route's perceived quality. The Analytical Hierarchy Process (AHP) is a decision-making method for prioritizing alternatives when multiple criteria must be considered, and arguably one of the most widely adopted in the field of logistics (Nydick and Hill 1992). Generally, decision-makers compare criteria pairwise, and assign each couple a relative preference score (judgement) within a matrix. The judgements are then normalized and used to calculate priority scores, which are ultimately used as weights for the criteria. Figure 6 shows the scale we applied for assigning judgements to the criteria.

Verbal judgment or preference	Numerical rating
Extremely preferred	9
Very strongly preferred	7
Strongly preferred	5
Moderately preferred	3
Equally preferred	1
Intermediate values between two adjacent judgments (when compromise is needed)	2, 4, 6, and 8

Figure 6: AHP judgement scale (Nydick and Hill 1992)

In this setting, the KPIs are the criteria and the experts panel consulted in Section 2.2.1 are the decision makers. For incorporating multiple perspectives within the AHP, Forman & Peniwati (1997) propose two distinguished methods, namely the aggregation of individual judgements (AIJ) and the aggregation of individual priorities (AIP). As the names suggest, the former aggregates the judgements filled within the matrices, whereas the latter aggregates the obtained priorities from the normalized matrices. The discriminator for using either of the two is the heterogeneity of the consulted decision-making group. When the decision-makers act in a shared interest, it is preferable to use the AIJ; this is suited for when the group consists of people of the same organization or with a shared goal. The AIP is applied in the opposite case, hence when it presumable the decision-makers might have significantly different perspectives and it consequently becomes important to stress the differences. Although we acknowledge there would be good arguments to use either method, we choose to apply the AIJ, as at the end of the day, all stakeholders involved look at the situation from the perspective of a SSLC employee, which arguably surpasses any individual view on the KPIs' relative importance.

After collecting the panel's responses, we build an aggregated matrix by taking the geometric mean of the individual judgements. Then, we normalize the matrix and obtain the priority values. Finally, following the steps described by Lalla (2020), we check whether the obtained results are consistent. We do so by calculating the Consistency Ratio (C.R.). We obtain a C.R. of 0.04, which confirms the weights obtained are consistent (CR < 0.10). Appendix D shows the values and calculations in detail (please note that five matrices are used, as we register a dropout of three participants at this point). Table 7 shows the KPIs with their weights in descending order. Interestingly, there is uniformity in the order of importance of the KPIs and their objectives. In fact, following the weights, we observe delay being the most important objective, followed by speed and costs.

Table 7: Weighted KPIs

KPI	Weight
Delivery delay	0.259
Total transit delay	0.246
Transit time	0.236
Transit time quotient	0.178
Total operating costs	0.081

2.3 Current routes performance

2.3.1 Data preparation

Selection of historical order data

For the route performance analysis, we combine two datasets: the sales and tracking data of orders booked by SSLC Netherlands. Each row in the sales data table represents an order and contains an all-round overview of the route, including origin and destination, route transit airports, flight carriers and operational costs. The tracking data table contains multiple rows per order: each row consists of a transport leg for that order. Here we find information like timestamps for planned and actual transit times, and the responsible partner for the leg. Figure 7 displays the two tables with their attributes.

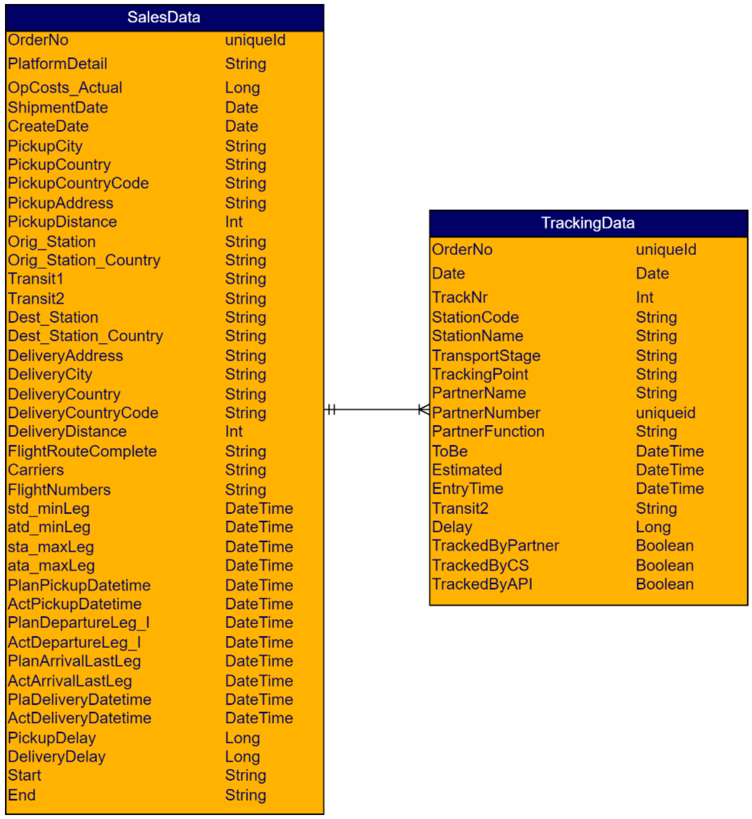


Figure 7: Initial datasets

The tracking dataset contains orders from the years 2021-2022 (up to July 7th, the last date we refreshed the data before analyzing it). The sales dataset contains data from 2017 onwards. Combining the two datasets means we would either need to drop the excessive datapoints from the sales table or extrapolate data from the tracking table. We choose for the former approach, as we deem extrapolation of such diverse data to be too complex. By matching the two tables by order number we obtain a new combined table containing 11032 unique order entries. Figure 8 displays the end table obtained with its attributes. In this new dataset, we introduce several calculated columns: the transit time, delivery delay, total transit delay, total operational costs and the number of airport transits. The latter measure is introduced as each transit implies additional risk of a disruptive event (e.g., missed connection) and therefore might be interesting while assessing risk. The original time values from the sales and tracking datasets are all set to local times. Therefore, calculating a time difference for a leg with two different time zones would yield incorrect values. To prevent this, we priorly convert all times to Central European Time. Next, we transform all calculated time values to minutes. Finally, we change the negative delay values (including both delivery and leg delays) to zero. These values are caused by shipments arriving earlier than scheduled at destination. Whereas taking negative values into account would distort the calculations of the weighted performance score of a route, we still do account for those shipments arriving early in our analysis. Section 2.3.3 provides more detail on these routes.

UniqueOrderNumbers	
OrderNo	uniqueId
Start	String
OriginStation	String
Transit1	String
Transit2	String
DestinationStation	String
Route	String
End	String
TransitTime	Long
DeliveryDelay	Long
TotalDelay	Long
TotalOpsCosts	Long
Transits	Int

Figure 8: Datasets combined

Grouping order routes by lane

Before proceeding with our analysis, we inspect the shipment data using the Map visualizer tool in Microsoft Power BI. We do so to get a first impression of the geographical collocation of the shipments' origins and destinations and see if we can spot interesting patters. Figure 9 displays a global view of the map graph we utilized. Here, a partial sample of randomly shuffled orders are plotted, with their corresponding pick-up and delivery points. We omit the type of location from the map as we consider it potentially sensitive information.

Figure 9: Pick-up and delivery points, global overview. The scarcity of extra-European points is motivated by missing datapoints in the tracking table.

By examining the map, two main considerations pop-out. Firstly, most orders are concentrated in Europe. We know by fact that Figure 9 is not representative for the complete routing spectrum offered by SSLC Netherlands. The scarcity of intercontinental orders in the available data is partly caused by a mismatch between the two databases we use. Whereas the sales table includes enough historical intercontinental orders, these are not present in the tracking table. This mismatch is caused by the lack of a data pipeline between the system where those orders are booked and the tracking database. Therefore, the analysis we carry in this section is unfortunately not exhaustive, considering the number of routes that are ignored due to this data mismatch.

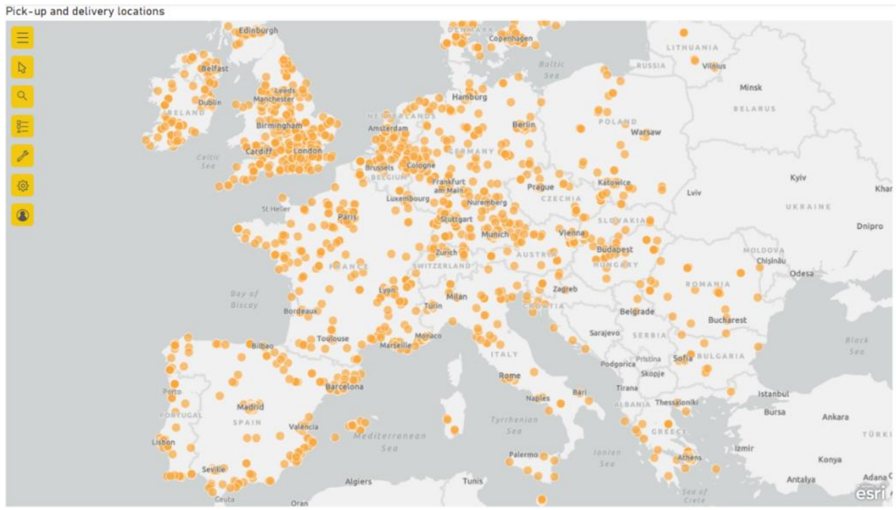


Figure 10: Locations sample of pick-up and delivery points, European view. The sparsity of locations makes it hard to sample them geographically and thereby to cluster routes.

The second and most important observation consists of the geographical sparsity of pick-up and delivery locations. We can observe this especially if we zoom in the European part of the map, as in Figure 10. When looking at a route singularly and without further reference points, we can make limited inferences on its optimality. Following this observation, it is far more interesting to cluster routes that share similar characteristics, like start and end points near to each other, and look at their differences/similarities in chosen transiting airports, flights and consequent KPI performances. To clarify this, let us take a fictive example. Suppose one route entails a shipment being picked up somewhere near the Amsterdam area and being delivered somewhere near Milan. The shipment is picked up, flies from Amsterdam to Frankfurt, then from Frankfurt to Milan and is delivered. By looking at this order solely, we would not be able to judge whether it is a good route or not. On the other hand, suppose we looked at three different orders with similar pick-up and delivery points: two of them stop in Munich instead of Frankfurt and display both shorter transit times, less delays but slightly higher costs. This is far more informative, as it might suggest flying over Munich would be preferrable than over Frankfurt. Similarly, grouping routes in this fashion facilitates comparison between different route clusters. Returning to what we observe in Figure 10, we can see that it becomes difficult to see a clear concentration of either pick-up or delivery points around a specific area. Therefore, we choose not to cluster routes based on proximity. Instead, we choose to group all routes that share the same combination of origin and destination airport. This approach has two advantages: it is simple to apply and non-arbitrary. Furthermore, we can assume most of the routes having the same origin-destination combination will have pick-up and destination points in comparable geographical areas.

Having defined our clustering method, we can calculate each route's transit time quotient: we do so by dividing each route's transit time by the highest transit time of that routing cluster. Finally, we introduce the concept of lane, to distinguish clusters from induvial routes. A route is a unique combination of pick-up and delivery points, visited airports and logistic partners involved; a lane is a unique combination of origin and destination airport, which groups all routes sharing that same combination. Following this definition, we group the 11032 routes into 1083 different lanes. We notice that 503 lanes are composed by single-order routes; following the reasoning discussed in the previous paragraph, we decide to exclude them from the analysis, leaving us with 580 lanes. For the remainder of this report, we omit the origin and destination airports of each lane, given it is considered sensitive information.

2.3.2 General performance analysis

Volume

Having defined the lanes, we first analyze which ones yield the most volume, hence number of orders. To do so, we use the ABC classification method. ABC classification is a widely adopted method within the domain of inventory management. Its primary objective is to classify Stock Keeping Units (SKUs) as to steer managerial attention; thereby, it channels the effort that should be put in inventory control strategies based on each SKU class. SKUs are generally ordered by decreasing turnover value and divided in three classes, each class encompassing items that cumulatively yield a percentage of the total turnover. A-class items, yielding together most of the turnover, should generally be monitored most closely, whereas C-class items should receive the least attention (Teunter, Babai and Syntetos 2009). In this case, the SKUs are represented by the flight lanes, and the classification parameter is the number of orders of each lane. Hence, we rank the lanes by decreasing number of orders, and classify them as follows: lanes yielding a cumulative volume value of 40% of all orders belong to A-class routes, lanes falling between the 40% and 80% are labelled as B-class routes, and the remainder as C-class. Figure 11 shows graphically the results of the analysis: the x-axis displays the percentage of lanes whereas the y-axis shows their cumulative order volume. As a result, 14 of the 580 lanes (approximately 2%) yield together 40% of the total order volume of the past two years. Interestingly, we observe that lanes behave following the famous Pareto principle, which states that within a population, 80% of the consequences (in this case the cumulative volume) comes from 20% of the causes (here the number of lanes), asserting an unequal relationship between inputs and outputs (Brock 2022).

ABC classification for route volume

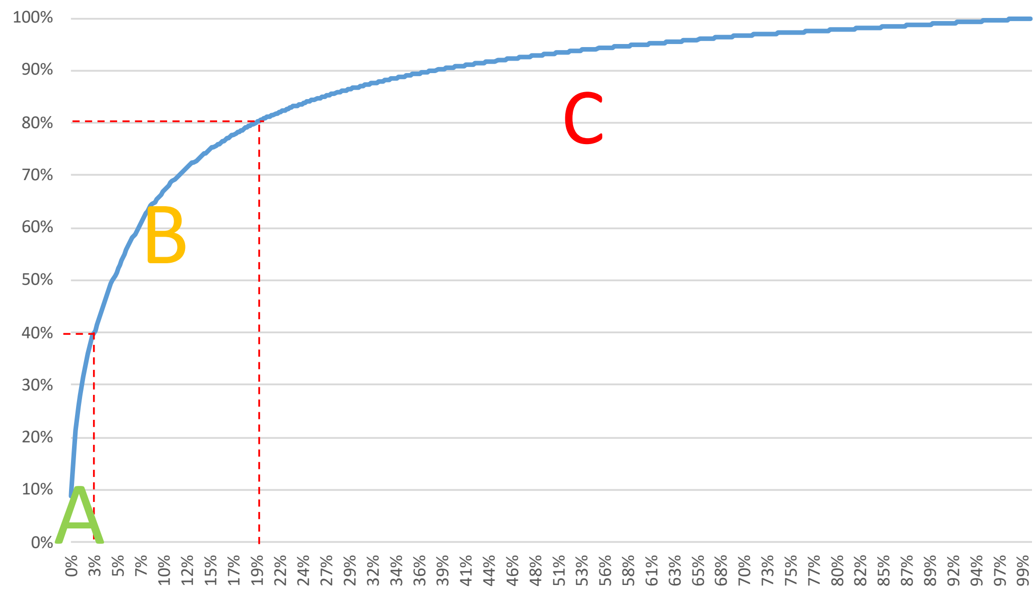


Figure 11: ABC classification of lanes according to cumulative volume. A-lanes yield together 40% of the number of orders and are the most frequently used ones, B-lanes an additional 40% and C-lanes the remaining 20%.

Performance

We use the same ABC principle to distinguish lanes based on their performance. For each lane, we calculate the average KPI values as described in Section 2.2 , then normalize them and obtain a weighted score using the weights presented in section 2.2.2 . By observing the lanes' weighted score values, we detect an anomaly in one of the KPIs' behaviors. In fact, we see that lanes performing relatively worse than others have a lower transit time quotient, whereas better performing lanes present a higher value for this metric. This is the opposite of what we predict would happen. When selecting this KPI, we expected that instable routes would present a higher transit time quotient, as their transit time would approach the highest (and therefore worst) transit time possible. We acknowledge, however, that the opposite is true. When a lane is stable there is low variation in flying transit times, as generally there is a direct connection between origin and destination airport. This low transit time variance naturally yields transit time quotient values closer to 1, as the "worst" registered transit time is close to the average value for that route. Needless to say, the opposite becomes true for unstable routes. Given that in this case, higher values for each KPI imply negative performance on a given lane, using the transit time quotient would distort the weighted score values. We overcome this issue by replacing it with another metric. We choose to use the transit time coefficient of variation: this is calculated as ratio between standard deviation and average of the routes' transit times within a lane. In this case, instable lanes present more variation in transit time, and hence a higher value for this chosen metric. The latter ensures our overall performance measurement remains consistent. To ensure the assigned weight for this KPI is still valid, we verify this change with the panel of experts. As the panel members do not wish to alter their registered judgments, we keep the same weight as for the transit time quotient. Having adjusted the metrics and recomputed the weighted scores, we further inspect our data for eventual outliers. Figure 12 displays a box plot containing the lanes' weighted scores. Of the 580 data points, we observe 42 outliers, labelled as blue dots in the graph.

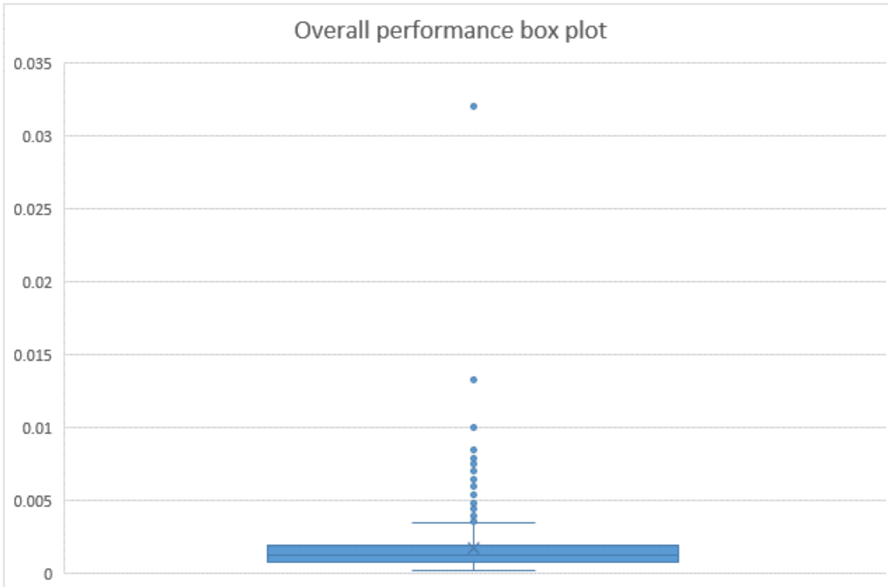


Figure 12: Identification of outliers using a box plot. The plotted data consists of the aggregated score per lane, lanes with an extremely negative score compared to the average are plotted as single points (above the box plot).

For some lanes we know that the abnormal values are caused by exceptional situations in the past, like major strikes or impeditive weather conditions. However, it is hard to distinguish for all outliers whether these are caused by such exceptional cases or additionally by incorrectly registered data values. In the latter case, it would be good practice to drop the outliers from the analysis. Since this is however not clearly the case, we choose to keep them for the data analysis. Even though some values might not be fully representative for a lane's general performance, these outliers still highlight a fact that holds for all logistic operations in general: sometimes, things can just go oddly wrong.

By examining the lanes in general, we already observe some interesting figures. Of all orders, grossly 68% are delivered on time (all routes with a delivery delay greater than zero minutes are considered late). More specifically, 17.5% of all orders have a delay greater than 30 minutes and only 11.6% higher than 1 hour. Another interesting observation is that of all routes that incur a delay during transit, about 71% are recovered and delivered on time at the customer. The explanation for this number is twofold. Firstly, this is because the routing system adds (apparently sufficient) buffer times between transportation legs, as to ensure that intermediate delays do not lead to severe consequences, like missing a flight. Secondly, this is also indicative of the amount of effort the CC&M team puts in ensuring shipments are recovered and delivered on time. Whereas this number can be seen as positive from a customer service perspective, punctuality is arguably not the only important thing in logistics. In Section 2.3.3 we analyze these on-time orders in more detail and observe that they can indeed be improved on other KPIs.

Using the same ABC method as illustrated earlier, we make a classification based on the lanes' weighted scores. The C routes include all routes which, cumulatively, take up to 40% of the total weighted score sum; these are hence the worst performing routes. Similarly, B lanes lie between the 40% and 80%, and A lanes are the remaining and hence best ones. Figure 13 shows the obtained classification. We observe that 83 lanes (approximately 14% of the total) are classified as worst performing; 237 (41%) perform medium-well and 260 (45%) belong to the best-performing tier.

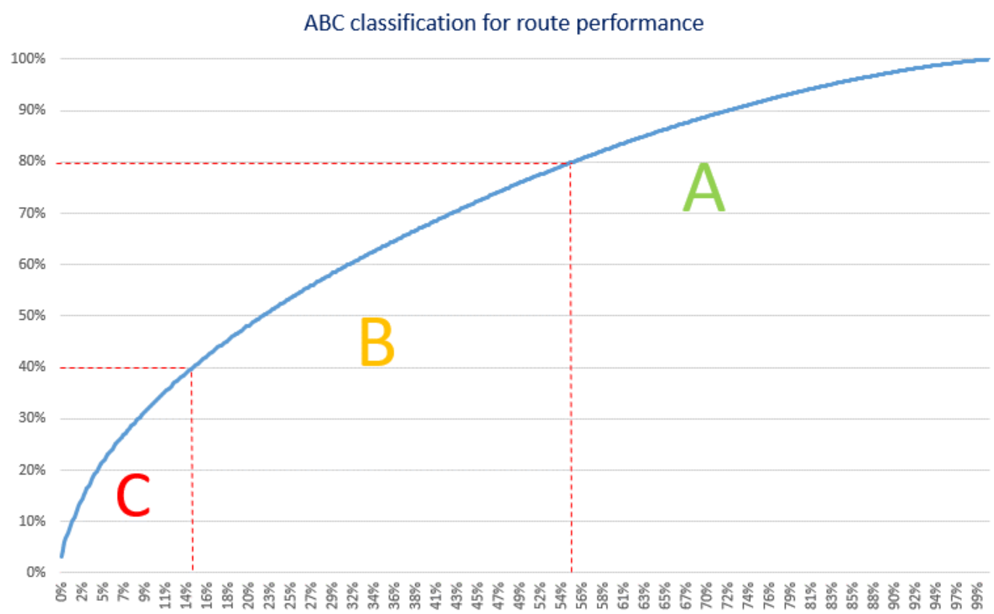


Figure 13: ABC lane classification from worst to best weighted performance

Following this analysis, we intersect the results showed in Figure 11 and Figure 13 to observe the distribution of lanes in terms of both volume and performance. Table 8 shows how each volume-based group (rows) is distributed among the performance-based groups (columns) of lanes. For example, within the A-volume routes, we observe 14% belong to the C performance class (worst), 36% to B (medium) and 50% to A (best).

Table 8: Volume-performance comparison.

Code	%Volume	C performance (worst)	B performance (medium)	A performance (best)
A	40%	14%	36%	50%
B	80%	9%	65%	26%
C	<80%	15%	36%	49%

This table helps us in localizing where SSLC should focus when improving their routing. For example, we can argue that the lanes in the AC class need improvement most urgently, as they are the most used and also worst performing lanes at the moment. A large majority of medium-busy lanes belongs to B-performance class: this also suggests there is room to shift them towards the A-performance group.

Interestingly, most of the least intensively used lanes perform relatively well: therefore, we argue these are the ones that need least attention right now.

2.3.3 In-depth analysis

This section zooms into a selection of best and worst performing lanes. First, we examine the routes that are delivered on-time (or earlier). The majority of these routes belongs to the A-performance class, which is not a surprise: these routes perform best on delay-related metrics, which are on turn the ones with the highest weights on overall performance. We are interested in observing whether other KPIs suggest room for improvement. Next, we take the A- and C-performance lanes and compare their KPI values, in order to understand the magnitude of the gap between them. Finally, we look at the worst-performers, and make some considerations.

Early deliveries

As described in the previous section, about 68% of the analyzed routes is delivered (at least) on-time at the customer. Among these routes, we observe that about 46% is delivered from zero to half hour earlier than agreed. Of the remaining orders, 27% are delivered between 30 and 60 minutes earlier, and the remaining 27% arrives over one hour earlier at destination. We point out that SSLC does not face any directly measurable consequence for deviating from the agreed delivery time. In fact, although punctuality and speed of service are essential selling points of the company, a delay in parcel delivery does not directly imply cost penalties; nevertheless, delivering systematically late can naturally lead to the termination of a business relation. In the same way, SSLC does not gain any type of direct benefit by delivering earlier than agreed. Given the overwhelming number of early deliveries, we are interested in discovering whether this has consequences for the performance on the other measured KPIs. We subdivide these orders into the three groups mentioned based on earliness. Table 9 shows the three groups and their KPI performance. The first column shows the average deviation from the agreed time per group: please note that the negative sign indicates the delay is negative, hence the orders are delivered early. When examining the total transit delay values, we learn that over 94% of the orders incurs a delay during transit of at least 30 minutes. By dividing each route's transit delay by their number of transit legs we obtain their average transit delay per leg. Overall, the average delay per leg is 1 hour and 45 minutes. Thus, on average, using this value as buffer in between transit legs can prevent a shipment from arriving late at the end destination. If we look at the category of lanes represented in this group, we see the overwhelming majority belongs to the A-performance class. This does not come as a surprise, since these routes perform best on delay-related metrics, which are on turn the ones with the highest weights on overall performance. Nevertheless, if we compare them with the orders that are not delivered on time, we see that the former group performs better on all KPIs in just 41% of the cases. In particular, transit time seems to be the most penalized KPI. This fact yet underlines the tradeoff that needs to be made between delivering shipments fast and incurring risks of delay. Finally, if we look again at the on-time group of routes and divide them according to how early they are delivered (one group consisting of orders delivered up to 30 minutes earlier, the second between 30 and 60 minutes earlier and the third with more than one hour earlier) we see that the middle group performs best in general. Table 9 displays the average KPI values for each group. Interestingly, if we normalize and add them using the weights, these suggest that shipments arriving between 30 to 60 minutes early perform better than the other groups. All-in-all, these observations confirm that even for this group of routes, there is room for improving KPIs.

Table 9: Performance of the on-time orders. Orders are divided in three groups based on earliness: 30 to 0 minutes early, more than 30 minutes up to 1 hour early and over 1 hour early delivery. Their KPI performance is evaluated and compared. All times are shown in minutes; negative values imply earliness w.r.t. the agreed time.

Order type	Average delivery earliness	Average total transit delay	Average delay per leg	Average transit time	Average transit time cv	Average costs	#orders	percentage
30 to 0 minutes early	-11.37	675.55	127.14	1400.99	0.22	387.71	3414	46%
30 to 60 minutes early	-43.68	385.00	78.29	1295.01	0.20	416.37	2041	27%
over 1 hour early	-172.69	456.19	91.76	1309.39	0.19	454.46	2032	27%

Best-worst gap

Table 10 provides a summary of the difference per KPI between A- and C- performance lanes (please note all time-related metrics are expressed in minutes, costs in euros).

Table 10: Comparison current best/worst lanes per KPI.

Best/worst comparison	Number of lanes	Average bookings per lane	Mean transit time	Mean transit time CV	Mean delivery delay	Mean total delay	Mean ops costs
Best lanes	260	12	2906.16	0.24	16.85	452.83	374.01
Worst lanes	83	22	2703.69	0.64	776.46	3457.6	427.58

The most striking difference resides in the delay-related metrics, which are also being identified as most relevant to describe a lane's quality. Another metric popping up is the difference in flight stability: whereas the best lanes have a variance relative to the mean closer to zero, the worst one's approach one on this metric. On the other hand, operational costs incurred do not seem to be significantly different: this observation partially dismisses our initial hypothesis that delays might be directly correlated with costs. As a next step, we further investigate on factors that can potentially explain the differences in time-related performance.

Surprisingly, the only factor which shows a significant difference is the involvement of customs clearance during transit. Whereas only 7% of the best-performing lanes involves customs operations, an overwhelming 60% of the worst lanes must deal with them. This is not by chance: when looking further in the timestamps data of the C-worst routes, we observe the most frequent delays occur within the customs legs. When looking at the involvement of major European hubs, we observe that 19% of the worst lanes either depart or arrive at those, whereas for the best lanes this number is just about 29%. Finally, the average number of transit airports are similar, with 0.7 transits for the worst lanes and 0.5 for the best ones.

C-performance routes examination

To conclude this analysis, we look closer at the worst lanes, to see if more patterns can be discovered. The first thing we notice, is that almost half of them (46%) consist of lanes with origin and/or destination station in the UK. This might indicate the negative impact of Brexit on SSLC'S operations in general. Having identified delay as the worst performing KPI type, we look at each lane to see on which transit leg the most delays occur, where these initiate and what is the scale of the biggest registered delays. In general, we observe that the first delays occur quite early in the process, as almost 50% of the lanes experience their first delay during pickup, whereas the remainder occurs at export clearance (which is usually performed at the airport prior to flight). The average value of the lanes' initial delay is about 3 hours, which is significant: such a delay may easily lead to a parcel missing its departing flight. If we look closer to the irregularities registered in the data, it becomes however difficult to distinguish a clear cause of the delays. Delay causers within pickup can be several: it might just be as likely the courier's fault as the customer's (for example, some customers provide wrong shipping addresses which cause delay). Being this not specified in the data at hand, we cannot make any inferences on a specific cause. Nevertheless, it is no surprise that the worst delays occur overall during the delivery leg (UK shipments represent an exception, here the leg with most delays is the import clearance). As delay initiates early during transport, it accumulates creating a snowball effect; this leads ultimately to an average worst delay value of approximately 28 hours, considering all worst lanes.

2.4 Conclusion

In this chapter, we first answer the question: "How does the current system in use calculate a route?". The process of calculating an air freight route starts with an incoming customer request for a transportation job, which includes information about the pickup and delivery locations, goods to be transported and the shipment's availability time. An operations agent enters this information in the system, which on its turn interacts with an external data provider and returns a set of possible routes, all using the same origin/destination combination and sorted by final arrival time at delivery. There are several bottlenecks within this process, including the fact that flight schedules are combined to provide only with the fastest option, disregarding other important aspects such as disruption risk and costs. Furthermore, the system limits the variety of options as it is only capable of comparing one combination of origin/destination at the time; this affects the quality of the returned options in general.

Next, we define a set of metrics to measure the routes' quality and prioritize them, answering the research question: "How can we measure routing performance?". By consulting related works and a panel of experts, we define transit time, transit time coefficient of variation, delivery delay, total transit delay and operational costs being measures of interest. By applying the AHP we assign weights and combine them in a linear weighted function, which integrally measures route performance.

As a final step, we proceed answering the question: "How do the air freight routes at SSLC Netherlands perform?". In general, we observe that the majority (68%) of orders arrives on time at the end customer. Of those orders, a significant number is delivered earlier than agreed. From further inspection we learn that these early arrivers do not perform better than other routes on remaining KPIs, meaning there is room for improvement. Of all lanes, 14% are being used intensively by SSLC and at the same time score the worst KPI values. Given their strategic importance, those are the lanes on which SSLC should focus for a performance improvement. Finally, by comparing worst and best lanes, we learn the overall gap is significant. In particular, we see the biggest differences in the delay-related metrics: a possible explicator of such difference might be related to the involvement of customs, and the choice of origin and destination airports in general.

Chapter 3 - Literature review

This chapter collects and summarizes existing literature on two main topics. In Section 3.1 we gather information about existing routing problems that resemble the context at SSLC and explain their similarities and differences. After this comparison, we select the problem formulation closest to the one we are dealing with. As a final point, we state the changes that need to be applied to the formulation found in literature as to adapt it to our real-world problem. Thus, in this chapter, we answer the research question: "Which routing optimization problem fits best to the context of SSLC?". Next, having selected a base problem, we look further for mathematical approaches to solve it in Section 3.2 . By comparing and selecting a solution method, we answer the research question: "What is proposed in literature to solve the SSLC routing problem?".

3.1 Problem identification

3.1.1 Multimodal transportation

We first explore existing areas of study within the field of multimodal transportation. Multimodal freight transportation is defined as the transportation of goods carried out by at least two different modalities (SteadieSeifi, et al. 2014) and is thus essential to the core business for freight forwarders like SSLC. Archetti, Peirano and Speranza (2021) categorize problems based on their decisions:

- Location problems focus on optimizing the location of facilities within the transportation network.
- Network design problems optimize the definition of networks in general (location problems can be viewed as a specific case of network design problems).
- Scheduling problems are concerned with the optimization of service, transshipment, yard operations and transport schedules.
- Transportation problems entail deciding on the best set of services used to transport commodities from one location to another.
- Resource allocation problems deal with deciding how resources should be allocated to different locations/operations.
- Routing problems encompass identifying the route taken by either a single transportation unit or a set of multiple transportation units (this can be viewed as a specific subgroup of transportation problems).

Additionally, both Archetti, Peirano and Speranza (2021) and SteadieSeifi et al. (2014) distinguish planning problems by their decision time horizon. Here we discriminate between strategic, tactical and operational problems

Strategic planning problems

Strategic planning problems are concerned with long-term decisions, mostly related to infrastructures, such as to modifying an existing transportation network or designing a new one (Archetti, Peirano and Speranza 2021). (Hub) location and network design problems are the most common types of multimodal strategic problems (SteadieSeifi, et al. 2014), followed by scheduling and resource allocation problems (Archetti, Peirano and Speranza 2021).

Tactical planning problems

Tactical planning is oriented towards medium-term decisions and is mostly concerned with using existing assets optimally. SteadieSeifi et al. (2014) identify two main categories of tactical planning problems: network flow planning problems and service network design problems. Archetti, Peirano and Speranza (2021) mention network design, scheduling and resource allocation problems of tactical nature for air, rail, maritime and multiple modality networks. Macharis and Bontekoning (2004) add a new category of tactical problems, which addresses the pricing strategies for the transportation of goods.

Operational problems

Operational problems deal with short-term decisions, and are mainly concerned with deploying already assigned assets, incorporating real-time requirements. These planning problems are affected by dynamicity and stochasticity that are not explicitly addressed at strategic and tactical level, making them therefore generally more complex. SteadieSeifi et al. (2014) distinguish operational problems in fleet management and resource allocation and itinerary planning problems. The former type is concerned with distribution of resources throughout the network, whereas the latter is focused on real-time optimization of schedules and/or routes in general. Archetti, Peirano and Speranza (2021) identify operational problems of either transportation, scheduling or routing nature. They also mention an overall increasing interest in recovery problems, which are concerned with the re-optimization of operations following an unexpected planning disruption.

3.1.2 Operational routing problems in graph representations

This section focuses on the existing literature regarding operational routing and transportation problems. Within the problem context at SSLC, decision-making is on a short-term (routes need to be calculated with ad-hoc information), making it an operational problem. Furthermore, a decision on both services (which partners to select for a shipment) and route (which airports to be visited) must be taken, making it eligible for belonging to both the transportation and routing problem types, following the definition provided by Archetti, Peirano and Speranza (2021).

Eldrandaly, Ahmed and AbdAllah (2008) classify routing problems into two main categories, namely path finding and tour construction problems. The first category includes problems whose main objective is to find the shortest path between given locations, while minimizing a prespecified cost function. These are relatively simpler than the other family of routing problems, which aims at building a complete tour in an existing network under a set of constraints. Among these, the Travelling Salesman Problem, Vehicle Routing Problem and Bus Routing Problem are well-known subgroups with a rich research history. Since our focus is to find a route from a specific starting point to a distinct destination (hence, not a tour), this family of problems is disregarded. Both problem categories typically use graph modelling techniques to represent their transportation network. Therefore, before proceeding to discuss the problem type relevant to our research context, we introduce some concepts of graph modelling. A logistic network can be represented by a set of nodes (or vertices) connected by edges (or arcs), which altogether form a graph. Conventionally, nodes are used to represent locations; these can be either starting, transit or destination points within the transportation network. Edges represent corridors from one location to another and are often assigned with a weight. This weight can represent the time, cost or any other measure incurred by traversing that edge. A route is then defined as a sequence of visited nodes and thereby traversed edges. Given this definition, we can represent SSLC'S network by a set of nodes, which represent either pickup or delivery locations and airports, and edges, which represent connections (either by vehicle or plane) between them. In the following paragraphs we discuss how graphs can be modelled based on the real-life setting of the problem under examination. We do so to define the type of graph we can use to model SSLC'S network, as this helps us in narrowing the search towards the best fitting type of path finding problem.

There are several variants in graph modelling. The first and most significant characteristic is how the edges behave: here we distinguish between static or dynamic, and deterministic or stochastic networks (Chabini 1997). When the graph is static, the edge weights (hence the cost incurred by traversing a particular edge) do not change over time. Alternatively, edge weights can be represented by a function of time; in this case the graph is called dynamic or time-dependent (Wellman, Ford and Larson 1995). The latter scenario arises, for example, when transportation is performed by means of scheduled services (e.g., public transportation). In this case, depending on the arrival time at the starting node, the total traversal time is equal to the predefined edge weight plus some waiting time. A network is said to be deterministic when the edge weights are known with certainty. When randomness comes into play, they are represented by random variables following some probability distribution (Gendreau, Ghiani and Guerriero 2015). In most real-life settings edges tend to be rather stochastic: transportation networks are generally subject to random factors, like traffic and weather conditions, which hence influence travel times and costs (Chabini 1997). Although transportation networks generally behave in a dynamic, stochastic fashion, it can be beneficial to simplify them by modelling either static and/or deterministic edge weights, as these variants are easier to solve. Dynamic deterministic methods can be reduced to their static counterpart by applying time-expansion. The latter means that each time-dependent edge is modelled by multiple distinct edges, each one containing the weight value corresponding to a discrete time interval. This way, the graph is indeed expanded into a static, deterministic version (Chabini 1997). Similarly, stochastic graphs can be simplified to a deterministic variant by taking the estimates of the probability distributions for each edge (Miller-Hooks and Mahmassani 1998b) or alternatively the lowest/highest value (Miller-Hooks and Mahmassani 1998a).

In addition to edge behavior, Chabini (1997) distinguishes graph-based path finding problems based on their time representation (continuous vs. discrete), whether the First-In-First-Out (FIFO) property holds (this implies that departing at a later time from a given node cannot result in an earlier arrival time at its successor), whether waiting at nodes is allowed, the time horizon (finite or infinite), how many routes are generated (one, all possible, k-routes), the sign of link travel times (integer or real valued) and the type of objective (minimum cost, fastest path, most reliable paths, bi- or multi-criteria). A final distinction can be made based on the type of routing strategy used. The most classical approach to path-finding problems dictates calculating a route a priori. This means that the sequence of nodes and edges is determined based on known information beforehand; this sequence remains fixed and ignores potentially evolving information on the edge weights. Fu (2000) calls this strategy Non-Adaptive Routing (NAR). The opposite strategy is to decide at each visited node on which one to visit next. With this Closed-Loop Adaptive Routing (CAR), the outcome of a solution approach is a set of rules that dictate which edge to traverse next (called policy) based on the information at hand, rather than a fixed sequence of nodes and edges to visit. This information regards changing edge weights in dynamic graphs and enables the decision maker to generally make more robust routing choices. A hybrid approach is represented by Open-Loop Adaptive Routing (OAR), where an initial route is generated and at each node, the route is re-evaluated with real-time information collected during traversal (Fu 2000).

Starting with the desired output for our path-finding problem, we seek to find a complete route a priori. Agents need to be able to offer the route to their customer and then to book all required services to operate it beforehand, for which a closed-loop adaptive routing strategy would not be fit. Although we acknowledge accounting for real-time information can significantly improve a route's robustness, we choose not to take open-loop adaptive strategies into account, as this would significantly complicate our solution approach, besides being out of scope for this research. Instead, to incorporate variation into our model we use simulation-based techniques, more detail in Section 3.2.2 . Like in most real-life settings, SSLC'S transportation network shows both stochasticity and time-dependency. The former is caused by the uncertainty in transport times due to disruptions and delays, whereas the latter is a consequence of the use of scheduled flights for transport. However, we point out that modelling such network behavior would immediately imply a high degree of complexity. To avoid this, we simplify its stochastic nature by using estimations of travel times, costs and delays based on historical data. Hence, we assume edge weights are deterministic. On the other hand, we cannot disregard time-dependency, as we need to account for the flight schedules. To do so, several approaches (in addition to graph-expansion) exist: we discuss them in more detail in the next section. In addition, we assume a FIFO network: this is reasonable since, to the best of our knowledge, there is no way a flight can depart later than a previous flight from the same starting airport and arrive earlier at the same destination airport. Finally, we allow waiting at nodes and assume a finite planning time horizon.

3.1.3 The Multimodal Route Choice Problem

In this section we discuss publications found on the Multimodal Route Choice Problem (MRCP). The MRCP is a path-finding problem, most commonly formulated as a mixed-integer problem (MIP), where the shortest path is sought in a multimodal network, possibly by following either one or multiple objectives. The decision variable consists not only in selecting the sequence of edges to traverse in the network, but also the mode of transport used at each edge. This aspect is key to our problem setting, where we must not only choose which sequence of airports to visit, but also the partner for each transportation leg. To scope our research, we focus on a select group of publications. We review only MRCP-related works that account for the time component of transportation services within their models. Furthermore, we are only interested in problems related to freight transportation, where the route is found a priori and that employ quantitative solution methods. Finally, since we are dealing with multiple objectives (time, cost and delay risk), we review publications which at least optimize two of the three objectives. Some of the reviewed publications seemingly only optimize costs of transportation: however, in those cost functions time-bound aspects are included (like extra costs incurred by longer travel times or penalties for deviating from the agreed delivery time). Therefore, we take them into account as they indirectly minimize travel time as well.

In the previous section we introduce the need to account for time-dependency in the model as we are dealing with scheduled transportation services. The analyzed publications present various alternatives for dealing with this aspect, therefore we group them by similarity of approach. Ayed et al

(2010), Dong, Li and Zheng (2013), Song and Chen (2007) and Li, Negenborg and De Schutter (2013) use graph-expansion. To solve their MIPs, respectively, they use an Ant Colony Optimization (ACO) heuristic, (again) ACO, a backward label-correcting algorithm and a sequential Linear Programming approach. There is one major drawback in using graph-expansion when modelling time-dependency. With a large planning time horizon, this method can lead to a significant increase in the solution space, with consequent elevation of the computational effort to find an optimal solution (Nielsen, Andersen and Pertolani 2013). This makes this approach effective only to small-sized problem instances. Given the large number of departing flights per day per airport and the elevated number of airports within SSLC'S network, using graph-expansion would lead to an explosion in the number of modelled edges, thereby increasing the model's complexity. Therefore, we disregard this approach.

Most reviewed works use time-window constraints instead. Soft time-windows entail that traversing edges outside the indicated time intervals is still permitted but yields some kind of penalty, whereas in hard time-windows, traversal is rigidly constrained to the set boundaries (Sun and Li 2019). Ayar and Yaman (2012) model scheduled services explicitly using hard time-windows constraints. The authors' main contribution is proving that the general MRCP belongs to the class of NP-hard problems, for which, especially in large problem instances, heuristic algorithms are often more suited than exact solution methods. Their claim is supported by the overwhelming majority of MRCP publications that use approximation methods. Of the twenty-three reviewed works, only eight use exact solution methods. Also Xiong and Wang (2014) implement hard time-windows to model scheduled services; additionally, they also assign a delivery timeslot chosen by the decision maker. They solve the problem with the objective of minimizing cost and travel time by using a bi-level multi-objective Taguchi genetic algorithm. Peng, Yang and Luo (2020) use hard time-window constraints at nodes; they minimize transportation costs and travel time and prove the NSGA II algorithm to be a very suited solution method. Kumar et al. (2014) solve a similar problem instance with a two-staged method: first, they apply a nested partition method to obtain feasible solutions for both objectives. Second, they propose a Technique for Order Preference by Similarity to the Ideal Solution (TOPSIS), a multi-attribute decision making tool, to choose among the feasible solutions. Hrusovsky et al. (2016) similarly minimize costs, travel time and emissions, and combine the exact solution of the deterministic problem formulation and simulation to improve their solution's robustness. Domuta et al. (2012) formulate a MRCP using hard time-windows to minimize travel time and costs. In their proposed algorithm, they take time-dependency into account, and show it is effective for solving the problem. Sun and Lang (2015), propose a MIP formulation to minimize costs and carbon emissions, and use hard constraints to model the scheduled services' departure times. Chang (2008) proposes an alternative use of hard time-window constraints: he models edge traversal costs as a piecewise linear function with values dependent on the time-window selected at each node. He solves the problem by applying a Lagrangian relaxation and reoptimization approach. Finally, Mnif and Bouamama (2017a) also use hard time-windows to formulate their MRCP. They minimize costs and travel time for a real transportation case in the fertilizers industry in China and use linear programming as solution method.

The same authors formulate an alternative MRCP using soft time-window constraints for scheduled services. To ensure timeliness, they propose moving the minimization of penalties incurred by delivering either too early or too late within the objective cost function, rather than enforcing timeliness by constraints. They argue this provides more flexibility in the solution-finding process; they solve the problem again with linear programming (Mnif and Bouamama 2017b). A similar approach is used by Zografos and Androutsopoulos (2008), who minimize the travel time, the number of transits (visited nodes) and waiting time at the nodes, the latter being penalized in the cost function. Unlike most similar works, they propose a dynamic programming-based label setting algorithm to solve the problem. Chen and Lai (2015) use a similar time modelling for their problem based on a Taiwanese transportation network. They choose a multi-objective MIP-formulation, but reduce the problem to a single-objective problem using a weighting method and a normalization process and solve it exactly using CPLEX. Archetti and Peirano (2019) formulate a type of MRCP tailored to the context of air freight forwarding, making it particularly interesting to the research context. In their subsequent publication, they propose a swarm intelligence-based metaheuristic to solve it (Angelelli, Archetti and Peirano 2020). Like in the previous two works, the authors use soft time-windows and penalty parameters in the cost function to incorporate the time-dependency in the transportation networks. Unlike most works, they promote arriving earlier than an agreed delivery time with a benefit parameter. Lei at al. (2014) minimize three objectives: costs (including penalties for either early or late arrival, like in Mnif and Bouamama (2017a), Chen and Lai (2015) and Zografos and Androutsopoulos (2008)), travel time and risk. The latter is modelled by two distinct parameters associated with the risk of traversing a particular edge, using a transportation mode, and risk of visiting a certain node. The authors formulate the problem as an MIP and propose a swarm intelligence heuristic to solve it. Li et al. (2011) propse a genetic algorithm for solving a very similar problem. Given its explicit modelling of three objectives which are relevant to our own setting, these papers are of particular interest. Finally, Sun and Li (2019) propose an alternative way of including time-windows and time dependence, namely by introducing concepts from fuzzy programming to model soft time-windows at nodes and fuzzy edge costs. Fuzzy programming is yet another way of dealing with parameter uncertainty; parameters can be represented by a triangular set of estimations, taking either the most negative, positive and likely variable value. This method is often used when it becomes hard to estimate probability distributions for such variables directly, and estimations are ought to be done by means of rather qualitative data Sun and Li (2019).

Unlike the publications mentioned so far, Kaewfak, Ammarapala and Huynh (2021), Kengpol and Tuammee (2014), and Koohathongsumrit and Meethom (2020) do not explicitly consider time-dependency into their models as to represent scheduled services. Instead, they consider methods that form and rank possible solutions based on pre-set goal levels for relevant objectives. These goal levels are however also formulated in terms of timeliness of transportation. Arguably, this can also be considered as an alternative to account for time-dependency within the network. Furthermore, all works provide with interesting frameworks that allow to model qualitative risk factors converting them to goals to be evaluated quantitatively; therefore, we include them into our review. To solve the problem, Kaewfak, Ammarapala and Huynh (2021) and Kengpol and Tuammee (2014) use Zero-One Goal Programming (ZOGP) whereas Koohathongsumrit and Meethom (2020) employ TOPSIS.

So far, we described MRCP formulations differing in optimization objectives, modelling time components and solution approaches. Needless to say, there are other ways to discriminate between modeling variations, depending on the nature of the network being studied. Sun, Lang and Wang (2015) and Sun and Li (2019) both propose interesting schemes for classifying the Multimodal Choice Route

Problem. By combining their works with aspects we consider in Section 3.1.2 , we derive the classification scheme displayed by Figure 14. The MRCP attributes this scheme presents are:

- **Optimization criteria:** can be either a single or multiple cost functions.
- **Transportation service:** transport services for travelling on arcs can be flexible (i.e. not time-bound), scheduled or mixed.
- **Network resources:** the network can be subject to capacity constraints or be uncapacitated
- **Network behavior:** this describes whether the network presents some form of uncertainty (stochastic) or not (deterministic).
- **Commodity integrity:** this is whether the transported commodities are splitable at nodes or not
- **Planning strategy:** this aspect is related to whether routes are planned a-priori, dynamically or in a hybrid way.
- **Commodity type:** whether the transportation problem is about a single or multiple commodities.
- **Consolidation:** whether consolidating the commodities is allowed or not.

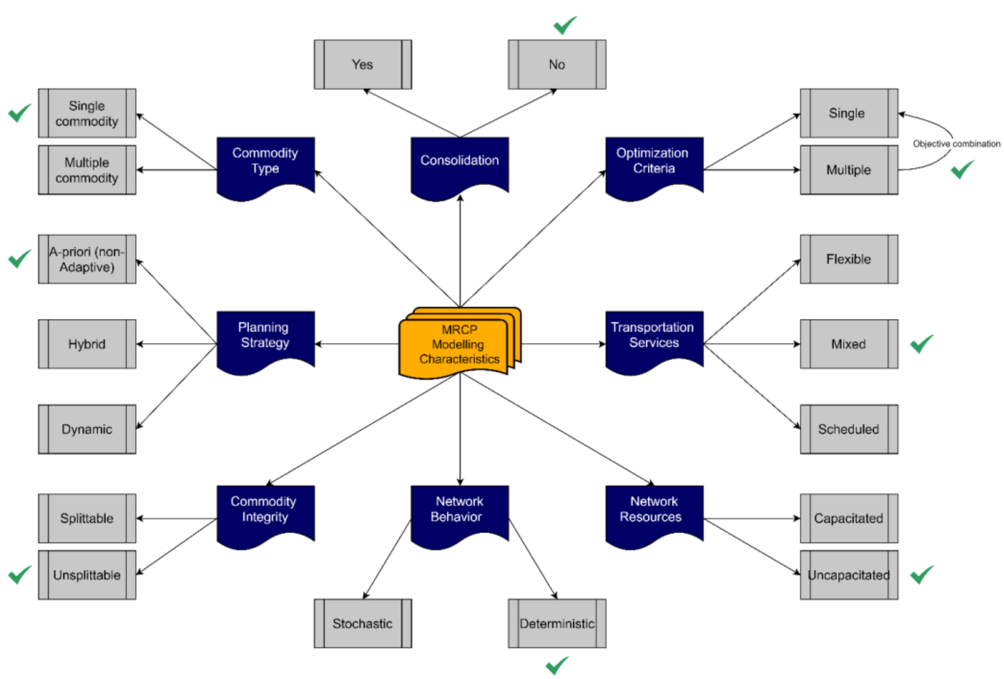


Figure 14: Classification scheme for the Multimodal Route Choice Problem based on network attributes. The scheme is inspired by the works of Sun, Lang and Wang (2015), Sun and Lin (2019), Fu (2000) and Chabini (1997). The checks indicate which attribute value is selected to model the SSLC routing problem.

In Section 3.1.4 we make modelling choices to describe our operational problem; these are marked by a check in Figure 14. Using this scheme, we summarize the studied MRCP publications by some relevant attributes in Table 11. Please note that all publications transport an unsplittable type of commodity and all assume a deterministic network.

Table 11: MRCP literature summary. Problems are classified according to their objective functions, the transportation services used, network resources, commodity type, whether consolidation is allowed, the type of approach to model time-dependency and the utilized solution methods. More detail about this classification in Figure 14.

Publication	Objectives	Transportation service	Network resources	Optimization object	Consolidation	Time-dependency	Solution methods	Case study
(Kaewfak, Ammarapala and Huynh 2021)	C, T, R	Mixed	Uncapacitated	Multiple	×	N	Zero-one goal programming	Thai coal transportation network (rail, sea, road)
(Koohathongsumrit and Meethom 2020)	C, T, R	Flexible	Uncapacitated	Multiple	×	N	TOPSIS	Vietnamese distribution network (rail, sea, road)
(Peng, Yong and Luo 2020)	C, T	Scheduled	Uncapacitated	Multiple	×	H	NSGA II	Fictive transportation network (rail, sea, road)
(Archetti and Peirano 2019)	C	Mixed	Capacitated	Single	✓	S	Swarm based intelligence	Italian air freight forwarder (air, road)
(Sun and Li 2019)	C, T	Mixed	Capacitated	Multiple	✓	N	Linear programming	Road-rail transportation of hazardous materials
(Mnif and Bouamama 2017a)	C, T	Scheduled	Capacitated	Multiple	×	H	Linear programming	Chinese fertilizer transportation network (road, rail, sea)
(Mnif and Bouamama 2017b)	C, T	Mixed	Capacitated	Single	×	S	Linear programming	Chinese fertilizer transportation network (road, rail, sea)
(Hrusovsky, et al. 2016)	C, T, E	Scheduled	Capacitated	Multiple	×	H	Hybrid (exact and simulation)	European distribution network (rail, road, sea)
(Chen and Lai 2015)	C, T	Mixed	Capacitated	Multiple	×	S	CPLEX	Taiwanese freight forwarders (road, rail)
(Sun and Lang 2015)	C, E	Mixed	Capacitated	Single	×	H	Linear programming	Chinese export business (rail, road)
(Kengpol and Tuammee 2014)	T, R, E	Flexible	Uncapacitated	Multiple	×	N	Zero-one goal programming	Distribution network between Thailand and Vietnam (road, rail, sea and air)
(Kumar, et al. 2014)	C, T	Scheduled	Capacitated	Multiple	×	H	Nested partition and TOPSIS	International air-based transportation network (air, road)

Objectives: Costs (C), Travel time (T), Number of transiting nodes (N), Waiting time at nodes (W), Risk (R), CO2 emissions (E). All objectives are minimized.
Consolidation: allowed (✓), not allowed (×).
Time-dependency: Hard time-windows (H), Soft time-windows (S), Graph-expansion (G), Not used (N).

Table 11 (continued)

Publication	Objectives	Transportation service	Network resources	Optimization object	Consolidation	Time-dependency	Solution methods	Case study
(Lei, et al. 2014)	C, T, R	Mixed	Capacitated	Single	×	S	Swarm intelligence algorithm	Fictive transportation network (road, rail, sea and air)
(Xiong and Wang 2014)	C, T	Mixed	Uncapacitated	Multiple	×	H	Taguchi genetic algorithm	Fictive transportation network (road, rail, sea)
(Dong, Li and Zheng 2013)	C, T	Mixed	Capacitated	Single	×	G	ACO heuristic	Simulated scenarios (road, rail, sea and air)
(Li, Negenborn and De Schutter 2013)	C, T	Flexible	Capacitated	Multiple	×	G	Linear programming	Fictive transportation network (road, rail, sea)
(Ayar and Yaman 2012)	C	Mixed	Capacitated	Multiple	✓	H	Linear programming	Third partly logistics distribution network (sea, road)
(Domuta, et al. 2012)	C, T	Scheduled	Capacitated	Multiple	✓	H	Martin's algorithm	Fictive transportation network (unspecified transportation modes)
(Li, et al. 2011)	C	Flexible	Capacitated	Single	×	S	Genetic algorithm	Intercity transportation network (unspecified modes)
(Ayed, et al. 2010)	C	Scheduled	Uncapacitated	Single	×	G	ACO, Dijkstra	-
(Zhang and Peng 2009)	C, T	Mixed	Capacitated	Single	×	H	NA	-
(Chang 2008)	C, T	Scheduled	Capacitated	Multiple	×	H	Lagrangian relaxation and reoptimization	Fictive transportation network (unspecified modes)
(Zografos and Androutsopoulos 2008)	T, N, W	Scheduled	Uncapacitated	Single	×	S	Label correcting algorithm	Athens urban transport network (road, rail)
(Song and Chen 2007)	C	Scheduled	Uncapacitated	Single	×	G	Label-correcting algorithm	Fictive transportation network (sea, air)

Objectives: Costs (C), Travel time (T), Number of transiting nodes (N), Waiting time at nodes (W), Risk (R). All objectives are minimized.
Consolidation: allowed (✓), not allowed (×).
Time windows: Hard (H), Soft (S), Graph-expansion (G), Not used (N).

3.1.4 Problem selection and adaptation

This section explains which problem formulation we take as basis to model SSLC'S routing problem, as well as the adaptations we make to fit it to our context. Within literature, the Multimodal Route Choice Problem seems to best resemble what is required at the company. In particular, we choose to take the publication by Lei et al. (2014) as benchmark. The authors propose a multi-objective MIP, with the objectives of minimizing transit time, operating costs and risk of transportation disruptions. Some model characteristics by Lei et al. (2014) are not applicable to our case, for which we modify or discard them. Above all, the authors' model assumes a capacitated network. Although in reality flights do indeed have capacity restrictions, most of the Express shipments handled by SSLC can be loaded on board due to their limited size. Furthermore, we use historical flight data to construct flight schedules as input to our model: from this data alone, it becomes hard to make realistic estimations of the capacity limitations. Therefore, we assume the network is uncapacitated. Additionally, the authors do not provide with a flow constraint to generate a path between origin and destination; we correct this mistake and add the constraint to our model. Figure 14 summarizes our modelling choices for the MRCP: we optimize multiple (three) functions, model mixed transportation services, assume an uncapacitated network, with time-windows and deterministic arc weights, the transported commodities are single, cannot be consolidated nor split and with an a-priori routing strategy.

3.2 Solution approach

3.2.1 Solution approaches for multi-objective optimization problems

This section discusses possible approaches to solve the selected problem type. As anticipated, our MRCP deals with multiple objectives, making it a multi-objective problem (MOP). When dealing with such problems, it is very uncommon that a solution exists such that it is optimal for all objectives, as these might be conflicting. To deal with this, two approaches can be taken: either to reduce the MOP to a single-objective problem, hence with a single solution, or to consider all objectives simultaneously, leading to multiple possible solutions. In the former case, we use scalarization or criterion-based methods to aggregate the optimization objectives; in the latter case, we either use indicator-based methods or Pareto optimization. A solution is Pareto optimal when it cannot be improved on any objective function value without deteriorating either of the remaining ones. Furthermore, we say a solution Pareto dominates another when it improves the other solution on at least one objective (Talbi 2009). Hence, multiple Pareto optimal solutions may coexist; which one is preferred depends on the decision maker's (DM) preference. The Pareto front consists of a graphical representation of all non-dominated (thus optimal) Pareto solutions. Indicator-based methods differ from Pareto optimization in the fact they use key performance indicator values to rank solutions rather than using the concept of dominance (Talbi 2009). When applying scalarization, several methods exist. Here we briefly explain the most common ones according to Talbi (2009):

- Weighted sum: objectives are assigned a weight (the weights add up to one, and reflect the relative preference on the objectives by the DM). To avoid disproportionalities, objectives with different units of measurement are normalized by their upper bound values. Then, a single objective is obtained as weighted sum of the original objectives and the problem is solved by either exact methods or approximations, with one optimal solution.
- ε-constraint: this method solves the problem by optimizing one objective while keeping the others as constraint (using a threshold value ε). The ε-value reflects the DM's preferences; the major advantage of this approach is that its value is not necessarily needed upfront to solve the problem.
- Goal programming: in this approach, the decision maker pre-sets target goals for each objective; the solving program then finds a solution that minimizes the deviation from such goals. This deviation can be weighted to model the relative importance of the objectives (Kengpol and Tuammee 2014).

Finally, we have criterion-based methods. Within this category, the lexicographic method reduces a MOP with n objectives into a n-staged single-objective problem. It does so by solving the problem for each separate objective sequentially, following a decreasing priority order. The most important objective is treated first; then, the problem is solved for the following one by adding an additional constraint which ensures the new solution does not deteriorate the obtained value of the previously treated objective. The advantages of this method are its simplicity and the fact that objectives do not need to be normalized; the major disadvantage is that n problems need to be solved, requiring additional computational time (Isermann 1982).

Which method should be selected depends on the DM's preferences with regards to the functions being optimized. According to Talbi (2009) there are three possible moments when a DM can express its preference relative to the optimization objectives. A priori methods assume the DM's priorities are known beforehand, hence there is enough knowledge in place to rank the problem's objectives following some utility function and then solve it. When this is the case, it is preferrable to reduce the problem to a single objective with either scalarization or criterion-based methods, as optimizing for just one objective is simpler (Censor 1977). Generally, it is unlikely the DM knows beforehand which option is preferred. In the latter case, a posteriori methods can be employed in order to approximate[3] the Pareto front. The DM can then examine it and select either solution. Scalar methods like weighted sum and ε-constraint can still be used to approximate the Pareto front: by altering the weights or ε-constraint values one generates multiple solutions, approximating the Pareto front. The downsize is that typically, scalar methods only provide with limited regions of the front (Amine 2019). Typically, Pareto optimization is then used. Interactive approaches are a way-in-between where the DM progressively expresses its preferences as the front is approximated (Talbi 2009).

[3] Due to the complex nature of most MOPs, it is difficult to obtain the complete Pareto set, for which the obtained front is an approximation (Amine 2019).

To select the most appropriate method, we discuss the solution requirements with the company. Starting with the number of solutions, we observe that booking agents would require having multiple alternatives. This is motivated by the fact that distinct customers have differing priorities. Agents should, for example, be able to offer routes minimizing risk for the risk-averse customer, while offering the fastest route to the one valuing speed of delivery. At the same time, other customers have no specific priorities and rather prefer having a balanced option, for which solving the MRCP for the three objectives separately would not be exhaustive enough. We argue that presenting too many options would not be beneficial either. As discussed with several stakeholders, presenting the complete Pareto front to agents (or to the customer) to select a solution would only cause confusion. This evaluation is supported by the work of Votov and Mettien (2008), who describe the latter as being the major pitfall of a posteriori decision making. Following a goal-programming approach in this context is rather difficult, as each transportation order may vary on multiple routing components, making it therefore hard for agents to estimate target levels for each objective. Furthermore, agents are required to offer a route in short time upon receiving the order, meaning the process should take the least effort possible by the decision maker. This consideration is valid enough to rule out interactive and a posteriori approaches. In fact, the company's management stressed the necessity of providing some guidance to the agents, by indicating which solutions would be the most preferred. Given we have already obtained such preference information using the AHP in Chapter 2, we argue it is appropriate to use it. In short, multiple solutions are required (but not too many), at least one leaning more towards either objective, and one balancing alternative; furthermore, the route computation must be done quickly, thereby should require the least possible input from the decision maker. With these considerations, we choose to use the linear weighted method, by which we reduce our MOP to a single-objective problem and solve it for four weight combinations: one purely risk-, travel time- or cost-centric and one balanced, using the obtained AHP weights.

3.2.2 Metaheuristics and hyperheuristics

To solve our problem we can choose between exact and approximate solution types. Exact methods enable us to obtain an optimal solution with certainty or at least to return the so-called optimality gap, which is a measure of distance between the obtained solution and the global optimum. Exact methods have the disadvantage of being computationally more demanding, meaning they run slower on complex or big-sized problem instances. As the word suggests, approximation methods yield solutions that only approximate the optimal value: however, their main advantage is the increased computational efficiency which makes them suited for bigger and more complex problem instances (Talbi 2009). The MRCP belongs to the NP-hard class of problems (Ayar and Yaman 2012). Moreover, ours has a considerable size already if one considers the European hubs and flights only. Given the latter two considerations and the fact the solution needs to run fast and often on a daily basis, we argue computational efficiency is more important in this case than the solution's quality. Therefore, we disregard exact solution methods and discuss only approximations.

Approximate methods can be divided into approximation algorithms and heuristics. The former provide solutions whose quality and computational time are provable, in contrast to the latter category. A shortcoming, however, of approximation algorithms is that they are often ineffective in realistic-sized problems (Talbi 2009); for this reason we disregard them. Within the heuristics group, we are particularly interested in metaheuristics: these are problem-agnostic solving methods that, due to their effectiveness and ease of adaptation to the problem at hand, are particularly suited for solving complex problems. Talbi (2009) classifies them in single-solution (S-) and population (P-) metaheuristics. S-metaheuristics start with one initial solution and then try to explore neighbor solutions to improve it. A neighbor is defined as a solution that can be generated by altering a decision variable value contained within a starting solution, typically with a move or swap operation. P-metaheuristics start with a set of solutions and develop them to reach a point as near to the global optimum as possible. In all types of metaheuristics, a central aspect is the tradeoff between intensification (hence directed search towards the best performing solutions) and diversification (the exploration of unknown, potentially promising regions of the Pareto set). When too much emphasis is put on the former, one may end up in a local optimum; this is a solution which seems of good quality but could be potentially dominated by an unexplored solution. On the other hand, when too much diversification is performed, the heuristic might spend too much time in solution areas which are not promising. Generally, P-metaheuristics lean more towards diversification strategies, whereas S-metaheuristics are more of intensification nature (Talbi 2009).

P-metaheuristics

The most widely adopted P-metaheuristics are either evolutionary or swarm intelligence algorithms (Talbi 2009). Of the former group, Non-dominated Sorted Genetic Algorithm (NSGA II), Strength Pareto Evolutionary Algorithm (SPEA2) and Indicator-Based Evolutionary Algorithm (IBEA) are the most popular, whereas Ant Colony Optimization (ACO) and Particle Swarm Optimization (PSO) are the most represented for the latter group. Lei et al. (2014) use a hybrid approach of the latter two algorithms to solve the MRCP. Generally PSO is a simple approach, which requires less parameters compared to the remainder of P-heuristics. Its drawbacks are that it gets stuck easily in local optima and is not particularly suited for complex combinatorial problems (Lei, et al. 2014). On the other hand, ACO is more suited for combinatorial MOPs as it displays robust solutions due to its positive feedback mechanism. However, it generally suffers from a low running time (Lei, et al. 2014). NSGA II is arguably the most widely adopted MOP genetic algorithm. Genetic algorithms are inspired from the evolutionary theory (survival of the fittest): by combining two good performing “parent” solutions, it is assumed better “offspring” is generated; mutation operations are then performed on the offspring to enforce diversification (Talbi 2009). NSGA II, SPEA2 and IBEA all operate using these crossover and mutation operations. For conciseness, we do not describe their logic in detail; to the interested reader we recommend the works of Deb (2011) for the NSGA II, Hong et al. (2018) and Zitzler and Künzuli (2004) for IBEA, and Potti and Chinnasamy (2011) and Liu and Zhang (2019) for SPEA2.

Talbi (2009) points out that generally, S-metaheuristics are more suited for single-objective optimization, whereas P-metaheuristics are a better fit to Pareto optimization. In fact, P-based heuristics tend to run slower, as they delay convergence to a solution because they are focused on finding unexplored regions of the Pareto front within their initial iterations. This results in a more complete solution spectrum, however too much diversification at the start may lead to worse solutions than the ones found by S-heuristics. Jiang et al. (2021) illustrate this concept by comparing the performance of NSGA II with a Multi Objective Tabu Search (MOTS) on a Vehicle Routing Problem, showing favorable results for the latter, as it converges earlier and proves to be more efficient in exploring the solutions neighborhood. Given we reduce our problem to a single-objective variant and thereby do not require approximating the entire the Pareto set, we shift our attention to S-metaheuristics.

S-metaheuristics

Here we discuss the three most widely adopted S-metaheuristics: local search, simulated annealing and tabu search (Talbi 2009). Local search starts with an initial solution; at each iteration, a heuristic replaces the current solution by a neighbor that improves the objective function. The algorithm stops once all candidate neighbors are worse than the current solution. The main problem with basic local search is that it easily gets stuck at local optima. The Greedy Randomized Adaptive Search Procedure (GRASP) overcomes this issue by introducing a casual component. Greedy heuristics build solutions by adding elements that contribute best to the objective function. In GRASP, the next element is chosen randomly from a list of candidates, thereby introducing diversification. Although being simple and easily adaptable to any problem, GRASP has the major disadvantage of cycling: the same solution may be revisited several times, which is consequently computationally inefficient (Talbi 2009). Tabu search (TS) improves this by keeping a (tabu) list of already explored solutions: this way, it avoids accepting a solution that is kept on the list and forces the algorithm to explore further. At each iteration, the list is updated with a newly utilized move and the oldest one in the list is removed. The algorithm therefore accepts worse solutions to explore new solution spaces; the absolute best solution is stored for when the algorithm terminates (Talbi 2009). Overall, tabu search has proven to be very effective in diverse optimization problems (Marett and Wright 1996). However, its performance heavily depends on parameter settings (e.g., the length of tabu list, whether it should be static or dynamic, the stopping criterion, etc.) (Xia and Fu 2018). Simulated annealing (SA) also allows the degradation of a solution; this is done with decreasing probability. At the start of the algorithm, almost all solutions are accepted, including worsening ones; as it moves towards the maximum iterations this worse-acceptance probability decreases until only strictly improving solutions are accepted. SA differs from TS as it is a memoryless algorithm: information gathered during the search is not used (Talbi 2009).

Hyperheuristics

Besides metaheuristics, we may also consider the use of hyperheuristics. This class of solution methods chooses between several given heuristics at various decision points in an optimization problem. Well-known hyperheuristics are Large Neighborhood Search (LNS) and Adaptive Large Neighborhood Search (ALNS), which combine the use of destroy and repair heuristics to explore different neighborhoods starting from an initial solution. Destroy heuristics alter a given solution’s attributes, making it infeasible. Repair heuristics then alter the solution making it feasible again, thereby repairing it (Pisinger and Ropke 2010). Another well-known method is Variable Neighborhood Search (VNS). Instead of initially destroying solutions, it alternates different heuristic operators to switch between neighborhood structures. A common denominator of these methods is randomness: for all three, the evaluation and selection of neighbor solutions happens stochastically (Hansen, N. and Brimberg 2010). The main advantage of hyperheuristics over metaheuristics is their capability of exploring even larger solution spaces, making them particularly suited for both large and complex mathematical problems. A downsize, however, is their increased complexity (Pirim, Bayraktar and Eksioglu 2008).

3.2.3 Solution approach

Tabu search

Given the considerations made above, we argue that both TS and SA are valid metaheuristics to solve our problem. We choose to use the former; here we describe the method in more detail and motivate the reason for our choice.

Like local search algorithms, tabu search requires an initial solution at the start; the solution is then first altered using a neighborhood operator. The latter can be defined as an operation which changes a solution's attribute, such that a new solution results from it. Common operator moves are swap (which swaps a solution's attribute with another one), add (which adds a new attribute to the solution) or delete (which deletes an existing attribute) (Glover and Taillard 1993). Within tabu search, the neighborhood is explored in a deterministic way. From the generated neighborhood, the solution yielding the best objective value is selected and compared to the current best one found so far. If it improves the current best, the found solution is registered as new best. Next, the same heuristic operation is applied to it and a new neighborhood is generated. This is done until a certain stopping criterion is met. A potential danger of deterministic neighborhood exploration is cycling, which entails keeping searching for a better candidate using the same search direction. To avoid the latter, a tabu list is kept. This list keeps track of moves (Glover and Taillard 1993) or entire solutions (Talbi 2009) of past selected neighbors. Solutions or moves in the list are tabu, meaning that a neighbor solution in the list cannot be selected as new candidate best solution. Whenever storing moves, an exception to this rule is the so-called aspiration criterion: for example, if a move is on the tabu list but it improves the best-found value so far, then this solution can still be selected. At each iteration, the tabu list is updated: the newly selected neighbor (or move) is added to the list and, generally, the oldest registered one is removed. Different rules to regulate the list's length can be found in literature. For example, a list may have a static, maximal length or a dynamic length; the latter one is then dependent on the occurrence of improving solutions found (Talbi 2009).

From the works examined in Table 11 we infer that this method has not been used yet to solve the MRCP. Nevertheless, tabu search has proven to be one of the best performing solution methods on a large scale of combinatorial optimization problems. Pirim et al. (2008) mention, among others, it has been successfully implemented to solve notorious NP-hard problems like job-scheduling, vehicle routing and capacitated arc routing problems. Teng et al. (2003) compare TS and SA on the vehicle routing problem with stochastic demands, with results in favor of the former. Particularly, their results show that the superiority of TS increases with the problem size. Arostegui et al. (2006) confirm this observation when comparing the performance of tabu search, simulated annealing and genetic algorithm on the facility location problem.

Overall, TS tends to be faster than SA as it proceeds more aggressively towards an optimum. Indeed, TS spends less time in regions with unpromising solutions than SA, on the premise that at each iteration it makes the best move available (Glover 1990). Furthermore, SA may not be stopped at any desired moment (Michiels, Aarts and Korst 2007) whereas TS can. This allows eventually to stop the algorithm once a feasible solution is found, which might turn advantageous for problems of very large size and with a solving time limit. In our problem setting, we argue that finding a good solution in reasonable time is preferred over finding an excellent one in more time; to this end, TS appears to be more suited.

Besides the running time, SA poses more challenges in terms of parameter tuning, as the number and variations of settings are greater than for TS and proper tuning would require some prior knowledge of the Pareto curve (Amine 2019), which we do not have. Finally, TS explores the solution's neighborhood deterministically: this makes the algorithm feasible to reproduce results on a particular problem instance (Brandão and Eglese 2008). In our case, we solve the same problem instances using different weights configurations for the considered objective functions (more detail in Chapter 4). We are therefore also interested in analyzing how the relative importance of speed, reliability and cost influences the model's routing choices. Therefore, to make better informed inferences, it is important to limit the effect of randomness on the obtained outcomes.

Finally, we acknowledge that hyperheuristics methods also present valid alternatives to solve the problem formulation. Nevertheless, the usage of a metaheuristic does not necessarily exclude the other. Indeed, several works exist showing effective implementations of hybrid methods. Particularly interesting are the works of Belhaiza et al. (2013) and Archetti et al. (2005), which both combine TS with VNS into a multiple neighborhoods tabu search (MNTS) for solving respectively a capacitated arc routing problem and team orienteering problem. Combining different neighborhood operators can be advantageous for our problem setting, as we discuss in Section 4.3 in detail. Therefore, we additionally use a MNTS method.

Simheuristics

The chosen model and solution method assume a deterministic problem setting. As we discuss in Section 3.1.2 , the real-life routing problem is subject to uncertainty. Due to this uncertainty, solutions generated may be sensitive to slight changes in the model parameters, diminishing their overall robustness. Simheuristics comprise a combination of metaheuristics and simulation approaches, to include stochasticity in the solution generation without overcomplicating the model formulation (Juan, et al. 2015). Within the simheuristics framework, stochastic problems are reduced to their deterministic counterpart, by taking the estimates of parameters subject to uncertainty. Next, a metaheuristic is run to find a set of solutions. For each solution, a reduced number of simulation iterations is run to evaluate its robustness. Parameters are sampled by their known probability distributions and used to evaluate the objective cost function of the solution. Based on the results, the solutions are re-ranked, and a reliability analysis is carried out to show the stability of each instance given parameter variability. One major advantage of this approach is that not only the quality of a solution is evaluated, but also the risk it poses when subject to variation. Furthermore, due to its relative simplicity, simheuristics provides an interesting alternative to more complex methods like dynamic or fuzzy programming, as it is capable to provide solutions of good quality to real-sized problems in reasonable computing times (Juan, et al. 2015). Since we choose to solve the problem with a metaheuristic, using a simheuristics approach is a natural step to account for its stochastic nature; hence we choose to adopt a tabu search – simheuristics solution approach.

3.3 Conclusion

In this chapter, we first answer the question: "Which routing optimization problem fits best to the context of SSLC?". To formulate the problem, we choose to model it as a Multimodal Route Choice Problem following the formulation presented by Lei et al. (2014). Its objective consists of finding an optimal sequence of edges to traverse within a graph, which represents the transportation network at hand, and selecting the modality at each edge traversal, as to minimize the travel time, costs and risk of disruption. Furthermore, we choose to use time-windows to model scheduled services and assume a deterministic, static arc behavior.

Next, we answer the question: "What is proposed in literature to solve the SSLC routing problem?". First, to deal with the different objectives, we use a scalar weighted method; we hence weigh the objectives and combine them into one according to different utility functions. To solve the problem, we choose to implement a tabu search metaheuristic. In addition, given the stochastic nature of the problem, we incorporate simulation to increase the robustness of the generated solutions. Our chosen approach yields two major academic contributions to the field of Operations Research. First, Lei et al. (2014) use parameters to actively model risk of visiting node sequences by either transportation mode; yet they do not provide with a method to estimate such risk factors. To the best of our knowledge, there is no literature providing with such a framework applied to this routing context. Therefore, we contribute to the knowledge status quo by researching how such risk parameters can be properly elicited. Secondly, we were not able to find tabu search-based solutions to the MRCP. Our second contribution is to assess the effectiveness of this solution method to this class of problems.

Chapter 4 – Solution design

This chapter answers the research question: “How should the solution approach be designed?”. As discussed in Section 3.1.4 , we alter the MRCP formulation by Lei et al. (2014) as to adapt it to our own context. Section 4.1 starts this process by describing the problem and its scope, then proceeds on stating the solution requirements and its assumptions. Based on this information, Section 4.2 presents the model’s mathematical formulation. Section 4.3 discusses the tabu search metaheuristic used to solve our MRCP. Section 4.4 formulates our setup to test the model. This entails describing which lanes of the ones analyzed in Section 2.3.2 we select as test sample and how relevant model parameters are estimated from the available data. Finally, Section 4.5 describes the simheuristic procedure we use to include stochasticity into the solution approach.

4.1 Problem formulation

4.1.1 Problem description

The problem at hand consists of finding a route between a given pickup and delivery point, starting at an indicated time of shipment’s availability. This process implicates making two decisions: choosing the sequence of airports visited between the pickup and delivery, and which carrier is used for each transportation leg. These two decisions should minimize the total transportation time, total transportation costs and overall risk of incurring delays. The network consists of $n \in N$ nodes; we distinguish here between the first- and last- mile locations (pickup and delivery points, $p, d \in N$) and airports ($a \in A \subset N$). Moreover, we also indicate the first and last airports visited, respectively the origin and ending airports ($o, e \in A$). As discussed in Chapter 2, the choice of these two influences the number and types of available flight options and therefore the overall quality of a route; furthermore, choosing the best start- and end-airports is crucial as additional operations like export/import preparation and customs clearance are performed here. Carriers for transportations between nodes ($k \in K$) are subdivided in vehicles ($v \in V \subset K$) and flights ($f \in F \subset K$). The former can be deployed both for transportation between pickup/delivery and airports and inter-airport transportation, the latter only between airports. Indeed, we include the possibility of using vehicles for transportation between airports, as to observe whether this could be interesting for SSLC to deploy on certain routes. To model time, we use hours over a predefined time horizon ($t \in T$).

The travel time, cost and risk incurred on a route come from three sources: the transportation legs between nodes, the service times at airports and, eventually, customs clearance operations. Therefore, we use one objective-specific parameter for each of these three components. For transportation legs, we indicate t_{ij}^k, c_{ij}^k and r_{ij}^k as travel time, cost and risk incurred by travelling between nodes i and j with carrier k. Similarly, parameters s_i, c_i, and r_i indicate the service time, cost and risk of transiting at node i. Besides these three parameters, we also account for the time needed for performing export operations at the origin airport o and import operations at the ending airport e. These include all activities necessary to ensure a shipment is ready to leave and enter a country (including packing, cross-examining documentation, and securing the shipment), and thereby require longer than transit operations, which are mainly focused with tail2tail transportation. We indicate the time needed for either export/import operations at airport i with parameter $exim_i$. Finally, cs_a, cc_a and cr_a indicate the time, cost and risk of performing customs clearance at airport a. Please note the latter is relevant only at the origin and destination airports, as we assume customs clearance can only be performed there, and whenever a shipment needs clearance (which we indicate using the Boolean parameter cu). By export we indicate clearance performed at the origin airport, while import is done at the ending one.

When selecting nodes and edges, we need to account for time constraints. In their formulation, Lei et al. (2014) do not account specifically for scheduled transportation between nodes; here we choose to model this aspect using cut-off times, following the work by Sun and Lang (2015). With dep_{ij}^k we indicate the cut-off time by which the shipment needs to be ready for loading at the gate. If the shipment arrives past that time, it cannot board its flight. Vehicle transportation is flexible, therefore we set their cut-off times always equal to the value the shipment is ready for departure. Besides transportation, we also need to account for office hours for airport operations and customs clearance. We model them using time windows: $[opsopen_a; opsclosed_a]$ model the opening and closing time of general airport services at airport a. Similarly, $[customsopen_a; customsclosed_a]$ model the opening and closing time of the customs office at airport a. If a shipment arrives outside these time-windows, it must wait until the next day to be serviced. We notice that some airports like major hubs operate 24/7 and thus can always service a shipment; in this case, we do not model time windows. The choice of which carrier to operate on a certain leg influences two variables. The first one is the arrival time at each node, calculated as the sum of the cut-off time for departure from its starting node (or shipment's availability time in case of flexible transportation) and the travel time to the destination node: $tarr_j^k = dep_{ij}^k + tij^k$. On its turn, the arrival time determines which transportation options are available for the following transportation leg. Additionally, if the shipment arrives at the gate for loading before the scheduled cut-off time, it also incurs waiting time w_i^k.

Figure 15 graphically summarizes how we choose to model the time component of our problem. At the origin and ending airports, the export/import service time $exim_i$ is incurred, which involve checking the shipment's packaging, documentation and security status. If needed, shipments need also to be customs cleared, incurring service time cs_i. Finally, if all these operations are finished before the scheduled cut-off time dep_{ij}^k, the shipment incurs waiting time w_i^k. When arrived and offloaded at its transiting airport(s), the shipment needs to be transported to its next flight/vehicle, incurring transit time s_i. Again, if the shipment is ready ahead of its scheduled departure time, it incurs waiting time. For transportation legs, t_{ij}^k models the time between the shipment's departure cut-off time (or availability time in case of vehicle transportation) and includes operations like on- and off- board loading and flight (or driving) time.

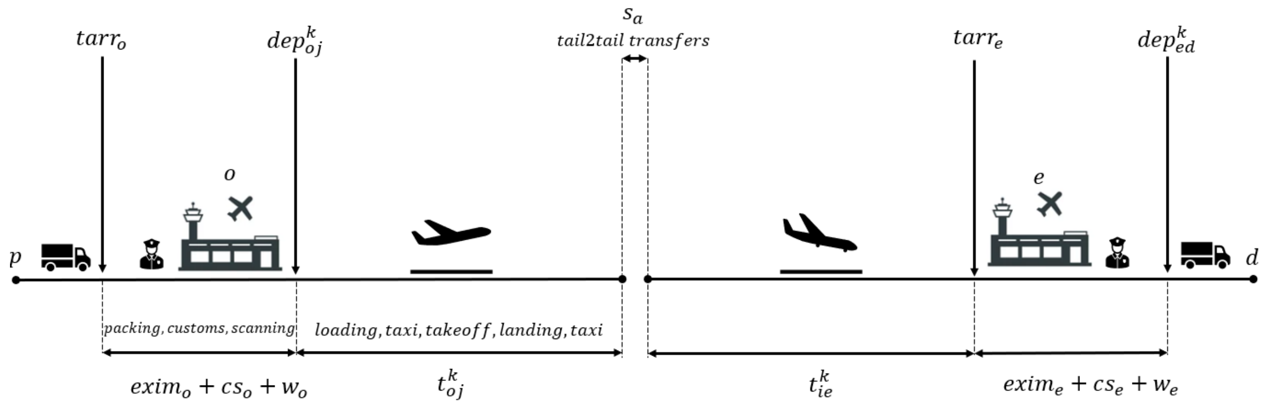


Figure 15: Schematic representation of time in the proposed model. Operations at the airport generally include tail2tail transportation, security scanning and eventually customs. The flight's travel time includes loading on and off board, and flight time.

As described in Chapter 1, when an agent selects a route, it provides the customer with an estimated delivery time. Chapter 2 discusses that most shipments deviates from the agreed delivery time: orders arrive either too early, or too late. In the same chapter we illustrate that this affects (negatively) performance on other KPIs, and although it does not yield any direct consequence like a penalty cost, such development may deteriorate a business relation on the long run. Following the model by Lei et al. (2014), we propose to model a soft time-window ($[t_{early}; t_{late}]$) for delivery at the final node d: we allow delivering outside this time boundaries, but at a cost of penalty tp_{early} for early or tp_{late} for late delivery. In Section 4.4 we describe in more detail how we set the boundaries and penalties.

We conclude this section by introducing the decision variables. The main decision variable x^k_{ij} indicates whether carrier k is selected for transportation between node i and node j. Variable y_i indicates whether node i is visited and variable z_i whether airport i is selected either as origin or destination airport. These two depend on the values of x^k_{ij} and are thereby not strictly necessary for the model formulation; however, they ease the readability of its mathematical formulation and overall understandability. For the same reason, we also use other dependent variables: $tarr^k_i$ indicates the arrival time at node i using carrier k, w^k_i stands for the waiting time at node i before transportation by carrier k, and finally variables $early$ and $late$ indicate the eventual deviation of final delivery from the earliest and latest agreed times $[t_{early}; t_{late}]$.

4.1.2 Problem scope

This section outlines the scope of the model. The problem formulation focuses on airfreight transportation, whereas SSLC also offers rail- and truck-based solutions. As discussed throughout the introduction to this research, the reason for this is that air route calculation is by far the most complex; furthermore, smoothing this process is important as it is the best sold transportation product by the company. Nevertheless, we argue the chosen model has the potential of being easily extendible to further types of transportation modality. The route calculation we propose includes the choice of carriers operating each transportation leg, as these are of major influence for the route's performance overall. However, we could also choose among multiple partners for the same type of ground operations at the airport. Despite this, we opt to avoid this level of granularity in our model and assume one general partner performs ground operations at each airport, and one broker for customs clearance. One key consideration backs this modelling choice: the context analysis in Chapter 2 shows that the routes performance at transiting airports does not vary significantly when comparing different operating partners. Therefore, including this aspect would add unnecessary complexity to the model and thus be of limited benefit. Finally, the model itself should be able to sustain calculations for the entire logistic network of the company. Implementing such solution would however exceed the time and workload available for this research: instead, we test it on a reduced instance of the original problem, taking a selection of routes as pilot. We address this in more detail in Section 4.4 .

4.1.3 Model requirements

This section lists the model requirements. Since we build a solution based on the existing routing system, most are overlapping with the validation requirements stated in Section 2.1.2 , although we make a few exceptions.

- Outbound flights from the chosen origin airport should have scheduled departure time within 24 hours from the shipment's availability time at the pickup location. This requirement originates from the API request described in Section 2.1.2 . Nevertheless, we acknowledge that, mathematically, this requirement might affect results negatively, as it narrows the solution space considerably. In Chapter 5 we investigate the difference between the solution's quality when enforcing and relaxing this constraint.
- Only contracted partners can be selected for operating a route. Given we generate options to input to the model based on historical data, this requirement is implicitly already satisfied.
- If a shipment needs to be customs cleared, then it must be cleared for export and for import, meaning the chosen airports of origin and destination should have the capability to process it.
- The office hours at airports, customs brokers as well as scheduled departure times of flights are hard time constraints. This means that a shipment cannot be serviced outside office hours, nor in case it arrives at a station on time but the total time to service it exceeds the preset time boundaries.

Additionally, we choose to relax the below requirements, which originally belong to the validation steps described in Section 2.1.2 :

- The first hard requirement we drop is the maximum number of connections allowed. Originally, this was set to three. The reason behind this is that the company considers routes with more than three air transits too susceptible to disruptions. However, since we actively take risk factors into account into our model, we are interested in seeing whether such assumption is true. It might be that routes with four transits would still yield an acceptable risk component, offering more options.
- As anticipated in the previous section, we model fictive vehicle options to see whether they could be an interesting option for inter-airport transportation.

- We drop the requirement which prohibits combining different airlines on the same route, as we are interested in verifying whether this could benefit a route's performance.
- Finally, as mentioned in Section 3.1.4 , mostly there is sufficient capacity on board to transport Express shipments. Therefore, we do not take capacity aspects into account into our model. This would not be possible anyway, since we use historical bookings data to construct flight schedules, and it is not possible to attain any information on on-board capacity from it.

4.1.4 Assumptions

Before proceeding to the formal, mathematical formulation of the problem, we discuss our modelling assumptions. Some have overlap with topics discussed in Chapter 3.

- For the reasons already stated in the previous section, we assume unlimited capacity for shipments on-board of each carrier.
- The model calculates routes per individual shipment (i.e., single commodity); within this calculation, we do not consider aggregation.
- We do not discriminate for commodity type, and hence assume goods can be transported on both freighter and passenger flights. For the same reason, we do not consider partner-specific agreements which restrict which airlines are allowed to transport certain good types.
- The NS&P team can close flights based on daily information on capacity or other aspects, like strikes. Since we do not have this type of information, nor an indication of how often flights are closed and on which lanes, we assume all flights provided are open.
- We assume only one airport handler and one customs broker can be selected for airport handling operations and customs clearance at airports; more detail in Section 4.1.2 .
- Similarly, station capabilities are guaranteed: we assume at each node at least one partner is in place and can service the shipment. Please note that this is not true for customs operations, as not all airports have the facilities to clear shipments.
- Customs clearance is only performed in the most standard way, namely export being done at the origin airport, and import and the destination airport. Special customs operations like T1[4] are excluded. Also, we neglect customer-specific agreements, which specify in which countries SSLC is allowed to clear goods for each individual customer. Although we acknowledge it would be interesting to integrate this aspect into our model, we choose to assume clearance is possible at the available airports regardless of the country, for the sake of simplicity.
- In the model, we do not differentiate between the service times for export and import at the airports, and model them by one general parameter $exim_i$. In reality, SSLC makes a distinction between them, mainly because of agreements they have with their operating partners (which offer different times based on the type of operation performed). From the parameter

[4] T1 customs implicate that a shipment can be cleared at the consignee instead of at the destination airport. T1 status shipments can enter a country of destination under condition and are cleared prior to being the delivered at their final destination.

estimation analysis (Section 4.4), however, we observe the time values for performing export/import do not differ significantly, allowing us to simplify them into a single parameter.

- Similarly, we do not distinguish between export and import clearance times and generalize them as customs clearance at airport a.
- When customs clearance needs to be performed, we assume the shipment is prepared for export first and then cleared at the origin airport o, and is first cleared and then prepared for import at the ending airport e. This has implications for the order in which the shipment is serviced. For example, if the shipment arrives at its origin o outside office hours for export/import, we assume it cannot be customs cleared, even if the arrival time is within the office hours of the customs office.
- Pickup and delivery can be performed at any time; hence we do not consider office hours. In reality, most of the pickup and delivery points are either offices or warehouses, which do have opening hours. However, we do not have access to this type of information for most of the customers. Normally, prior to booking a route, agents discuss pickup and delivery times with the customer and then amend the input in the system to find matching routes.
- We assume a static deterministic model, although we know its parameters behave stochastically and dynamically. To account for stochasticity, we later use a simheuristics approach, as discussed in Section 3.2.2 .
- We assume a pickup and delivery points are always given. There are cases with airport2door, door2airport or airport2airport shipments, which simplify the problem as the required origin and/or destination airports are already given. However, we are more interested in keeping that decision into our experimental setup.

4.2 Mathematical model

The information discussed in Section 4.1 is summarized here in the mathematical formulation of the model. In this order, we describe its indices and sets, parameters, variables, objective functions and constraints.

Sets and indices

Table 12: Description of the sets and indices used in the mathematical model.

Set name	Related indices	Description
$\boldsymbol{N}$	i, j: node p: pickup node d: delivery node	Set of all locations within the network, including customer locations (pickup and delivery) and airports.
$\boldsymbol{A}(\subset \boldsymbol{N})$	a: general airport o: origin airport e: ending airport	Set of all airports within the network, including the selected origin and ending ones.
$\boldsymbol{K}$	k, l: carrier	Set of all carriers within the network, including vehicles and scheduled flights.

$\boldsymbol{F}(\subset \boldsymbol{K})$	f: flight carrier	Set of all scheduled flights between airports.
$\boldsymbol{V}(\subset \boldsymbol{K})$	v: vehicle carrier	Set of all available vehicles, including first- and last-mile transports and road feeders.
$\boldsymbol{T}$	t: hour	Time horizon in hours.

Parameters

Table 13: Description of the parameters used in the mathematical model. Units of measurement are hours (time), euros (costs), normalized risk scale (risk).

Parameter name	Description
$\boldsymbol{t_{ij}^k, c_{ij}^k, r_{ij}^k}$	Time, cost and risk of traversing arc i, j using carrier k respectively.
$\boldsymbol{s_i, c_i, r_i}$	(Transit) service time, cost and risk of transit at node i, respectively.
$\boldsymbol{cs_a, cc_a, cr_a}$	Service time, cost and risk of doing customs clearance at airport a, respectively.
$\boldsymbol{tarr_p}$	Availability time of shipment at pickup point p (in hours)
$\boldsymbol{exim_a}$	Export/import service time at airport a.
$\boldsymbol{cu}$	Boolean parameter: 1 if the shipment needs customs clearance, 0 otherwise.
$\boldsymbol{dep_{ij}^k}$	Scheduled cut-off/departing time of carrier k from node i to j
$[\boldsymbol{opsopen_a}; \boldsymbol{opsclosed_a}]$	Time window for general airport handling at airport a.
$[\boldsymbol{customsopen_a}; \boldsymbol{customsclosed_a}]$	Time window for customs clearance at airport a.
$[\boldsymbol{t_{early}}; \boldsymbol{t_{late}}]$	(Soft) time-window for delivery at delivery point d.
$\boldsymbol{tp_{early}, tp_{late}}$	Penalties for respectively early and late delivery.
$\boldsymbol{t_{max}, c_{\max}, r_{\max}}$	Maximum parameter values of respectively time, cost and risk.
$\boldsymbol{\alpha, \beta, \gamma}$	Weight factors for respectively objective functions Z_1, Z_2, Z_3
$\boldsymbol{M}$	Large number used to enforce constraints.

Variables

Table 14: Model variables, distinguished between decision and auxiliary variables.

Variable name	Description
$\boldsymbol{x_{ij}^k}$	1 if arc i, j is traversed with carrier k, 0 otherwise.
$\boldsymbol{y_i}$	1 if node i is visited, 0 otherwise.
$\boldsymbol{z_i}$	1 if node i is chosen either as origin airport o or ending airport e.

$\boldsymbol{tarr_i^k}$	Arrival time at node i using carrier k.
$\boldsymbol{w_i^k}$	Waiting time at node i prior transportation with carrier k.
$\boldsymbol{early}$	Nonnegative time difference of delivery with earliest allowed time t_{early}.
$\boldsymbol{late}$	Nonnegative time difference of delivery with latest allowed time t_{late}.
$\boldsymbol{r_1}$	Value is 1 if the shipment arrives after the earliest allowed time t_{early}, 0 otherwise.
$\boldsymbol{r_2}$	Value is 1 if the shipment arrives before the latest allowed time t_{late}, 0 otherwise.
$\boldsymbol{u_1}$	Positive integer indicating the day number in constraint 11.
$\boldsymbol{u_2}$	Positive integer indicating the day number in constraint 12.

Objective functions

$$\min Z_1 = \sum_{k\in K}\sum_{i\in N}\sum_{j\in N} x_{ij}^k t_{ij}^k + \sum_{i\in N-\{o,e\}}\sum_{k\in K} y_i(s_i + w_i^k) + cu\sum_{i\in N} z_i cs_{i+} \sum_{i\in N}\sum_{k\in K} z_i\,(exim_i + w_i^k)$$

Z_1: Minimize the total sum of transit time incurred at each transportation leg, airport transit, export/import and customs clearance.

$$\min Z_2 = \sum_{k\in K}\sum_{i\in N}\sum_{j\in N} x_{ij}^k r_{ij}^k + \sum_{i\in N} y_i r_i + cu\sum_{i\in N} z_i cr_i$$

Z_2: Minimize the total sum of risk incurred at each transportation leg, airport transit and customs clearance.

$$\min Z_3 = \sum_{k\in K}\sum_{i\in N}\sum_{j\in N} x_{ij}^k c_{ij}^k + \sum_{i\in N} y_i c_i + cu\sum_{i\in N} z_i cc_i + early * tp_{early} + late * tp_{late}$$

Z_3: Minimize the total sum of costs incurred during transportation legs, airport transit, customs clearance and penalties by either early or late delivery.

$$\min Z_4 = \alpha * \frac{1}{t_{\max}} Z_1 + \beta * \frac{1}{r_{\max}} Z_2 + \gamma * \frac{1}{c_{\max}} Z_3$$

Z_4: Minimize the normalized linear weighted sum of travel time, cost and risk incurred.

Constraints

Formula

$$(1) \sum_{k\in K}\sum_{j\in N} x_{ij}^k - \sum_{k\in K}\sum_{j\in N\setminus\{i\}} x_{ji}^k = \begin{cases} 1\ for\ i = p \\ -1\ for\ i = d \\ 0\ otherwise \end{cases}$$

(2) $\sum_{k\in K}\sum_{j\in N} x_{ij}^{k} \leq 1\,, \qquad \forall i \in N\backslash\{d\}$

(3) $\sum_{k\in K}\sum_{i\in N} x_{ij}^{k} \leq y_j, \qquad \forall j \in N$

(4) $\begin{cases} z_i \geq \sum\limits_{k\in K} x_{pi}^{k} \\ z_j \geq \sum\limits_{k\in K} x_{jd}^{k} \end{cases}, \qquad i \neq j;\ \forall i,j,p,d \in N$

(5) $tarr_p + t_{po}^{v} - tarr_o^{v} \leq M\left(1 - x_{po}^{v}\right), \forall v\ \in V; p,o\ \in N$

(6) $dep_{ij}^{f} + t_{ij}^{f} - tarr_j^{f} \leq M\left(1 - x_{ij}^{f}\right), \qquad \forall i\ \in N, \forall j \in N - \{i\}, \forall f \in F$

(7) $tarr_i^{k} + (1 - z_i)s_i + z_i exim_i + cu\, z_i cs_i + t_{ij}^{v} - tarr_j^{v} \leq M\left(1 - x_{ij}^{v}\right),$
$\forall i\ \in N, \forall j \in N - \{i\,\}, \forall v \in V, \forall k \in K$

(8) $tarr_i^{k} \leq dep_{ij}^{f} - (1 - z_i)s_i - z_i exim_i - cu\, z_i cs_i + M\left(1 - x_{ij}^{f}\right),$
$\forall i\ \in N, \forall j \in N - \{i,d\}, \forall f \in F, \forall k \in K$

(9) $w_i^{k} \geq M\left(x_{ij}^{k} - 1\right) + dep_{ij}^{k} - tarr_i^{l} - (1 - z_i)s_i - z_i exim_i - cu\, z_i cs_i,$
$\forall i \in N, \forall j\ \in N - \{i\}, \forall k,l \in K$

$(10a)$ $early \geq t_{early} - tarr_d^{k}, \qquad \forall d \in N, \forall k \in K$

$(10b)$ $late \geq tarr_d^{k} - t_{late}, \qquad \forall d \in N, \forall k \in K$

(11) $x_{oj}^{k} dep_{oj}^{k} - tarr_p \leq 24, \qquad \forall j \in N, \forall k \in K$

(12) $\begin{cases} tarr_a^{k} \geq 24u_1 + opsopen_a \\ tarr_a^{k} \leq 24u_1 + opsclosed_a - (1 - z_a)s_a - z_a exim_a \end{cases}, \qquad \forall a \in A, \forall k \in K$

(13) $\begin{cases} tarr_a^{k} \geq 24u_2 + cu * customsopen_a - exim_a - M(1 - z_a) \\ tarr_a^{k} \leq 24u_2 + cu * customsclosed_a - exim_a - cs_a + M(1 - z_a) \end{cases}, \qquad \forall a \in A, \forall k \in K$

(14) $x_{ij}^{k} \in \{0, 1\}, \qquad \forall\ i,j\ \in N, \forall k \in K$

(15) $y_i \in \{0,1\}, \qquad \forall\ i\ \in N$

(16) $z_i \in \{0,1\}, \qquad \forall\ i\ \in N$

(17) $tarr_i^{k}, w_i^{k}, early, late \in \mathbb{Q}^{+}, \qquad \forall i\ \in N, \forall k \in K$

(18) $r_1, r_2 \in \{0,1\}$

(19) $u_1, u_2 \in \mathbb{N}$

Description

1) This is the flow constraint, ensuring a path between the starting node p and ending node d is constructed.
2) At most one outgoing carrier (edge) can be selected to traverse arc i, j.
3) If arc i, j is traversed, then node j is visited.
4) If node i is visited after the pickup node p, then it is the origin airport. If node j precedes the delivery node d on the route, then it is the ending airport.
5) For the first-mile leg, the arrival time at the origin o using vehicle v is equal to the shipment's availability time $tarr_p$ plus the drive time t_{po}^{v}.
6) For scheduled carriers (flights), the time of arrival at a node is equal to the departure time from the preceding node plus the travel time, if that arc is traversed and 0 otherwise.
7) For flexible carriers (vehicles), the time of arrival at a node is equal to the time the shipment is ready for transportation (hence the arrival time at the previous node, summed with the transit time or export/import time, and customs clearance, if performed), plus the transportation time to that node.
8) If a shipment arrives past the scheduled departure cut-off time of flight f, then the shipment cannot be transported on this flight.
9) The waiting time at a node is equal to the difference between the scheduled departure time from that node, minus the time at which the shipment is ready for transportation, if the node is visited. It is 0 otherwise.
10) If a shipment is delivered at destination d earlier than the allowed time t_{early}, the $early$ time is equal to the difference between the boundary and the time of arrival, it is 0 otherwise. Similarly, the $late$ time is equal to the difference between time of arrival and the maximum allowed t_{late} if it is delivered past that boundary, 0 otherwise.
11) The carrier k selected to depart from origin airport o should depart within 24 hours from the shipment's availability time at p (t = 0).
12) If the shipment arrives past the (daily) scheduled closing time of airport services or it needs servicing that falls outside office hours, it is serviced at opening time the next day.
13) If the shipment arrives past the (daily) scheduled closing time of the customs office, or it needs clearance that falls outside office hours, it is cleared at opening time the next day.

14-19) Sign constraints for the corresponding variables.

4.3 Tabu search

This section describes the tabu search metaheuristic. We first cover the heuristic employed to generate an initial solution in Section 4.3.1 . From this initial solution, the tabu search procedure generates a set of alternative solutions (called neighborhood) and selects the best non-tabu neighbor, which is compared with the best solution found so far. This procedure is repeated over several iterations, ending with the best-found solution as result. We address these steps in detail in Section 4.3.2 . Finally, we propose a hybrid metaheuristic which incorporates randomness and multiple heuristic operators together with tabu search, described in detail in Section 4.3.3 .

4.3.1 Initial solution

To come up with a solution, the tabu search algorithm needs an initial solution as input. From there, a heuristic operator is used to alter the initial solution, thereby generating a number of alternative (neighboring) solutions. The set of generated solutions is called neighborhood. From there, the best neighboring solution is iteratively selected to generate a new neighborhood (Talbi 2009). To generate an initial route, we choose to use the same logic as the current system's (described in Section 2.1.1). This approach enables us to ultimately check how often the best-found solution differs from the original one, and hence gives us an indication of whether our proposed approach is indeed superior to the current routing logic.

The algorithm steps are described by the pseudocode in Figure 16. The first step consists of finding the nearest airports to the pickup and delivery points (lines 2-3), hence the origin and destination o, e. Next, the shipment's arrival time at the origin airport is calculated as the sum of the shipment's availability time and the driving time t_{po}^k (lines 4-5). Given the arrival time at o, the shipment's ready time is then calculated using the service time $exim$ and, if needed, customs service time cs (line 5). This is done accounting for the airport's office hours. If the shipments arrives prior to opening, it must wait until the opening time and is then serviced. If it arrives past closing time or cannot be serviced on-time, it waits until the next day. This logic is summarized by the procedure *OfficeHoursOrigin,* which we describe in more detail in Appendix E. We use a similar logic for calculating the service times at transit airports and at the ending airport within the procedures *OfficeHoursTransit* and *OfficeHoursEnding*. Once the ready time at the origin is known, the algorithm iterates over all possible airport combinations, with a maximum of three transits, and selects the option which arrives the earliest at the ending airport e (lines 9-31). Finally, the arrival time at the delivery point is calculated as the sum of the shipment's ready time at e and the last-mile leg t_{ed}^k (line 33), then the relevant objectives are calculated, and the solution is returned (lines 33-34).

ALGORITHM 1: INITIALSOLUTION

1 ***Initialize*** *solution, airports[], carriers[], time, cost, risk, ready, readyT1, readyT2, readyT3, bestArrival, carrierOne, carrierTwo, carrierThree, carrierFour, T1, T2, T3*
2 *Find origin o and ending airport e such that travel time from p and d is minimized respectively*
3 $\rightarrow o = arg_j \min[t_{pj}^k], e = arg_i \min[t_{id}^k]$
4 *Calculate time of arrival at o, then calculate the ready time after export and customs clearance*
5 $\rightarrow tarr_o^k = tarr_p + t_{po}^k$, $ready = OfficeHoursOrigin(tarr_o^k, o, cu)$
6 *Update cost and risk values*
7 $\rightarrow cost = c_o + c_e + c_{po}^k + c_{ed}^k + cu * (cs_o + cs_e)$, $risk = r_o + r_e + r_{po}^k + r_{ed}^k + cu * (cr_o + cr_e)$
8 *Set departure for vehicle carrier v, initialize arrival time* $\rightarrow dep_{oe}^v = ready;\ arrival = M$
9 ***for*** *(all carriers k from o to e)* ***do***
10   *Find carrier k that minimizes arrival at e, with* $dep_{oe}^k \leq (24 + tarr_p)$ *and* $dep_{oe}^k \geq ready$
11     $\rightarrow bestArrival = dep_{oe}^k + t_{oe}^k;\ carrierOne = k$
12 ***end for***
13 ***for*** *(all nodes i reachable from o)* ***do***
14   *Set departure for vehicle carrier v to ready time at departing station* $\rightarrow dep_{oi}^v = ready$
15   ***for*** *(all carriers k from node o to i)* ***do***
16   *Calculate time of arrival at i, then calculate the readyT1 time after transit, with* $dep_{oi}^k \leq 24 + tarr_p$ *and* $dep_{oi}^k \geq ready$
17         $\rightarrow tarr_i^k = depo_{oi}^k + t_{oi}^k$, $readyT1 = OfficeHoursTransit(tarr_i^k, i)$
18   *Set departure for vehicle carrier v from i to e* $\rightarrow dep_{ie}^v = readyT1$
19   ***for*** *(all carriers l from node i to e)* ***do***
20   *Find carrier l that minimizes arrival at e, with* $dep_{ie}^k \geq readyT1$
21             $\rightarrow tarr_e^k = dep_{ie}^k + t_{ie}^k$
22   ***if*** $tarr_e^k < bestArrival$ ***then***
23                 $bestArrival = tarr_e^k$, *T1 = i, carrierOne = k, carrierTwo = l*
24   ***end if***
25   ***end for***
26   ***for*** *(all nodes j reachable from i)* ***do***
27   *set departure time for vehicle carrier from i to j to readyT1, for all carriers from i to j calculate arrival and ready time at j. For all carriers from j, calculate arrival time at ending airport a; if this is the best arrival time found, set the arrival time as new bestArrival and save carriers and transit stations (similar to steps 14-24.*
28   *Similarly to step 27 and 14-24, calculate arrival time at ending airport via each g reachable from j. If the arrival time is lower than the lowest found, set it as new bestArrival, save carriers carrierOne, carrierTwo, carrierThree and carrierFour and transit airports T1, T2, T3.*
29   ***end for***
30   ***end for***
31 ***end for***
32 *Set airports = [o, T1, T2, T3 e], carriers = [carrierOne, carrierTwo, carrierThree, carrierFour]*
33 *time = OfficeHoursEnding(arrival, e, cu), calculate risk and cost as sum of transit and carrier-specific parameters*
34 ***return*** *solution = [airports, carriers, [time, risk, cost]]*

Figure 16: Pseudocode of the initial solution generation. The algorithm follows the same steps as the ones of SSLC'S current routing system.

4.3.2 Main heuristic

This section describes the main tabu search procedure. In our implementation, we use two different memory storage methods: one keeping track of utilized moves as proposed by Glover and Taillard (1993) and one keeping track of entire visited solutions as proposed by Talbi (2009). We call these versions tabu search M1 and M2, respectively. For the former, we use an aspiration criterion, meaning tabu moves can still be accepted if the obtained solution improves the current best one. Furthermore, we use static tabu lists and stop the algorithm after a maximum number of iterations is reached.

```
TABU SEARCH
Initialize TabuList, tabuMaxLength, iter, maxIter, counter
Solution = InitialSolution, CurrentBest = Solution
while (iter ≤ maxIter) do
    Neighborhood = GenerateNeighborhood(Solution, p, d, tarr_p, cu, α, β, γ, counter)
    counter += 1
    BestNeighbor = ChooseBestNeighbor(Neighborhood, TabuList, currentBest)
    Solution = BestNeighbor
    if Solution > CurrentBest then
        CurrentBest = Solution
    end if
    if (length of TabuList > tabuMaxLength) then
        TabuList = TabuList – lastElement
    end if
    add Solution to TabuList, iter++
end while
return CurrentBest
```

Figure 17: Pseudocode of the utilized tabu search algorithm.

Figure 17 illustrates the pseudocode of the chosen tabu search algorithm. The algorithm follows the general logic we describe in Section 3.2.3 , therefore we omit a detailed description here. Figure 18 shows how we generate the neighborhood.

```
ALGORITHM 2: GENERATENEIGHBORHOOD
1   Input: currentSol, p, d, tarr_p, cu, α, β, γ, counter
2   Output: neighborhood
3   Initialize: neighborhood[], airports[], carriers[], objValues[],  balValue; set currairports = currentSol[0],
    currcarriers = currentSol[1]
4   Set airports that can be added swapped to current route → swapAirports = allAirports – currairports
5   if (counter > length of currairports) then reset counter, swap first airport
6       counter = 0
7   end if
8   airports = currairports – currairports[counter] (remove to-be-swapped airport)
9   for (all swappable airports i in swapAirports) do
10      insert airport i in route → airports = airports.insert(counter, swapAirports[i])
11      find optimal carriers in new route →ReoptimizeRoute(airports, p, d, tarr_p, cu, α, β, γ)
12      calculate objective values objValues[ ], calculate balanced score
        → balValue = α * objValues[time] + β * objValues[risk] + γ * objValues[cost]
13      add new route, carriers, objValues and balValue to neighborhood[ ]
14  end for
15  sort neighborhood by ascending balanced score (best to worst) objective value balValue
16  return neighborhood
```

Figure 18: Pseudocode for generating the neighborhood in the classic tabu search.

Of an existing route, we choose to alter the sequence of visited airports rather than the carriers employed in the transportation legs. The airports sequence has more impact on the overall performance of a flight route, as it affects factors like service times, and the number and types of available transportation options. Given an existing sequence of airports, we select airport a and swap it with all other airports that are not in the existing route (line 9). The airport to be swapped is selected progressively, meaning we advance a position in the route sequence each time a new neighborhood needs to be generated. For example, with an initial solution containing three airports, we first generate a neighborhood by swapping the first one in the sequence; next, when the best neighbor solution is used as new starting point to generate the neighborhood, we swap the second airport in sequence. When the last airport in the original sequence is swapped, we reset the counter and start swapping again starting at the origin (lines 5-7). For each new airport combination, we find the optimal combination of carriers with *ReoptimizeRoute* (line 11); the procedure is described in detail in Appendix E). In this procedure, we select the best carrier available per transportation leg. This selection depends on the relative weights assigned to each objective: for example, if we solve the problem instance with $\alpha = 1$ then the fastest carrier per leg will be selected. Once the carriers are selected, we calculate the route's objective values for time, risk and cost. Using the relative weights, we calculate the aggregated objective value, which is then used to sort the neighborhood by ascending value (line 12). Hence, we return the neighborhood (line 16). Procedure *ChooseBestNeighbor* (Appendix E) selects the best non-tabu neighboring solution (line 6 in Figure 17). Once the algorithm terminates, we end up with the best-found route, consisting of the sequence of airports visited and carriers chosen, travel time, risk and costs.

4.3.3 Multiple neighborhoods tabu search

In Chapter 3, we compare different solution approaches to solve the MO-MRCP at hand. As emerged from the context analysis, a major factor influencing the quality of an airfreight route is the chosen combination of origin and destination airport. The principal pitfall of the current route calculation procedure is that the system does not compare different combinations of airports. The tabu search algorithm tackles this problem, as it explores the entirety of a solution's neighborhood in a systematic way, thus comparing all possible origin-ending airport combination, while using a relatively simple procedure. Therefore, we expect it to yield better routes than the current system. However, one limitation of tabu search is that, originally, only one heuristic operation can be performed to generate neighborhoods. The consequence of this aspect is that the number of airports visited still depends on the original outcome of the initial solution. In fact, by swapping airports we do not alter the number of transits employed. This reduces the explored solution space, meaning potentially superior routes with either less or more transits are overlooked. Following this observation, we propose a MNTS algorithm, in which we randomly choose between a swap, add or remove operator when generating a new solutions neighborhood.

Algorithm 3 (Figure 19) shows schematically how we switch between neighborhood structures. At each new iteration, we draw a random (0 to 1) number (line 5). Next, we check how many airports are in the input solution. If we have only two, then we may either choose between a swap move or a add move, with equal probability. The swap procedure is the same as described in Figure 18, except for the fact that we randomly choose which airport we swap. For the add move, we evaluate the neighborhood deterministically by adding all possible non-utilized airports as transit to the current route; then we reoptimize the carriers and calculate the objective values of each solution as usual (lines 13-14). We always add a transit, hence not at the first or last position in the current route; the position at which we add the airport for initial routes with more than two airports is chosen randomly. If we have an initial route with more than two airports, we have one-third chance of using either operator. If we choose the remove one, we obtain new routes by removing alternately each airport in the original sequence. Figure 20 shows graphically how each operator works.

Algorithm 3: Multiple Neighborhoods

```
1   Input: currentSol, p, d, tarr_p, cu, α, β, γ
2   Output: neighborhood
3   Initialize: neighborhood[], airports[], carriers[], objValues[], ran, balValue; set currairports =
    currentSol[0], currcarriers = currentSol[1]
4   Set airports that can be added swapped to current route → swapAddAirports = allAirports − currairports
5   ran = random(0,1)
6   if length of currairports = 2 then randomly choose swap or add operator:
7       if ran < 0.5 then swap airports:
8           randomly choose an airport a to swap from currairports
9           swap airport in route
10          reoptimize carriers & calculate objective values
11      else add transit airport:
12          for (all i in swapAddAirports) do
13              Airports = currairports[add i as transit]
14              follow step 10
15          end for
16      end if
17  else randomly choose swap, add or delete operator:
18      if ran ≤ 0.33 then swap, follow steps 8-10
19      else if ran > 0.33 and ran ≤ 0.66 then add transit airport to route, follow steps 12-15
20      else delete one airport from route, try all combinations:
21          for (all airports i in currairports) do
22              airports = currairports − [i]
23              Reoptimize route, calculate objectives and balanced score
24          end for
25      end if
26  end if
27  sort neighborhood by ascending balanced score (best to worst) objective value balValue
28  return neighborhood
```

Figure 19: Neighborhood generation for the MNTS. At each iteration, we randomly choose a swap, add, or remove operator.

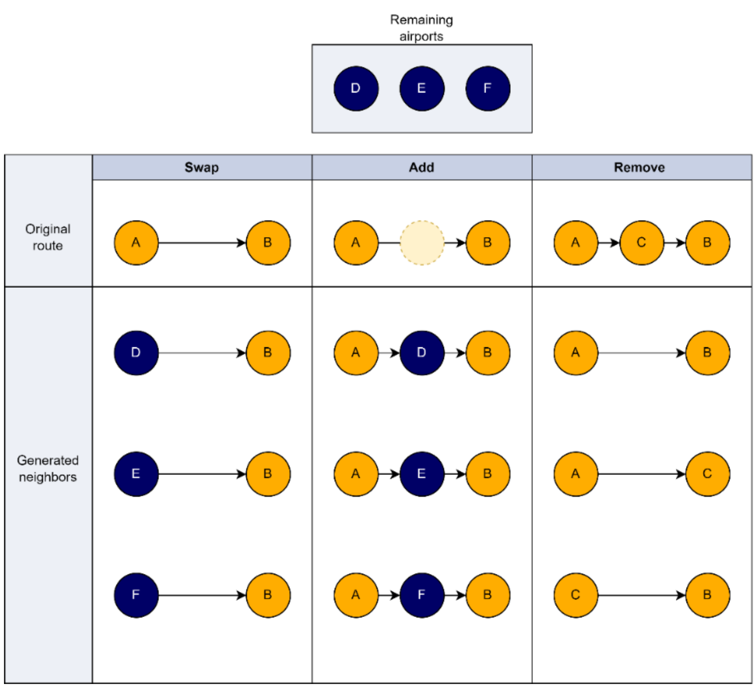


Figure 20: Schematical representation of neighborhood operators used in the hybrid tabu search.

4.4 Pilot study setup

This section addresses the selection of the problem instances on which we test our solution and how the relative parameters to input into the model are estimated. In Section 2.3.2 we classify lanes based on their volume (i.e. number of orders per lane) and aggregated performance. By comparing the two classifications, we acknowledge that lanes being used intensively (A, B volume) and with a relative worse performance (C, B) present with both more room and urgence for improvement. Among these lanes, we seek to find the ones which allow us to test the model entirely. Therefore, we do not consider lanes which do not involve customs clearance. Furthermore, we are particularly interested in lanes with a relatively higher variation in the number of transits, especially when multiple airports seem to be eligible as transit. Finally, we are particularly interested in lanes that present with the opportunity to improve their performance based on decisions that are within the model's intended scope. In other words, lanes in which we observe systematic delay within either the first- and last-mile leg are not suited candidates for experimentation. Indeed, the causing issue of delay is likely bound to miscommunication with the shipper and/or consignee, or low performance by the courier partner, both outside the focus of this research.

Based on these criteria, we select lane A (the airports involved are omitted due to information's sensitivity). To keep the problem instance within reasonable boundaries, we do not include the entire Express network; instead, we only consider those airports which, either directly or indirectly (within a fixed number of stops), offer flight connections between A's origin and destination airports. This results in a network with twelve airports, seven possible pickups and twenty-four possible delivery points, yielding a total of 226 flights and 132 driving options.

4.4.1 Parameter estimations

To estimate the values for the required model parameters, we use the same historical data employed throughout the context analysis. We classify the parameters into three main categories: airports, flights and driving.

Airport parameters

Service and transit times

In terms of time, two types of airport parameters are distinguished: the service time needed for the export or import preparation $exim$ and the transit time s. For both parameters we collect timestamps in the sales and monitoring datasets and group them by airport. For each parameter type, we seek to find the estimate, which is used to solve the model in the deterministic part of the solution approach, and the probability distribution that best models the time distributions at each airport. From the latter, we randomly draw time values, which are used within the stochastic part of the solution. Chapter 5 elaborates in more detail on their utilization while solving the problem instance.

To find the time estimates and distributions, we use the fitdistrplus analysis package in R. This package allows to easily plot data against candidate probability distributions and compare values of well-known goodness-of-fit tests all at once. For simulation purposes, service times can be represented by various distributions. We choose among the most common ones: Weibull, exponential, Gamma and lognormal (Walck 2007). We first plot the time distributions against the density plots of these distributions and visually inspect them. Figure 21 shows an example for $exim$ times (in minutes) at airport a.

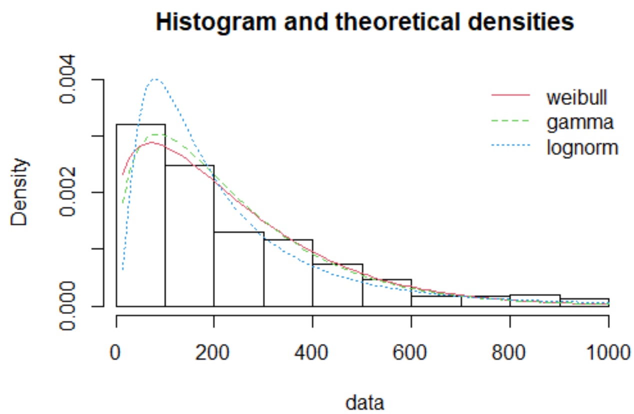


Figure 21: Example plot of the probability density function for service times at airport a. The plot visually compares the goodness-of-fit of the time distribution with the Weibull, Gamma, exponential and lognormal density functions.

By plotting all service time types for each airport, we observe that most shapes resemble what is illustrated in Figure 21. Hence, time-distributions tend overall to be right-skewed. For this reason, we choose to use the data's mode instead of average value as a parameter estimate, as it is more representative of the most common service times (Bhandari 2020). Besides the density functions, we additionally compare the candidate distributions by plotting the cumulative distributions, P-P and Q-Q plots; examples of such plots are attached to Appendix F. As a final step, we compare the outcome of several goodness-of-fit tests[5], following the fitdistrplus documentation (Delignette-Muller 2014) and work by Stephens (1974), and select a probability distribution for each parameter. For a few (minor) airports, insufficient data is available to accurately estimate a probability distribution: to overcome this, we assume they follow the same distribution as a comparable airport (i.e. an airport of comparable size an in the same region, where the time distribution could be estimated with enough confidence).

Risk factors

To estimate the risk r_a (of incurring delay) when visiting a certain airport, we again use timestamps form the sales and monitoring datasets. One of the simplest ways to quantify is risk is to multiply the probability of an event times the size of its consequence (Ragheb 2020). In this case, we calculate the probability of delay at an airport as the number of delay occurrences divided by the total number of timestamps, and the consequence as the average delay. We then multiply them to get the estimated delay per airport and normalize them to get a uniform risk scale.

Costs

[5] Chi-squared, Kolmogorov-Smirnov, Cramer-von-Mises and Anderson-Darling tests, together with the Akaike's and Bayesian Information criteria.

For the estimation of costs of airports services c_a, we use costing data of the Express network. As described in Section 4.1 , several ground handling operations like security scans, transportation and packing are generalized as airport service within the proposed model. Therefore, we sum the fees of all these operations and take the average of those sums, taking into consideration different handling partners, and thus obtain a cost estimate per shipment. Given the sensitivity of costing data, we do not use accurate cost estimates on purpose, but choose rather old fares taken from an unspecified year in the costs sheet.

Customs service time, risk and cost

The proposed model additionally differentiates regular airport operations from customs clearance, as the latter is not always required and is usually bound by stricter requirements, like additional documentation and tighter office hours. To estimate customs-specific parameters (cs, cr, cc), we group customs-related datapoints per airport and estimate parameters and time distributions like for the regular airport operations.

Flight parameters

Flight times

To estimate the distributions of travel times per flight carrier, we use SSLC'S historical flight data, and follow a similar approach as for the airports service times. To ensure that the statistical tests can be performed at the required level of significance, we group the available datapoints by combination of origin and destination airport, rather than by individual flight number. From there, we estimate the flying time distributions using again the fitdistrplus package and assign that distribution to all flights sharing that same origin-destination combination. Again, we choose to use the mode rather than the mean as statistical measure for the estimate of t_{ij}^{f}, given the data's skewness.

Airline risk factors

To estimate the risk factors (r_{ij}^{f}), we use historical flight data and follow the same method used for the airport's parameters. For each flight number, we calculate the probability of delay and multiply it by the average delay; the obtained expected delays are then again normalized into a uniform risk scale.

Airline costs

To estimate the flying costs (c_{ij}^{f}), we use costing data of the Express network. The costs are calculated based on airline-specific fees per shipment. Again, we use non-actual data given the sensibility of information on partners' contracted fees.

Driving parameters

Driving times

For the estimation of driving times, we use the Google Maps Distance Matrix API as data source, given SSLC does not have (sufficient) driving data for all modelled transportation legs (e.g. the fictive transports between airports). The API allows to request three types of estimates: a pessimistic, neutral and optimistic driving estimation. The first one gives an indication of the travel time given high traffic conditions, the second one is the average driving time in normal conditions and the third one represents the expected travel time with minimal traffic on the road. For parameter t_{ij}^{v}, we use the middle one as estimate. To include variability for the stochastic evaluation of the solution, we use the optimistic and pessimistic time estimates as boundaries for a uniform distribution.

Driving risk factors

While the Google API is useful for the estimation of driving times, it cannot be used to estimate/predict delays on a certain route, nor the probability of such delay occurring. To overcome this issue, we use the historical monitoring data at hand containing first- and last-mile legs to estimate the expected delay (in minutes) per km. We observe that, overall, both the frequency and magnitude of delays diminishes with longer driving distances. Based on empirical analysis, we estimate the expected delay per km as shown in Table 15, where a certain factor is applied within a predefined km range. For example, for a driving leg of 200 km, the expected delay is then calculated as 0.11 * 200 = 22 minutes. After calculating all expected delays, we normalize the values on a scale 0-1, and thus obtain the risk factors r_{ij}^{v} for each driving leg.

Table 15: Expected delay per km. For each driving leg, the factor belonging to the respective distance in km is taken and multiplied by the leg's distance to obtain the expected delay.

Expected delay p/ km	
km	factor
3	0
159	0.45
315	0.11
471	0.07
627	0.06
783	0.015
> 783	0.01

Driving costs

Finally, to estimate the driving costs c_{ij}^{v}, we derive an average cost per km over the available courier partners using the Express costing dataset. The amount and type of partners available for each leg depends on the country at which the leg starts: in case of multiple possible couriers, we take the average of their fees per km as estimate. Again, we use different values than in reality due to data compliance.

4.5 Simheuristics procedure

Having estimated distributions of all (variable) modelling parameters, we use them to stochastically evaluate the solutions generated by either tabu search algorithm. Hence, we implement a simheuristic. Simheuristic algorithms use information about the stochastic nature of the problem being solved to simulate scenarios. For each scenario, the algorithm evaluates the deterministic solution: by simulating multiple scenarios we can thus get an estimate of the solution's overall performance in realistic settings (Juan, et al. 2015).

The main source of stochasticity in the routing problem is time. Various events can either cause longer or shorter service and/or transportation times. This can eventually lead to delays and potentially additional costs. For example, a parcel missing its flight due to a longer drive time to the airport will probably be delayed with respect to the expected delivery time; furthermore, the rebooking of a new flight brings additional costs. Lastly, if the parcel's delay falls outside the maximum allowed late time, an additional penalty cost will be charged. Figure 22 shows schematically how the simheuristic algorithm works. We start by generating the initial solution following the steps described in Section 4.3.1 . Next, we evaluate the generated solution against a simulation scenario. First, we randomly draw a time realization for the first-mile leg. Based on this time value, we calculate the new arrival time at the origin airport. If the time differs (positively) with the arrival time calculated by the deterministic solution, we register a delay. Next, we randomly draw a value for the export time and customs clearance, and thus calculate at what time the shipment is ready for its next transport. If the shipment misses its flight, we check for the next best option leaving for the following airport on the route; this may either be a flight or a vehicle. Delay and extra costs for rebooking are eventually registered. Similarly, we carry on drawing randomly transportation and service times, while keeping track of time, costs and delays, until we calculate the final delivery time realization.

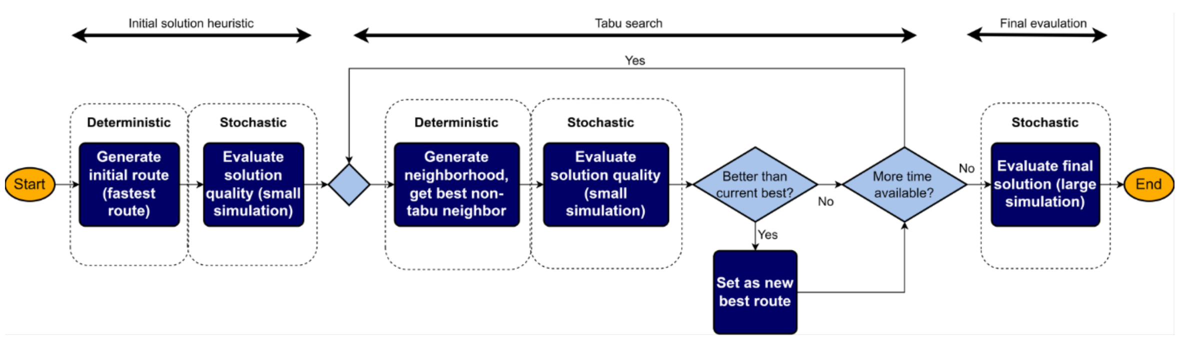


Figure 22: Simheuristic solving process. The initial solution is generated and evaluated with a short simulation. The deterministic solution is then used by the tabu search to generate and solve new neighbors deterministically. Each neighbor is first evaluated stochastically and then compared to the current best solution. The final solution is then evaluated stochastically.

By comparing the realized delivery time with the original one given by the deterministic solution, we calculate the deviation from the agreed delivery time. If the deviation exceeds either the earliest or latest allowed delivery time, we add an extra penalty to the total costs. As mentioned, SSLC currently does not have a defined system by which such penalties are enforced, as this differs for each customer and the corresponding agreements. Normally, customers try to get a discount on the shipment's price, depending on the magnitude of delay. In concordance with SSLC, we decide to initially penalize shipments deviating 12 hours from the agreed delivery time (either too early or too late), with a starting discount of 50% of the shipment's price. The discounted price can accumulate up until 24 hours of deviation: from there we enforce a penalty of 100% of the total price. Naturally, the penalty system has a significant influence on the model outcomes; in Chapter 5 we perform a sensitivity analysis to highlight the effect of the penalties on route calculation.

We repeat the steps described above for several iterations and take the average time, delay and cost. With these values we ultimately calculate the aggregated objective and store these values as current best. We then proceed with the tabu search metaheuristic. For each new neighbor, we evaluate its quality by running the same number of simulation iterations and comparing the outcomes with the current best ones. Once the algorithm has solved, we end up with the best-found solution so far. This solution is ultimately evaluated by running an additional number of simulation iterations. This enables us to assess the route's robustness in terms of the three objectives. By running the solution for different modelling configurations, we are ultimately able to compare their relative robustness. Section 5.2.4 shows the results of this stochastic evaluation.

4.6 Conclusion

In this Chapter, we answer the research question: "How should the solution approach be designed?". Section 4.1 outlines the assumptions, requirements and scope upon which we base the mathematical formulation of the problem. The solution's focus is bound to the optimization of transit time, risk of delay and operationg costs of SSLC's air freight routes within the Express network. The model's focus is shifted towards providing an optimal route given the best sequence of airports to visit and combination of carriers to operate. In the decision making process, constraints bound to the airports' available facilities, office hours and flights' departure times are taken into account. Furthermore, the model seeks to enforce the timely arrival at delivery by penalizing arrival times outside of predefined boundaries. To keep the model's complexity within reasonable margins, we make several assumptions. For example, we do not account for special customs cases, nor do we consider capacity or goods-related constraints that would restrict the number of flight options for certain shipments. The choice of which partner for airport handling operations or courier partner for the first- and last-mile legs are also left out for simplicity. Based on these modelling decisions, we formulate the formal mathematical model to be solved in Section 4.2 . Section 4.3 presents the algorithm to solve our problem instances: first, we generate an initial route using the same logic as currently employed by SSLC's operating system. This solution is then altered and evaluated iteratively in the tabu search algorithm, which seeks to find better route alternatives by swapping airports in the route seqeunce, and keeping track of visited solution areas by keeping a tabu list. Here we test two variants: one stores moves and one entire solutions in the tabu list. Furthermore, due to the limited reach of the classic tabu search, we formulate a hybrid version of the algorithm, which adds to possibility to either add or remove airports in seqeunce. In Chapter 5 we run all three versions and choose the best performing one. To test our solution, we select one lane with a sufficient number of possible problem instances. Using historical data, we estimate the required modelling parameters directly for both the deterministic as the stochastic parts of the solution. Finally, in Section 4.5 we design the simheuristic. To evaluate a deterministic solution, we randomly draw time estimates for each leg within the input solution. By doing so sequentially, we calculate the eventual delays and additional costs. We repeat this procedure for a predefined number of iterations and thereby obtain an average solution performance under uncertainty.

Chapter 5 – Solution results

This chapter answers the research question: "How does the proposed solution perform compared to the current routing algorithm?". The first step is to determine the experimental design to test the solution. Section 5.1 tackles this part. Once the design is defined, we proceed on executing the experiments sequentially. Section 5.2 presents their outcomes. In Section 5.2.1 , we report the outcomes of the tuning experiments, and hence the settings we use to run each solution. In Section 5.2.2 , we solve the problem deterministically. First, we check the quality of the two tabu search algorithms (M1 and M2) and the multi neighborhood tabu search against an exact solution; next we analyze the algorithm outcomes in more detail. In Section 5.2.3 we evaluate the stochastic solution outcomes and make inferences about the best performing one, compared to the current situation at SSLC. In Section 5.2.4 we analyze the robustness of the proposed solutions and compare the stochastic evaluation with historical data outcomes. Finally, in Section 5.2.5 , we perform a sensitivity analysis to evaluate the change in routing decisions given a chosen delay penalization policy.

5.1 Experimental design

In this section, we define the experiments we execute to comprehensively test our proposed solution. We define their setup, intended purpose and order of execution. In total, we distinguish five categories of experiments, summarized in Table 16. We execute all experiments using a Python 3.9.13 engine on the Spyder IDE, on a computer with Intel Core i5 10310U processor of 2.21 GHz with 16 GB RAM.

Table 16: Overview of experiments with explanation of the goal of each experiment

Number	Experiment	General goal	Practical goal
1	Model Tuning	Finding model- and algorithm-specific settings for optimal solving performance.	Having the model ready for executing the main experiments (2-5).
2	Deterministic solution evaluation	Comparison of the proposed solution algorithms without considering uncertainty.	Establishing how good the proposed solutions are compared to an exact method and finding the one with the best improvement to the initial solution (and hence the current routing system).
3	Simheuristic solution evaluation	Analysis of the solutions' general performance under uncertainty and comparison of the best solution with the current situation.	Selecting the best-performing solution and providing SSLC with insights into how it improves their current routes.
4	Stochastic evaluation	Robustness analysis of the proposed solution.	Providing insights into how randomness affects the results from experiment 4.

5	Sensitivity analysis	Analysis of the solution's routing behavior under differing penalty cost settings.	Providing alternatives for SSLC on how to penalize delivery deviations in a potential future routing system.

Throughout experiments 1-5, we use five different combinations of the model's objective function, each one representing a particular preference towards the optimized objectives. Table 17 summarizes them.

Table 17: Objective weights configurations of the five tested models.

Model name	**Objective weights configuration**		
	α **(time)**	β **(risk)**	γ **(cost)**
Time – greedy	1	0	0
Risk-greedy	0	1	0
Cost – greedy	0	0	1
Balanced	0.33	0.33	0.33
Weighted	0.41	0.51	0.08

5.1.1 Model tuning

The first category of experiments is concerned with finding the settings for each combination of model setup and solving algorithm yielding optimal results against reasonable computational time. Within our solution approach, we distinguish two parameter categories, namely simulation-specific and metaheuristic-specific. The number of iterations for the intermediate ($iter_i$) and final ($iter_f$) solution evaluations within the simheuristic procedure belong to the former class. These parameters mainly influence the variability of results. To evaluate the simulation outcomes with accuracy, we therefore seek the configuration that minimizes the variation of simulation outcomes. Metaheuristic-specific parameters affect the tabu search's performance, and therefore, the quality of the obtained solution. Hence, we execute experiments to find the combination of tabu list length (T_{length}) and number of tabu iterations (T_{iter}) which returns the best results for all objectives combined. Finally, we investigate the impact of the constraint of departing within 24 hours from the origin (we refer here to it as $depwithin$) on the solutions quality, and ultimately decide whether we should relax it or not.

We start by finding the best configuration for the simulation-related parameters, as the variability of results directly influences the quality of solutions found. To measure the variation of results, we use the coefficient of variation[6] of each objective. Additionally, we also measure the run time needed to solve the problem instance, as the model should solve within reasonable time. Next, we perform tuning

[6] $\frac{\sigma}{\mu}$ (Introduced in Chapter 2).

experiments to find the optimal values for T_{length} and T_{iter}. To measure the quality of results, we observe the average outcomes for all three (time, delay and cost) objectives and the run time. Finally, we use the same latter metrics to compare the solution's quality either with or without the usage of the $depwithin$ constraint.

As mentioned in the previous section, we need to tune five model configurations. We solve each model using three solution methods: tabu search M1 and M2 and MNTS. This yields a total of fifteen solutions: for each solution we perform separate tuning experiments, as the optimality of settings varies depending on the type of model being solved and the method used. We make an exception for the simulation-related parameters: since the simheuristic procedure is the same regardless of the model and algorithm used, we execute tuning experiments only on one combination (classic tabu search with moves storage, on the balanced model) and extend these settings to the remaining solutions.

To find promising ranges to perform the tuning experiments, we execute preliminary tests. Per parameter, we compare the outcomes from largely differing settings (e.g. with $iter_f = 10$ and $iter_f = 1000$, $iter_f = 1000$ and $iter_f = 10000$, etc.) and thus identify the range yielding the most promising outcomes, at a reasonable computational burden. Having identified those ranges, we can narrow the search. We start by setting a parameter to the range's lowest value, while keeping all others fixed. We then increment the parameter gradually until reaching the maximum value in the range, select its best performing value and proceed with tuning the next parameter (Crainic, et al. 1993). Table 18 summarizes the tuning experiments.

Table 18: Overview of the tuning experiments, with the respective parameter, the experimental range, the value by which we increment the parameter and the measured metrics.

Parameter	Experiment range	Increment	Measurement metrics
$\boldsymbol{iter_i}$	100-5000	100	Coefficient of variation per objective (time, delay, cost), average running time.
$\boldsymbol{iter_f}$	1000-30000	1000	Coefficient of variation per objective (time, delay, cost), average running time.
$\boldsymbol{T_{length}}$	1-9	1	Average objective value (time, delay, cost), average running time.
$\boldsymbol{T_{iter}}$	50-1500	50	Average objective value (time, delay, cost), average running time.
$\boldsymbol{depwithin}$	24 or ∞	N.A.	Average objective value (time, delay, cost), average running time.

Within the experiments, we increment parameters differently. For $iter_i$, we advance the value by 100 units; for $iter_f$ by 1000, for T_{length} by 1 and for T_{iter} by 50. Finally, the tuning of $depwithin$ consists of two experiments, with value set either to 24 or to a very large number (meaning in practice the constraint is relaxed). Section 5.2.1 reports the outcomes of the tuning experiments.

5.1.2 Deterministic solution evaluation

The next step is to compare the generated solutions by solving the problem deterministically. This experiment consists of two parts. First, we implement the model formulation described in Section 4.2 into Gurobi, which is a Python library that allows us to solve the problem exactly. For each solved problem instance, we hence obtain either its optimal solution or the optimality gap from the returned objective value. Next, we solve the problem using the proposed metaheuristics and compare the outcomes with the exact results. This allows us to get an impression of how close to optimality the metaheuristic solutions get. Next, we take a closer look at the results obtained with the metaheuristics. The goal of this analysis is twofold. First, we examine routing decisions of the three proposed solution algorithms and check whether they make sense. Secondly, we seek to find the solution approach that yields the most improvement compared to the generated initial solution. As discussed in Section 4.3.1 the algorithm generating the initial solution follows the same logic as SSLC'S current routing system. By comparing its outcomes with the other solutions', we can make inferences on which approach could represent an improvement in the current route calculation process and hypothesize which one is likely to perform best in the stochastic evaluation.

To obtain exact solutions within a reasonable time boundary, we set a time limit of five minutes for the MIP, as this is the maximum time allowed by the current system to find a route. If no optimal solution is found within that limit, the model returns a suboptimal solution with the corresponding optimality gap. Using historical data from the test batch selected in Section 4.4 , we generate 49 test instances. Each instance is generated by using the respective order's pickup and delivery location, pickup time and day of the week. We solve all instances using the five combinations of objective weights reported in Table 17, store relevant KPIs and compare the average outcomes for each solution. Section 5.2.2 elaborates on the results.

5.1.3 Simheuristic solution evaluation

In this experiment, we consider the stochastic behavior of travel and service times, incorporate it into our routing decisions and analyze its effect on the overall performance of a route. Here we generate the same 49 instances as described previously and solve them using the approach displayed by Figure 22. First, we are interested in observing the degree to which considering uncertainty affects the solutions' routing decisions. Therefore, we compare the generated routes with the ones calculated deterministically in the previous experiment. Secondly, we are interested in observing whether the best-performing solution improves the current situation as analyzed in Chapter 2, given the average results of the simulated scenarios. We thereby compare the average outcomes of the 49 solved problems with the KPI values measured during the context analysis. Finally, we are interested in observing whether the solution's (eventual) improvements are significant. To check this, we substitute the test lane's actual KPI values with the ones obtained by this experiment and re-classify the lane using the same ABC method applied during the context analysis. Section 5.2.3 presents the outcomes.

5.1.4 Stochastic evaluation

Having incorporated randomness in our solution, we are not only interested in the outcomes in general, but also in their variability. Hence, we examine their robustness given a stochastic setting. To illustrate the stochastic effect on each solution, we select and analyze two of the 49 previously solved problem instances (we call them instance 1 and 2). In particular, we pick the two having the most commonly used pickup and delivery points, in order to get a representative picture of how the simulation affects the deterministically calculated solutions. First, we examine box plots showing the spread of datapoints per relevant KPI resulting from the 15000 final simulation iterations for each solution type. Next, we seek to validate the simheuristic outcomes. To this end, we take the original (historical) routes corresponding to the two test instances and compare their outcomes with the ones we obtain with the simheuristic. This way, we verify whether the simulated scenarios are realistic compared to real orders.

5.1.5 Sensitivity analysis

Finally, we analyze the impact of the designed penalty system used to prevent large deviations from the preset delivery time. As explained in Section 4.5 , we start at a deviation of 12 hours and a penalty of 50% of the total costs and accumulate this amount up until a maximum deviation of 24 hours, after which we add a penalty corresponding to 100% of the costs. Following this formulation, we can summarize the calculation of penalty costs with the formula:

$$penalty = total\ cost * \left(1 - \frac{upper\ bound - time\ difference}{upper\ bound}\right) \mid time\ difference \geq threshold$$

Here, the $upper\ bound$ represents the maximum deviation allowed after which we enforce a 100% cost penalty; the $threshold$ is the minimal time difference at which we start penalizing the route. For our experiments, we distinguish three alternative settings for the $upper\ bound$ and the $threshold$. The earliest at which SSLC accepts claims is for deviating 6 hours from the agreed delivery time. On the other hand, parcels delivered later (or earlier) than 48 hours from the agreed time imply almost certainly a 100% restitution of the agreed fee. Given these two boundaries, we distinguish a “strict” penalizing policy (we penalize deviations from 6 hours up until a maximum of 12), a “relaxed” policy (starting at 24 hours with a maximum of 48) and a “moderate” policy (starting at 6 up until 48 hours, thus with a slower percentage increase of penalty costs).

Since the penalizing policy only affects routing decisions within the models which optimize the cost objective, we test the above variants on the cost-greedy, balanced and weighted model variants. Furthermore, we do not use all 49 problem instances for the test, as it would be too time-consuming and not very relevant either. Instead, we only select problem instances whose outcomes of delivery delay fall within the range of 6-48 hours. We thereby select three problem instances as test sample.

5.2 Experimental results

5.2.1 Tuning experiments

This section presents the results of the tuning experiments and hence the settings we use to run our solutions. Given the large number of performed experiments, we omit the detailed outcomes; instead, we summarize the results in Table 19 and attach their performance metrics in Appendix G. In the summary, we refer to the tabu search storing moves as tabu search M1 and to the one storing entire solutions as tabu search M2.

By first testing the simulation-bound parameters, we find the optimal combination for $iter_i$ and $iter_f$. As mentioned in Section 5.1 , we keep these same settings on all solution approaches. For the metaheuristic parameters, the most striking observation is the similarity between the settings for the tabu search variants. During the preliminary experiments we observe that the latter two yield identical solutions for all optimized model types. This is unusual, since we would expect to see differences in the output, given their differing use of memory in the search process. In Section 5.2.2 we elaborate on this finding in more detail. Assuming the two variants have indeed the same routing behavior, it then becomes trivial that they have almost identical settings for the tabu list and tabu iterations. Another observation is that the MNTS performs best with lists slightly longer than the normal version. Similarly, when using an aggregated objective function, better solutions are found using an increased tabu list length. The most apparent difference lies within the number of tabu search iterations, with the multiple neighborhood model returning increasingly better results up to 1000 iterations. The latter explores a bigger solution space than the classic ones, since it utilizes more heuristic operators simultaneously. Consequently, to explore this space with better results, more diversification is generally needed at the start of the solution procedure (achieved by keeping longer tabu lists), and, at the same time, more iterations are needed to converge towards the optimal solution. Finally, we notice that all solution types perform better (or at least as good) whenever the $depwithin$ constraint is relaxed. Since it limits the explorable solution space, this outcome is expected; we thus relax the constraint for all remaining experiments.

Table 19: Summary of the experimental settings per solution type, found after the tuning experiments.

Tabu search M1					
α, β, γ	time-greedy	risk-greedy	cost-greedy	balanced	weighted
$iter_i$	1500	1500	1500	1500	1500
$iter_f$	15000	15000	15000	15000	15000
T_{length}	5	5	5	7	7
T_{iter}	100	100	100	100	100
$depwithin$	∞	∞	∞	∞	∞
Tabu search M2					
α, β, γ	time-greedy	risk-greedy	cost-greedy	balanced	weighted
$iter_i$	1500	1500	1500	1500	1500
$iter_f$	15000	15000	15000	15000	15000
T_{length}	5	5	5	6	7
T_{iter}	100	100	100	100	100
$depwithin$	∞	∞	∞	∞	∞
Multiple neighborhoods tabu search					
α, β, γ	time-greedy	risk-greedy	cost-greedy	balanced	weighted
$iter_i$	1500	1500	1500	1500	1500
$iter_f$	15000	15000	15000	15000	15000
T_{length}	6	6	6	7	7
T_{iter}	1000	1000	1000	1000	1000
$depwithin$	∞	∞	∞	∞	∞

5.2.2 Deterministic solution performance

In this section we present and discuss the results of solving the problem deterministically. First, we compare the results of the MIP model and the metaheuristics; next, we discuss the latter results and corresponding routing decisions in more detail. Before proceeding, we make some general considerations. We first notice that the two tabu search variants (M1 and M2) provide with identical results, confirming the behavior already observed during the tuning experiments. This is unexpected, as the algorithm's way of using memory throughout the search process influences its outcomes, meaning the two variants should thus yield different routes. A possible explanation for this could be related to the size of the problem instance we study: since it is not real-sized and thus with less options, it might well be the case that both variants end up finding the same solution, which becomes also more plausible if we consider that both still use the same swapping procedure and thus search in similar directions. In the remainder of this section, we report the TS-M1 and TS-M2 outcomes as one.

MIP model vs TS and MNTS

Table 20 shows the average performance comparison of solving the 49 generated problem instances with each model configuration, using exact and metaheuristic solutions. For the MIP model, we report the average objective value, optimality gap and running time. Please note that for the greedy models, we only report the average value of the optimized objective, whereas for the balanced and weighted models we report all three (in order time, risk, cost). For the TS and MNTS metaheuristics, we show the average percentual difference with the MIP objectives and the average run time in seconds.

If we compare the MIP run times and optimality gaps from the greedy models, we observe a notable difference between the time-greedy one and the remainder. Whereas the MIP model is almost always able to solve the problem optimally considering either risk or costs, the opposite is true when the transit time is optimized. This is explainable considering that the number of possible combinations of decision variables influencing the time objective is significantly bigger than for the other two. In fact, in our problem setup, the price or risk of operating a specific carrier or station does not vary based on the day of the week. On the other hand, selecting a carrier on a particular day of the week has impact on the final delivery time. Additionally, flight carriers belonging to the same airline all have the same (fixed) cost regardless of the operated route, while we also do not discriminate between flights operating on the same origin/destination combination in terms of carrier-bound risk. Naturally, flights from either different (or the same airlines) have several departure schedules for the same origin/destination combination; hence, for the same route we have more options affecting the time objective than for the other two. Consequently, the solution space of the time objective is significantly bigger than the others, explaining why it is harder to obtain an optimal solution within the same time limit. Because of this, we observe that the metaheuristics yield the closest performance for the time-greedy configuration, with the MNTS even slightly outperforming the MIP model. Interestingly, we find confirmation to these considerations if we compare the MIP's performance on the balanced and weighted models. In comparison to the former, the latter model attributes more importance to the time objective. Consequently, the average optimality gap increases as the average run time. If we increase the MIP's solving time limit to 30 minutes, we observe that the percentual objective difference between the TS and MNTS and the MIP increases (hence, the MIP outperforms the algorithms). However, the optimality gap reduces by just 11% on average. In general, we observe that the MIP model cannot solve all 49 instances optimally for neither model type. Additionally, both TS and MNTS algorithms obtain solutions fairly close to the MIP model and in considerably less time, justifying our decision of opting for these methods to solve the problem.

Table 20: Comparison of results for the MIP model and the TS and MNTS metaheuristics. For the greedy model, we present the average value for the optimized objective; for the balanced and weighted model we present in order, the average transit time, risk and costs obtained.

	MIP			TS (M1 & M2)		MNTS	
Model type	Objective value	Optimality gap	Run time (s)	% Difference	Run time (s)	% Difference	Run time (s)
Time-greedy	20.22	39.65%	300.00	0.84%	39.24	-2.18%	150.08
Risk-greedy	0.10	0.93%	60.25	30.05%	40.23	22.94%	150.76
Cost-greedy	435.89	0.88%	59.85	13.63%	40.06	12.80%	159.83
Balanced	20.67, 0.28, 484.28	9.40%	224.83	4.20%	40.20	3.78%	162.20
Weighted	21.68, 0.28, 501.47	16.33%	288.50	3.75%	40.22	3.15%	161.79

Algorithm performance and routing decisions

Table 21 shows the solutions' average performance compared to the initial solution, given the respectively solved model configuration. A complete overview of the results is furthermore attached in Appendix H. Besides comparing the average KPI values (transit time in hours, costs in euros and risk using a scale 0-1) and running times (in seconds), we also attach the percentage of problem instances where the final route differs from the one yielded by the initial solution. The green-marked numbers represent a decrease (hence improvement) in the objective values, the red ones an increase.

Table 21: Overview of the average performance comparison per solved model type. We compare the initial solution's average KPI performance with the three proposed solution algorithms. Furthermore, we attach the percentage of scenarios for which a different route than the initial solution's is returned and the average run time (s) per solution.

Time-greedy					
Solution type	Transit time	Total risk	Total cost	Initial change	Run time
Initial solution	26.53	0.39	730.82	N.A.	19.87
Tabu search M1	-23.14%	-21.31%	45.98%	63.27%	37.02
Tabu search M2	-23.14%	-21.31%	45.98%	63.27%	39.20
MNTS	-25.44%	-24.96%	53.31%	77.55%	150.08
Risk-greedy					
	Transit time	Total risk	Total cost	Initial change	Run time
Initial solution	26.53	0.39	730.82	N.A.	19.87
Tabu search M1	285.85%	-67.19%	881.04%	100.00%	37.33
Tabu search M2	285.85%	-67.19%	881.04%	100.00%	43.12
MNTS	255.60%	-69.00%	998.20%	100.00%	150.76
Cost-greedy					
	Transit time	Total risk	Total cost	Initial change	Run time
Initial solution	26.53	0.39	730.82	N.A.	19.87
Tabu search M1	90.65%	-25.30%	-30.94%	69.39%	38.02
Tabu search M2	90.65%	-25.30%	-30.94%	69.39%	42.10
MNTS	35.31%	-28.73%	-33.45%	75.51%	159.83
Balanced					
	Transit time	Total risk	Total cost	Initial change	Run time
Initial solution	26.53	0.39	730.82	N.A.	19.87
Tabu search M1	18.69%	-25.16%	-30.83%	71.43%	38.02
Tabu search M2	18.69%	-25.16%	-30.83%	71.43%	42.37
MNTS	18.92%	-28.37%	-33.38%	77.55%	162.20
Weighted					
	Transit time	Total risk	Total cost	Initial change	Run time
Initial solution	26.53	0.39	730.82	N.A.	19.87
Tabu search M1	15.11%	-25.43%	-28.71%	71.43%	38.09
Tabu search M2	15.11%	-25.43%	-28.71%	71.43%	42.35
MNTS	11.07%	-26.61%	-31.51%	77.55%	161.79

If we run by each model setting, we observe that for each respective weights combination, all three methods mostly alter the initial solution. Exception to this is whenever the initial solution generates a route with a direct flight, which in most cases is still the best option. In general, all three solutions improve the initial one considerably. With this we can infer that all three could represent a valid candidate to improve the logic beneath the current routing system at SSLC. In particular, the MNTS appears to improve the initial solution the most, in all settings. This is expected, as the algorithm is capable of exploring a considerably larger solution space than the remaining methods, as it utilizes more heuristic operators and includes randomness in its search. The latter aspect has, however, implications for the run time: in fact, this solution is considerably slower than the others. In the remainder of the section, we analyze the routing choices made by either solution and how they affect the performance. Table 22 provides an overview of the most significant figures. Starting with the chosen origins and destinations, we categorize airports as either major (MA), medium (ME) or minor (MI) hubs, depending on their yearly passenger volume[7]. We thus show the percentage of airports from either category chosen as origin/destination for a route. Furthermore, we also indicate the solution's preferred carrier type, being either flight, vehicle or mixed (the latter means a combination of flights and driving options are used for transportation between origin and destination airports). Finally, we show the number of routes with at least one transiting airport. We group the results of the two classic solutions together, as they are identical.

Table 22: Overview of the solutions' routing policies for solving the problem deterministically. For each solution, we indicate the percentage of major (MA), medium (ME) or minor (MI) hubs chosen either as origin or destination. Additionally, we indicate the solution's preferred carrier type and the number of generated routes with at least one transit.

Time-greedy								
Solution type	Preferred origin			Preferred destination			Preferred carrier type	Number of transits
	MA	ME	MI	MA	ME	MI		
Initial solution	43%	57%	0%	67%	14%	18%	Flight	13
Tabu search (M1 & M2)	76%	4%	20%	63%	22%	14%	Flight	13
MNTS	76%	8%	16%	55%	37%	8%	Flight	9
Risk-greedy								
Solution type	Preferred origin			Preferred destination			Preferred carrier type	Number of transits
	MA	ME	MI	MA	ME	MI		
Initial solution	43%	57%	0%	67%	14%	18%	Flight	13
Tabu search (M1 & M2)	0%	100%	0%	0%	100%	0%	Vehicle	13
MNTS	0%	100%	0%	0%	100%	0%	Vehicle	0
Cost-greedy								
Solution type	Preferred origin			Preferred destination			Preferred carrier type	Number of transits
	MA	ME	MI	MA	ME	MI		
Initial solution	43%	57%	0%	67%	14%	18%	Flight	13
Tabu search (M1 & M2)	94%	6%	0%	67%	14%	18%	Flight	13
MNTS	94%	6%	0%	67%	14%	18%	Flight	3
Balanced								
Solution type	Preferred origin			Preferred destination			Preferred carrier type	Number of transits
	MA	ME	MI	MA	ME	MI		
Initial solution	43%	57%	0%	67%	14%	18%	Flight	13
Tabu search (M1 & M2)	94%	6%	0%	69%	14%	16%	Flight	13
MNTS	94%	6%	0%	67%	14%	18%	Flight	6
Weighted								
Solution type	Preferred origin			Preferred destination			Preferred carrier type	Number of transits
	MA	ME	MI	MA	ME	MI		
Initial solution	43%	57%	0%	67%	14%	18%	Flight	13
Tabu search (M1 & M2)	88%	6%	6%	69%	14%	16%	Flight	13
MNTS	90%	6%	4%	69%	14%	16%	Flight	18

[7] MA > 10000, 5000 < ME ≤ 10000, MI < 5000 passengers yearly, the values used are the averages of the years 2020-2022, source: IATA.

Time-greedy model

Within the time-greedy setting, all three solutions improve the transit time KPI considerably. This is done at the expenses of the route's total costs, which increases by approximately 50% in all three cases. If we look at the solutions' routing policies, we observe that all three tend to increasingly use minor hubs as their starting airport. Indeed, most minor hubs in the network have some very interesting direct flight options, but they are also more expensive to operate. The main advantage of using minor hubs is that they are not as burdened by high cargo volumes as major hubs and so have potentially faster service times. Furthermore, they can be additionally beneficial to a route's speed whenever they are closer to the pickup point than other airports. The major downsize of using them is that there are less frequent flights, meaning that they can be used on limited days of the week and moments of the day. When this is not possible, the model then prefers operating through major hubs. The use of minor hubs also implicates more risk, as a parcel missing a flight from a minor hub needs to wait considerably longer for the next option than at a bigger hub. This is reflected by the results of the simheuristic in Section 5.2.3 . In the results in Table 21 we also observe a reduction in total risk factors. Both models predilect using direct flights, as they are usually faster and less risky. We observe this fact clearly in the MNTS solution's decisions, as it predilects eliminating the initial route's transit (please recall that the classic tabu search cannot alter the number of transits) and take a direct option from another hub. The preferred carrier of all solutions is the aircraft, as it is faster than driving. However, when stochasticity affects the service times at the airports, we observe this is not always the case. More detail in Section 5.2.3 . Finally, the time-greedy setting has the most variation of carriers used: at least five distinguished airlines are booked, which is significantly more than in other models. This is because, cumulatively, they provide more ad-hoc options, which is beneficial for the transit time.

Risk-greedy

The first figure that pops out from the results of the risk-greedy model is that all three solutions always yield a different route than the initial one. If we look at the total risk KPI, we observe a substantial improvement by all three solutions, with the multiple neighborhood one performing slightly better. However, by analyzing the remainder KPIs, we can argue this is not per se beneficial. The risk reduction comes at the expense of both transit time and costs, which increase respectively by about a factor 3 and 10. By looking at the routes we quickly learn why: they are almost identical for all solutions, regardless of the solved problem instance. In particular, the most frequently used origin is often the farthest airport for almost all generated pickup points, explaining the exaggerated transit times and costs. This exposes a major blind spot of the model's design. Indeed, the model accounts only for the individual risk of delay attached to airports or carriers. Therefore, a purely risk-aversive model will always choose the combination of airports and carriers with the lowest risk factor, regardless of any other factor, that would otherwise be trivial to consider when planning a route. This leads to routes that do not make sense in practice and are therefore, inoperable. Furthermore, a third source of risk is overlooked in this setting, namely the risk of delay given a route with too tightly planned connections. This becomes clearer when including randomness in the simheuristic procedure. The hybrid model tends to eliminate transits, as each additional transit adds up in risk. Moreover, all solutions prefer the use of vehicles. As mentioned, the airports with less risk in this particular network are generally also the farthest from most pickup and deliveries. As shown by Table 15 in Section 4.4.1 risk reduces significantly for driving options as the driving distance increases, making it therefore a "better" option. In general, we can conclude that the routes provided by the risk-greedy configuration are not practical and therefore not useful to SSLC.

Cost-greedy

Within the cost-greedy model, we see all three solutions improve both costs and risk, affecting the transit time negatively. Again, the MNTS solution yields the best improvement overall. With respect to the initial solution, we observe a significant increase in the use of major hubs at the origin. Major hubs are in fact cheaper, especially those associated with the SSLC group. All three solutions predilect the use of flights, as those are cheaper options than vehicles. We also observe a significant drop in variation of airlines used: only two (the most cost-efficient) are chosen. The MNTS solution shows a significant reduction of transits: this is trivial, since more transits imply additional costs. When introducing stochasticity, however, this rule of thumb does not always apply: we discuss this in the next section.

Balanced

In the balanced setting, we observe an improvement for the risk and costs and a deterioration in the transit time, for all three solution methods. This is not surprising, since all three alter an initial, transit-time efficient solution by putting more emphasis on other, contrasting objectives. Again, all three solutions show preference for using bigger hubs. Major hubs are cheaper and thus favor the cost objective; at the same time they have more (direct) flight options, thereby reducing the cumulative risk. The MNTS solutions yet again reduces the number of transits, for the same reasons as aforementioned. Overall, this is the setting where the three solutions score most similarly.

Weighted

For the weighted model, we observe a similar trend as in the balanced one when it comes to the objective values. The MNTS solution outperforms the others, as it deteriorates the transit time slightly less and has better improvements on the other objectives. If we compare the routes to the balanced ones, we see an increased usage of minor hubs. This discerns from the fact that more emphasis is put on the time objective and that minor hubs can speed up shipments because of their proximity and lower service times. An interesting observation regards the different use of those hubs compared to the time-greedy model. The latter uses minor hubs for direct flights; in this setting, most minor hubs are used in combination with a transit at a major hub. This sounds counterintuitive, but it can be a beneficial move. On one hand, the export procedure at the origin yields less time, as the first-mile leg. On the other hand, flying with a transit at certain major hubs can be cumulatively less or at least as risky than flying between a minor and a medium/minor hub directly. Additionally, direct flights between minor and medium hubs for this particular lane are more expensive, as they are not operated by SSLC'S usually contracted airline (which we omit), making it more cost-efficient to use it with an extra flight and transit. This is clearly showed by the MNTS solution's transits figure: this solution type tendentially adds transits to the initial route, which is the contrary of what we observe in other settings.

5.2.3 Simheuristic solution performance

In this section, we present the results of the simheuristic, hence the effect of stochasticity on the proposed solutions. For the evaluation, we use the same three tabu search variants as in the deterministic analysis. Since the observed results of the M1 and M2 tabu search procedures are yet again alike, we only compare one of the two versions with the multiple neighborhood tabu search, for sake of conciseness. Henceforth, we refer to the former as tabu search.

Introducing stochasticity

We first compare the average outcomes of the deterministic and stochastic solutions per model type. To do so, we first convert the deterministic risk objective, as this only represents a summation of risk factors, whereas the stochastic objective measures the realized average total delay, making them not directly comparable. For this conversion, we simply take the deterministic solution's routes and sum the expected delays of all respective carriers and nodes used. Table 23 presents the comparison summary. In it, we show the percentage of instances for which the deterministic and stochastic solutions yield different routes. Furthermore, we attach the average KPI values (time and delay in hours, cost in euros) of the deterministic solutions and the relative change of the stochastic ones.

Table 23: Performance comparison between the deterministic and stochastic solutions. Time and delay are expressed in hours, cost in euros. The route change metric indicates the percentage of stochastic routes differing from the deterministic ones.

Tabu search							
		Deterministic objective values			% Change		
Model configuration	%Route change	Time	Delay	Cost	Time	Delay	Cost
Time-greedy	61.22%	27.19	9.46	964.43	-12.17%	238.91%	62.91%
Risk-greedy	100.00%	80.12	5.68	6135.73	-32.01%	193.31%	-35.71%
Cost-greedy	34.69%	51.29	9.12	452.00	5.77%	104.47%	19.26%
Balanced	24.49%	38.68	9.11	452.94	20.48%	155.37%	20.68%
Weighted	34.69%	35.01	8.99	468.45	-13.55%	137.05%	32.04%
Multiple neighborhoods tabu search							
		Deterministic objective values			% Change		
Model configuration	%Route change	Time	Delay	Cost	Time	Delay	Cost
Time-greedy	65.31%	26.66	9.08	1012.41	-11.67%	281.55%	74.06%
Risk-greedy	100.00%	72.01	5.54	6896.57	-4.72%	288.44%	-66.17%
Cost-greedy	42.86%	43.56	8.82	432.04	82.77%	141.64%	18.56%
Balanced	36.73%	38.97	8.88	432.78	16.75%	221.30%	15.91%
Weighted	59.18%	35.68	8.98	444.41	-15.28%	172.43%	17.41%

The most evident figure is that, in all models, both solution types increase the average total delay substantially, compared to the deterministic solution. In general, we would expect to observe an increase in the delay, since every solution's average value includes a few extreme delay cases, which consequently lift its value. However, this increase is bigger than originally thought of. This is attributable to a major shortcoming of our model design, which we already notice in Section 5.2.2 . When taking risk of delay into account, we only consider individual carriers or airports, neglecting the risk coming from too tightly planned transits. When examining all solutions, we see that this aspect has a bigger impact on the route's timeliness than the other risk sources. Especially for the time-greedy model, where transits are planned as tight as possible as to minimize waiting time, major delays are incurred because of the limited slack. Another figure confirming this finding is the 100% change of routes for the risk-greedy models. This tells us that avoiding risky airports or carriers has limited impact on reducing the route's expected delay.

To analyze the reasons behind the differences showed in Table 23 we inspect the routing decisions taken with and without stochasticity. In Table 24 we show a comparison between the deterministic TS (D-TS), stochastic TS (S-TS), deterministic MNTS (D-MNTS) and stochastic MNTS (S-MNTS) solutions. Again, we indicate the percentage of routes using either major, medium or minor airports as origin and/or destination and the preferred carrier. To further put these observations into perspective, we also include the summary of the historical routes analyzed during the context analysis, which we refer to as current in the first row in Table 24. Finally, in its last column, we show the percentage of generated routes per solution type which uses the same combination of origin and destination airports compared to the current ones, and hence would be grouped under the same lane.

An interesting observation from Table 23 is that the cost-greedy solutions show the lowest delay deteriorations, whereas the models which do not consider costs at all have the highest. This is no coincidence. In fact, we learn that in the current design minimizing the total costs indirectly reduces delay between transits. The reason is that whenever a transit is missed, additional costs are incurred to re-book the next available option. Additionally, there is an increased chance of incurring delivery penalties if the shipment ultimately arrives late at destination. To avoid the latter two, solutions which optimize the cost objective tend unconsciously to prefer routes with more time between transits, hence less likely to miss a connection. Related to this observation is another interesting model behavior introduced by randomness, namely the tendence of the MNTS, cost-optimizing solutions to add transits to the initial route. This is opposite to what we observe in a deterministic setting: the solution eliminates transits from the original route and takes direct flight instead. In this case however, there is a chance to miss a flight and thereby incur re-booking and penalty costs. For some options which have direct but riskier flights, it is then overall cheaper to choose a more reliable route, for example with transit at a major hub. Major hubs turn out to be advantageous for both a route's reliability as costs: first of all, they are cheaper than medium or minor hubs. But most importantly, they provide with more flight options, thus mitigating the effect of a parcel missing its flight.

An increase in delay paired with a decrease in transit time seems counterintuitive at first: we observe this especially in the time-greedy solutions. This can be explained by looking at the number and type of routes changed relative to their deterministic counterpart. In almost all of them, both solutions replace two major hubs with a direct flight with two minor or medium hubs with either a direct flight or a driving option. As explained in Section 5.2.2 , minor and medium hubs generally have shorter service times and can be favorable due to fast flying options, making them more time efficient. The number of outgoing options is, however, both limited as less frequent and thus overall riskier; therefore, relatively short transit times are compensated by more frequent delays. As a consequence, the simheuristic tends to predilect riskier routes, thus with more delay, but with significant potential gains in transit time. This fact also motivates the increased usage of vehicles. The shipments operated on these routes are more likely to miss a connection, due to the tightly planned transits. At smaller hubs this becomes problematic, as they have to wait longer for the next flight. Vehicles are not bound to scheduled departure times, and can therefore mitigate the delays, becoming thus faster options than aircraft. Altogether, these routing decisions also have a significant impact on the total costs, which have the most significant increase relative to the other presented solutions.

Interestingly, also the MNTS weighted solution shows improvement in transit time and deterioration of delay. The latter is less significant than in the time-greedy ones, because this solution finds a smart speed-reliability balance. For some routes, it uses minor or medium hubs as origins, reducing the shipment's initial service time, and flies to two subsequent major hubs. The transiting one is a well-connected hub, meaning that missing the first flight can mostly still be recovered by taking the second one of the day, therefore alleviating the effect on the route's overall delay. This is also reflected in the total costs, which deteriorate less notably than in the time-greedy model.

Table 24: Comparison of the routing decisions of the deterministic and stochastic solutions and historical routes (benchmark). D-TS = deterministic tabu search; S-TS = stochastic tabu search; D-MNTS = deterministic multiple neighborhood tabu search; S-MNTS = stochastic deterministic multiple neighborhood tabu search.

Benchmark									
Solution type	Preferred origin			Preferred destination			Preferred carrier type	Number of transits	% Original lane
	MA	ME	MI	MA	ME	MI			
Current	100%	0%	0%	100%	0%	0	Flight	8%	100%
Time-greedy									
Solution type	Preferred origin			Preferred destination			Preferred carrier type	Number of transits	% Original lane
	MA	ME	MI	MA	ME	MI			
D-TS	76%	4%	20%	63%	22%	14%	Flight	27%	4%
S-TS	51%	10%	39%	59%	20%	20%	Flight	27%	6%
D-MNTS	76%	8%	16%	55%	37%	8%	Flight	18%	2%
S-MNTS	37%	16%	47%	59%	24%	17%	Mixed	29%	8%
Risk-greedy									
Solution type	Preferred origin			Preferred destination			Preferred carrier type	Number of transits	% Original lane
	MA	ME	MI	MA	ME	MI			
D-TS	0%	100%	0%	0%	100%	0%	Vehicle	27%	0%
S-TS	73%	25%	2%	61%	19%	20%	Flight	27%	0%
D-MNTS	0%	100%	0%	0%	100%	0%	Vehicle	0%	0%
S-MNTS	20%	64%	16%	59%	14%	27%	Flight	78%	8%
Cost-greedy									
Solution type	Preferred origin			Preferred destination			Preferred carrier type	Number of transits	% Original lane
	MA	ME	MI	MA	ME	MI			
D-TS	94%	6%	0%	67%	14%	18%	Flight	27%	55%
S-TS	96%	2%	2%	66%	14%	20%	Flight	27%	55%
D-MNTS	94%	6%	0%	67%	14%	18%	Flight	6%	55%
S-MNTS	88%	10%	2%	65%	14%	21%	Flight	37%	47%
Balanced									
Solution type	Preferred origin			Preferred destination			Preferred carrier type	Number of transits	% Original lane
	MA	ME	MI	MA	ME	MI			
D-TS	94%	6%	0%	69%	14%	16%	Flight	27%	57%
S-TS	90%	6%	4%	69%	2%	29%	Flight	27%	55%
D-MNTS	94%	6%	0%	67%	14%	18%	Flight	12%	61%
S-MNTS	82%	16%	2%	61%	14%	25%	Flight	39%	41%
Weighted									
Solution type	Preferred origin			Preferred destination			Preferred carrier type	Number of transits	% Original lane
	MA	ME	MI	MA	ME	MI			
D-TS	88%	6%	6%	69%	14%	16%	Flight	27%	53%
S-TS	73%	15%	12%	76%	12%	12%	Flight	27%	57%
D-MNTS	90%	6%	4%	69%	14%	16%	Flight	37%	47%
S-MNTS	74%	18%	8%	66%	12%	22%	Flight	49%	37%

Best-performing solution and relative improvement

Having included randomness in the solutions' evaluation, we can make inferences on which one has the best potential to improve SSLC'S current situation. In Appendix I, we show the detailed results of all simheuristic experiments; here we provide a summary in Table 25. We show the benchmark's original KPI performance and the relative change per solution type.

Table 25: KPI comparison between the simheuristic solutions and the benchmark. The average transit time, total delay and delivery delay are expressed in hours, costs in euros and the run time in seconds. The benchmark's run time is an estimate.

Benchmark						
Solution type	Transit time	Transit time CV	Delivery delay	Total delay	Total cost	Run time
Current	41.5	0.5	3.2	26.9	647.9	120
Time-greedy						
Solution type	Transit time	Transit time CV	Delivery delay	Total delay	Total cost	Run time
TS	-44.12%	-36.22%	107.13%	-1.73%	119.79%	713.69
MNTS	-45.08%	-51.50%	102.26%	3.51%	142.81%	2447.96
Risk-greedy						
Solution type	Transit time	Transit time CV	Delivery delay	Total delay	Total cost	Run time
TS	36.82%	-1.11%	115.81%	-37.94%	394.26%	638.82
MNTS	65.69%	-12.97%	62.89%	-20.07%	263.73%	2338.08
Cost-greedy						
Solution type	Transit time	Transit time CV	Delivery delay	Total delay	Total cost	Run time
TS	21.58%	36.70%	74.93%	-36.86%	-17.22%	751.8
MNTS	48.59%	15.17%	62.22%	-22.52%	-22.35%	2631.84
Balanced						
Solution type	Transit time	Transit time CV	Delivery delay	Total delay	Total cost	Run time
TS	-1.64%	13.30%	50.61%	-19.71%	-14.71%	753.47
MNTS	-3.10%	4.13%	25.30%	-18.19%	-22.80%	2637.15
Weighted						
Solution type	Transit time	Transit time CV	Delivery delay	Total delay	Total cost	Run time
TS	-29.85%	-12.32%	4.42%	-23.83%	-4.89%	664.04
MNTS	-29.15%	-13.73%	-5.62%	-18.81%	-20.31%	2357.34

In comparison with the results presented in Table 21, we see that the improvements brought by the two solution approaches are more similar for all models. In most cases the MNTS model is slightly better, but the difference when introducing stochasticity becomes less evident. In order to compare the relative magnitude of the solutions' improvements, we normalize the respective KPIs, and aggregate them using the same weights as elicited with the AHP method in the context analysis (Table 7). Figure 23 shows this comparison graphically. From this figure, we can see two solutions which overall improve the current situation, namely the two tabu search variants solving the problem with the weighted objectives configuration. The MNTS solution performs slightly better, which is in line with what we expect. Furthermore, we observe that both time-greedy and balanced solutions approach the current situation. They both improve the overall transit time metrics, but worsen the delay and costs, making them non-improving. The remaining solutions perform worse than the current situation.

If we compare the MNTS weighted solution with the benchmark, we observe that for only 37% of the corresponding instances the airport choice coincides. In fact, the solution displays significant more variation in the types of chosen hubs. It also shows a notable increase in the number of transits (49% of the orders compared to the benchmark's 8%). This is attributable to the speed-reliability balance the solution seeks by combining the usage of minor/medium and major hubs on a route, as previously discussed. Interestingly, this variation also translates to the different usage of airlines. In fact, the solution uses five distinguished airlines almost evenly, whereas 92% of the benchmark routes predilect one of those five. But most interestingly, the solution combines different airlines on almost half of its routes with transits. This is remarkable, as the current system imposes to use only associated airlines (e.g. all belonging to the SSLC group) on a single route and not mix different groups, as SSLC believes that combining them increases the likelihood of miscommunication and, consequently, of delay.

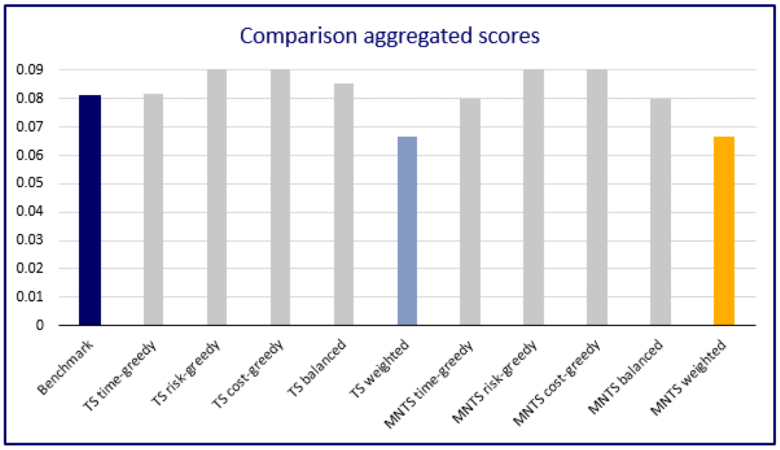


Figure 23: Comparison of the solutions' aggregated scores per model type. The current lane's performance (dark blue) is improved by the TS (light blue) and MNTS (yellow) solutions applied on the weighted model. The hybrid solution has the lowest and thereby best score.

To conclude the analysis, we check the relative improvement compared to the other lanes' aggregated score. We substitute the benchmark's original KPI values with the ones of the MNTS weighted solution and recategorize the lane with the ABC method introduced in Chapter 2. Figure 24 graphically shows the results. We see that the solution's improvement is not sufficient to lift the examined lane to the best performing category; nevertheless, we acknowledge it still represents a significant advance in the right direction.

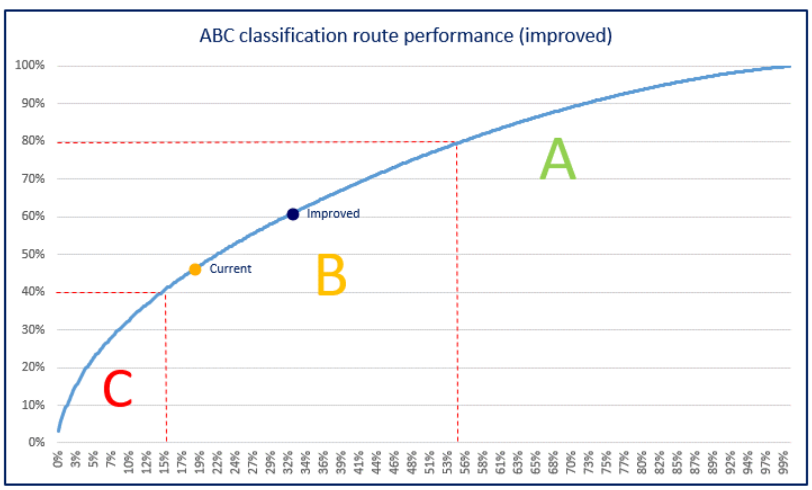


Figure 24: Performance gap between the current situation and the proposed solution. The blue dot indicates the performance of the proposed solution compared to other lanes from the years 2020-2022, the yellow dot indicates the actual performance of the analyzed test lane. The red-dotted lines delimit the three performance classes.

5.2.4 Stochastic evaluation

Having examined the solution's average performance, we proceed on analyzing its robustness, given the problem's stochastic nature. For conciseness, we examine only two problem instances: they represent orders with the most common pickup and delivery points and are thereby grossly representative of the 49 instances solved in the previous section. We call them instance 1 and instance 2.

Below we present the results of their final evaluation, where we measure the cumulative impact of the generated simulation scenarios on each solution. In each boxplot, we indicate the results yielded by the time-greedy model with Sol1, risk-greedy with Sol2, cost-greedy with Sol3, balanced with Sol4 and weighted with Sol5.

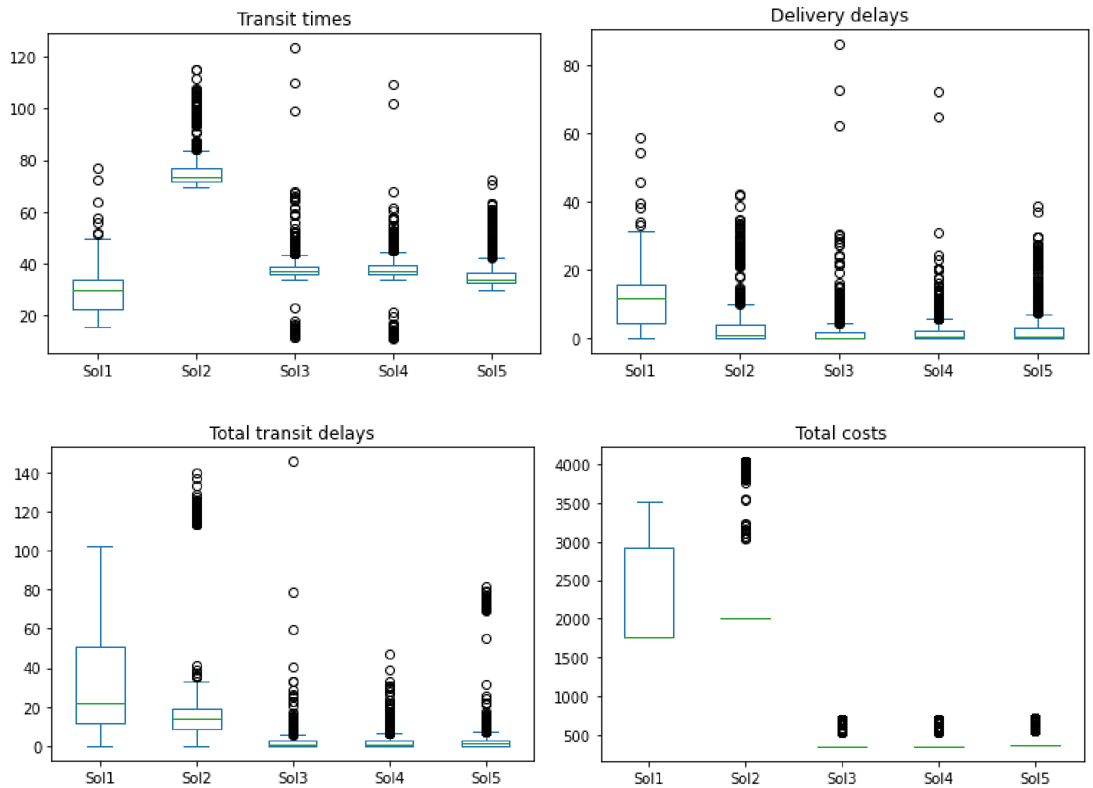


Figure 25: Boxplots of the solutions' stochastic evaluation for instance 1. The four plots display the spread of objective values for the time-greedy (Sol1), risk-greedy (Sol 2), cost-greedy (Sol 3), balanced (Sol 4) and weighted (Sol 5) solutions. The green line represents the median value, circle points are outliers.

Figure 25 shows the results per KPI of the sim-evaluation of each of the five solutions to problem instance 1. In terms of generated routes, all solutions yield a different result except for the cost-greedy and balanced, which have the same one. From the results we learn that the time- and risk-greedy solution types are more sensitive to randomness, and thus relatively unstable. This is reflected both in the larger spread of the transit times, as in the delivery delays and total transit delays. Furthermore, the time-greedy solution is more likely to incur penalty costs (this happens in 49% of the generated scenarios), which can be noted by observing the cost figure. On the contrary, the cost-greedy, balanced and weighted solutions are far more risk-aversive, therefore stabler. Due to the less tightly planned transits, the delays are overall lower as is the variance in total costs. Mostly, the observations we make for Figure 25 are valid for most of the problem instances solved with the simheuristic.

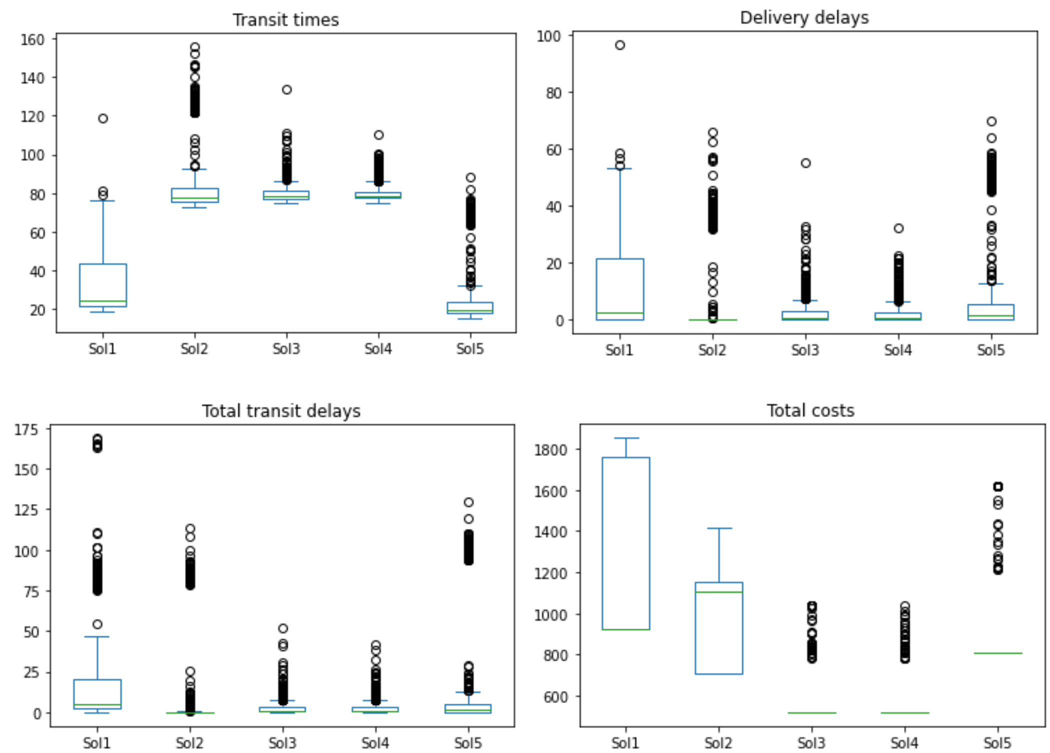


Figure 26: Boxplots of the solutions' stochastic evaluation for instance 2. The four plots display the spread of objective values for the time-greedy (Sol1), risk-greedy (Sol 2), cost-greedy (Sol 3), balanced (Sol 4) and weighted (Sol 5) solutions. The green line represents the median value, circle points are outliers.

Figure 26 presents the evaluation of the results for instance 2. Like in instance 1, the cost-greedy and balanced solutions present the same route, while the remaining solutions have different routes. In contrast with what we observe above, we see that the weighted solution presents with a faster option than the time-greedy one. We also see that the risk-aversive yields a fairly reliable option, whereas the opposite is true in most cases. Overall, in comparison with instance 1, we observe more variability in all solutions results, with larger differences between the first and third quartiles and more outliers.

In general, both solved problem instances show numerous outliers; furthermore, most results are not distributed uniformly, but form rather asymmetric box plots. If we analyze this further and compare the results' average values with their mode, we see observe a clear skewness. In Table 26 we show that, for each model type, the percentual difference between the results mean and mode is significant. This means that rather extreme scenarios are generated by the simulation, which deteriorate the solutions' average performance significantly, whereas the most common KPI values are far lower. This has implications for the outcomes of Section 5.2.3 . Given that in our design we select solutions over others based on their averages, this means that we potentially predilect solutions with less (extreme) outliers over solutions that, most of the times, yield better values but are affected by more extreme scenarios. Finally, if we compare the solutions average KPIs with historical data, we find confirmation to the considerations made above. For both instances, the time-greedy and balanced models yield the same route as the original orders. By comparing their average and mode values with the actual order metrics, we learn that the former overestimates time values by 76.9% on average, whereas the latter has a smaller gap, with an average difference of 29.3% from the original values. This figure tells us, hence, that the delay distributions used in the simulation are tendentially pessimistic. If we look at the cost gap, the solutions average is about 30.4% lower than the actual values, whereas the mode lies on a gap of 32.4%. This is however not surprising, as the cost estimations used in the model are entirely realistic on purpose.

Table 26: For instance 1 (left) and 2 (right), we compare the percentual difference between each KPI's average value and mode. On the bottom part of the table, we report the average gap per KPI type. Instance 2 shows the highest gap and thus the most skewness of results.

	Sol 1							
	Intance 1				Intance 2			
	Transit time	Delivery delay	Total delay	Total costs	Transit time	Delivery delay	Total delay	Total costs
Mean	41.90	10.85	31.16	2357.72	51.31	8.43	23.61	1175.59
Mode	31.40	0.00	52.20	2921.98	21.29	0.00	0.00	926.00
% Difference	25.06%	100.00%	-67.52%	-23.93%	58.51%	100.00%	100.00%	21.23%
	Sol 2							
	Instance 1				Instance 2			
	Transit time	Delivery delay	Total delay	Total costs	Transit time	Delivery delay	Total delay	Total costs
Mean	89.70	4.37	23.28	2241.93	107.63	6.89	15.30	1047.98
Mode	71.60	0.00	8.20	2016.00	75.80	0.00	0.08	708.00
% Difference	20.18%	100.00%	64.78%	10.08%	29.57%	100.00%	99.48%	32.44%
	Sol 3							
	Instance 1				Instance 2			
	Transit time	Delivery delay	Total delay	Total costs	Transit time	Delivery delay	Total delay	Total costs
Mean	51.07	1.83	2.60	364.87	100.90	2.20	2.60	533.49
Mode	35.80	0.00	0.00	351.00	77.30	0.00	0.00	521.00
% Difference	29.90%	100.00%	100.00%	3.80%	23.39%	100.00%	100.00%	2.34%
	Sol 4							
	Instance 1				Instance 2			
	Transit time	Delivery delay	Total delay	Total costs	Transit time	Delivery delay	Total delay	Total costs
Mean	51.20	1.99	2.57	364.41	100.86	2.09	2.73	531.90
Mode	35.80	0.00	0.00	351.00	77.30	0.00	0.00	521.00
% Difference	30.08%	100.00%	100.00%	3.68%	23.36%	100.00%	100.00%	2.05%
	Sol 5							
	Instance 1				Instance 2			
	Transit time	Delivery delay	Total delay	Total costs	Transit time	Delivery delay	Total delay	Total costs
Mean	48.90	2.90	5.48	386.85	47.79	8.68	16.11	934.11
Mode	32.60	0.00	0.00	363.00	18.60	0.00	0.00	809.00
% Difference	33.33%	100.00%	100.00%	6.17%	61.08%	100.00%	100.00%	13.39%
	Overall difference							
Average	27.12%	100.00%	59.45%	-0.04%	43.14%	100.00%	99.90%	14.29%

5.2.5 Sensitivity analysis

This section concludes our experiments with a sensitivity analysis. We solve three problem instances using the cost-greedy, balanced and weighted model configurations and with the three penalty systems defined in Section 5.1 . Table 27 shows the results. For each model configuration, we compare the average KPI values of the three instances solved with the benchmark (hence the penalty policy we define in Section 4.5) and the strict, relaxed and moderate policies. We also show the percentage of routes differing from the original ones, the percentage of simulation instances incurring penalties and the type of change applied per policy type.

Table 27: Experimental results of the sensitivity analysis. For each model type, we compare the average KPIs of the three instances solved with the benchmark penalty system and the alternatives presented in Section 5.1. Additionally, we show the percentage of changed routes, the percentage of simulation scenarios incurring penalties and the route's change.

Cost-greedy							
Policy	Time	Delivery delay	Total transit delay	Costs	%Change	%Penalties	Route change
Benchmark	62.54	18.83	90.31	574.66	N.A.	41.33%	N.A.
Strict	12.86%	-21.50%	-6.27%	-9.86%	33.33%	55.20%	Add MA transit
Relaxed	1.72%	-21.46%	-6.10%	-10.85%	33.33%	32.00%	Add MA transit
Moderate	1.64%	-21.94%	-6.63%	-14.10%	33.33%	48.47%	Add MA transit
Balanced							
Policy	Time	Delivery delay	Total transit delay	Costs	%Change	%Penalties	Route change
Benchmark	39.15	7.95	67.04	432.69	N.A.	53.00%	N.A.
Strict	-11.92%	4.40%	54.85%	50.19%	66.67%	55.27%	Add MA transit
Relaxed	-13.23%	-3.20%	15.44%	8.68%	66.67%	46.80%	Remove MA transit
Moderate	-11.78%	4.35%	54.36%	20.22%	66.67%	55.30%	Add MA transit
Weighted							
Policy	Time	Delivery delay	Total transit delay	Costs	%Change	%Penalties	Route change
Benchmark	29.94	7.30	75.72	554.65	N.A.	52.97%	N.A.
Strict	4.89%	-9.79%	3.16%	37.24%	66.67%	55.07%	Add MA transit
Relaxed	-0.56%	-8.65%	4.60%	10.81%	66.67%	46.00%	Remove MA transit
Moderate	3.95%	-10.97%	1.57%	10.40%	66.67%	54.47%	Add MA transit

From the experimental results, we observe a fundamental difference in routing behavior between the cost-greedy and the remaining two solutions. Firstly, the former tends to keep the routes unaltered, regardless of the policy. Indeed, it shows the lowest change percentage of the three. This is because the generated routes are already the ones yielding the lowest costs: we can see this if we look at the benchmark's cost figure, which is relatively low considered its elevated percentage of instances incurring penalties. This means that in most cases, altering the routes to avoid penalties from stricter policies (strict and moderate) does not outweigh the additional costs incurred by the route change. We observe only one instance in which, for all three policies, the solution finds a superior route, explaining the average improvement of almost all KPI values. This is however rather an exception and thereby not representative of how the different penalty systems influence this model's behavior.

The remaining two models appear to be more reactive to the change of policy. For both, 2 out of 3 problem instances are solved with a new route. Furthermore, both show a similar way of reacting to either policy. For example, both tendentially add a major hub transit to the benchmark route with the strict and moderate policies. In the latter two, we start penalizing earlier (from 6 hours time difference) than in the benchmark (here we start at 12). This is reflected in the increase of percentage of simulation instances incurring penalties. Since the costs deteriorate anyway, both models appear to look for a way of compensating by improving the other objectives. In the balanced model, the major hub is added because it has flights that are better connected with the destination airport, and thus foster the overall transit time. The weighted model, on the contrary, improves the delivery delay metrics, as it finds slower, yet more reliable flight connections.

When we solve the instances with the relaxed policy (start penalizing at 24 hours deviation, up until 48), for both models a transit at a major hub is removed, leading to a slight deterioration in the cost metric and total transit delay. Since less penalties are incurred, this initially sounds counterintuitive. If we look at the instances where the route is changed, we learn that the departing and arriving airports are major hubs as well, with a direct flight between them. The direct flight is however more expensive than the options with a transit, as it operated by a carrier seldomly used by SSLC. By reducing the frequency of incurred penalties, the model apparently prefers taking the direct option, which is cumulatively more expensive, but also faster. Overall, we can conclude that the changes in penalty systems have more impact on the routing decisions on the balanced and weighed models than the cost-greedy. In both cases where more or less penalties are incurred, the former models try to adapt the compensate by improving the transit time and delay metrics, respectively.

5.3 Conclusion

In this chapter we answer the question: “How does the proposed solution perform compared to the current routing algorithm?”. We do so by first setting up and executing five types of experiments. First, we tune the proposed solutions to find their optimal settings. Next, we solve the problem deterministically. We do so by generating 49 instances using historical order data. By comparing the results, we observe that the two tabu search algorithms used yield identical solutions, whereas the MNTS generates different routes. Compared to the MIP model, all three algorithms yield similar results. Furthermore, all three show significant improvement compared to the initial solution, demonstrating they can be valid candidates to improve the current routing system. For the following experiments, we incorporate the three solution methods in the simheuristic procedure described in Section 4.5 .

By introducing randomness we observe an overall deterioration of the deterministic KPIs, most notably observable in the delay and cost metrics. Whereas we expect to observe such development, the delay deterioration is most significant, given the current model’s design does not account for the risk of delay discerning from too tightly planned transits. This is particularly observable in the time-greedy solutions, where transits are reduced to minimize the waiting time. To our surprise, the models accounting for the cost objective partially mitigate this effect, as routes with tight transits are avoided as to reduce the frequency of re-booking and penalty costs. The risk-aversive solutions yield the worst results, as they deteriorate almost all KPIs and generate routes which do not make sense from a practical point of view. We observe that the time-greedy solutions improve the transit time but deteriorate the delay and costs KPIs as they generally yield riskier routes. The balanced and weighted solutions provide the best results. The former solutions are more cost-oriented and prefer therefore the usage of major hubs and two specific cost-efficient airlines; the latter combine the use of minor and medium hubs with transit at majors and use five different airlines in combination. This fosters the routes’ speed, although it is more expensive than the balanced ones.

By using the KPI weights obtained in Chapter 2 we compare the results with SSLC’s current routes. We conclude that the MNTS weighted solution yields the best results, improving them by 17.7% on the aggregated KPI score. Having analyzed the solutions’ performance, we examine their robustness. By inspecting box plots of the simulated scenarios, we observe that the time- and risk-greedy solutions present generally the most variability, whereas the remainder solutions are more stable. Overall, the generated time values present high outliers, which influence the average outcomes. Finally, we analyze how the penalization system for deviating from the delivery time affects routing decisions in the models optimizing the cost objective. Whereas the change of policy yields little change in the cost-greedy model, it affects the balanced and weighted ones. In general, both solutions tend to compensate the deterioration in the costs objectives by improving either transit time or delay metrics, regardless of the type of policy used. To this end, they either add or remove a transit at a major airport to their initial route.

Chapter 6 – Conclusions and recommendations

This final chapter concludes the research. Section 6.1 summarizes the research findings, presents our conclusions and contributions to practice and science. In Section 6.2 we provide recommendations and in Section 6.3 we discuss this research's limitations and give suggestions for potential future work.

6.1 Conclusions

In this research, we explore approaches within the domain of mathematical programming to improve the current calculation of Express airfreight routes at SSLC. We thereby formulate and answer the main research question: "How can mathematical programming support route calculation at SSLC as to improve the performance of their routes?".

Following the context analysis, we identified several limitations to the current routing process, some related specifically to the logic behind the routing system. In fact, the latter significantly limits the number and quality of its routes, as it selects the origin and destination airports solely based on geographical proximity to the shipment's pickup and delivery points and is incapable of comparing routes having different combinations of starting and ending airports. Most importantly, the current system only considers routes which arrive the earliest at delivery, neglecting other relevant aspects like likelihood of incurring delays, operating costs and the network's stochastic nature in general. Our solution overcomes these issues, as it can compare multiple combinations of origins and destinations, returning routes with the best options. It calculates routes accounting for the transit time, expected delay and total operating costs, returning cumulatively better options than the current ones. Finally, our method also accounts for stochasticity, which enables CS agents to be more aware of the robustness of its generated options. However, the fact that SSLC does not own the flight data necessary for the route calculation remains an issue, as it prevents the company from being able to redesign the process in a more efficient way. Whereas we cannot comment on whether the costs of eventual data ownership would be outweighed by the benefits of such process redesign, we demonstrate that our solution yields clear benefits from an operational point of view.

To improve the current situation, we modelled the problem at hand as a Multi-Objective Multimodal Route Choice Problem, where we minimize the total transit time, risk of delay and cost. We modelled the scheduled flights' departures with hard-time windows and penalized both late and early arrival at the delivery point with soft time-windows. To account for all three objectives, we aggregated them linearly in a single objective function, using weights to specify their relative importance. To learn how favoring either objective influences routing decisions, we solved the model with five different weight settings. To solve the problem, we used three variants of the tabu search algorithm: one storing moves (TS-M1), one storing entire solutions (TS-M2) and a hybrid one using multiple neighborhood operators (MNTS). To initiate the algorithms, we input an initial route, which we generated following the logic used by the current system.

We executed five types of experiments. With the tuning experiments we obtained the optimal settings to run our solutions on each model. Next, by solving the problem deterministically, we proved the quality of each algorithm compared to an exact solution to the MIP and obtained an indication of the degree to which the three solution methods improve their input initial solution. Hence, we investigated their potential to improve the current routing system's algorithm. By solving the problem with the simheuristic, we obtained the main research results. We learned which solution approach performs best, to what degree it cumulatively improves the actual Express routes and what routing decisions yield such improvement. By analyzing the spread of results obtained within the final simulation step of our simheuristic, we evaluated the role of randomness within the obtained outcomes. Finally, with the sensitivity analysis we explored how a change in penalizing policies affects the routing behavior of solutions accounting for the cost objective. The main outcomes of our experiments are summarized in Table 28.

Table 28: Summary of the main experimental outcomes.

Experiment number	Experiment	Main outcomes
1	Model tuning	• Optimal model settings • Departure constraint of current system can be dropped
2	Deterministic analysis	• Compared to the MIP solutions, all three tabu search methods provide with overall similar results in less time • All three tabu search methods improve the current routing algorithm on all modeling configurations
3	Simheuristics	• Weighted solutions yield cumualtively better routes than the current ones. • Time-greedy and balanced solutions yield similar aggregated performance compared to the current routes • Cost-greedy solutions yields cheaper but slower options than the current routes • Risk-greedy solutions perform significantly worse than the current routes and are overall inoperable
4	Stochastic evaluation	• The generated solution outcomes are largely influenced by delay outliers • The results overestimate reality in terms of average delay and underestimate the costs
5	Sensitivity analysis	• A change in penalty costs does not seem to affect the cost-greedy setting in a significant way • Penalizing delivery deviations at lower or higher thresholds yields similar effects on the balanced and weighted models; both add transits at a major hub to improve the other objectives

From the tuning experiments, we obtained the model settings for the subsequent experiments. We learned that enforcing routes to have options departing from the origin within 24 hours from the shipment's availability time is not beneficial, therefore we relaxed this model constraint. Using 49 historical orders, we estimated model parameters directly from the data and generated problem instances to test our model. By solving the problem deterministically, we first observed that, for all model configurations, solving all instances optimally is not possible within a time limit of five minutes. Furthermore, all three algorithms perform similarly to the MIP model and require less solving time, thereby justifying our choice to prefer them to exact methods. Additionally, both TS and MNTS methods improve the initial solution significantly. From this we can conclude that the proposed tabu search variants are suited candidates to improve SSLC's current routing algorithm. Next, we incorporated stochasticity in the simheuristics and solved the same 49 instances. Table 29 summarizes our findings.

Table 29: Summary of the routing policies for each of the five solved model types.

Solution	Routing policy	Advantages	Disadvantages
Time-greedy	• Use more regional and medium-sized hubs and direct options • Use vehicles to avoid long waiting times for the next flight	• Significantly faster routes	• High risk of delay • Frequent rebookings and penalties • Overall high operating costs
Risk-greedy	• Choose combination of least risky carriers and airports	• Partly reduces carrier- and partner-bound delay	• Worst performance on all KPIs • Practically inoperable routes
Cost-greedy	• Use major hubs with frequent and reliable flights • Use only cost-efficient airlines	• Reduces transit delay as to avoid extra rebooking costs • Best cost reduction	• Slower routes
Balanced	• Use major hubs at departure • Choose destination airport to reduce last-mile leg	• Overall good KPI performance	• Deterioration of transit time at the expense of cheaper routes
Weighted	• Choose minor or medium hub as starting airport • Transit at a well-connected major hub • Combine multiple different airlines on same route	• More emphasis on transit time and reliability	• More expensive than balanced and cost-greedy routes

Overall, we learned that time-greedy models tend to favor the usage of minor hubs as starting airports. Those hubs are generally less burdened by high cargo volumes and therefore have the main advantage of having shorter service times for the shipment's export preparation. Furthermore, they are closer to most generated pickup points and have fast direct flight options, making them interesting for transit time minimization. Those airports are however riskier: flights are less frequent, meaning that delays can quickly escalate. To mitigate this, most routes make use of driving options. As trucks are not constrained by a scheduled departure time, they can be used to mitigate the delay of a shipment which misses its flight. Therefore, the use of driving options between airports as a mean of delay recovery can have positive effects on a shipment's transit time. Those options are, however, notably expensive. In short, the usage of direct options between minor hubs and including vehicles as a mean for transportation overall results in the fastest routes. This comes however at the expense of high average delay and operating costs (respectively +105% and +132% w.r.t the current routes) making them less interesting for regular shipments.

The risk-greedy models turned out to perform worst. Most generated routes are inoperable in practice, showing significant deterioration on the transit time and cost metrics. Delays are minimized to a limited extent, showing that accounting for carrier- and airport-bound risks is an effective yet not exhaustive method to prevent delay. All in all, this model is not usable by SSLC. For cost-greedy options, the solutions predilected the use of major hubs and flying options, as they are overall cheaper. The model also showed a low variation in the utilized airlines, as it predilects employing cheaper airlines with a flight extra than more expensive ones with direct options. Paradoxically, these solutions tend to choose routes with an additional transit, mostly at a major hub. This is motivated by the fact that these routes are more reliable and thus help preventing missing flights (which result in rebooking fees) and arriving late at the consignee (resulting in penalty costs). Additionally, we learned that cost-greedy routes have more time between transits, as a way of preventing missing a connection. As a consequence, the cost-greedy model indirectly covers a third source of risk which we originally overlook in our design, namely the tightness of planned transits. Therefore, in this model setup, optimizing costs can be beneficial to improve the reliability of a flight route. The major disadvantage, however, of using a cost-centered model is that the chosen options are overall slower (+36% transit time w.r.t. the current routes). Eventually, this kind of routes can become useful for SSLC whenever order requests are received far ahead of the requested delivery moment, as this would enable the company to plan on a longer term and thereby operate slower but cheaper (and more reliable) routes without deviating from the requested transportation moment.

Within the balanced and weighted models we recognized routing decisions discerning from the greedy models: the balanced model predilects the usage of major hubs as origins and destinations as it puts more emphasis on the cost objective, whereas the weighted model uses more minor and medium hubs, to foster speed of delivery. Overall, both solution types provide with good options; given that SSLC generally prefers speed over cost, the weighted solutions are better suited to the company's needs. By comparing the latter's KPIs with the ones of the current routes, and weighing them using the importance weights we elicited from the context analysis, we saw that the weighted solution yields significant improvements on all metrics, thereby outperforming all other solution types. Indeed, it improves the transit time by 29.2%, its coefficient of variation by 13.7%, the delivery and total delays by 5.6% and 18.8% respectively, and the total costs by 20.4%.

If we compare the routes obtained by the weighted solution with their corresponding historical orders, we observe that only 37% of the routes match. In general, the weighted solution shows significantly more variation in its generated routes, as it uses more different hubs and airlines. Although most of the chosen origins are major hubs, this solution showed an increased use of medium and minor hubs (26% of all routes). In contrast with what we observed in the time-greedy solutions, a direct option is never chosen when using these types of airports; instead, a transit at a major one is preferred, from which then a flight to another major hub is taken. In this way the solution finds a balance between speed and reliability. The first airport choice favors the transit time. Since the chosen transit hubs are well-connected to the origins, eventual delays are easily mitigated. This has also a positive impact on the costs, which are higher than in the balanced and cost-greedy solutions but lower than the time- and risk-greedy ones. Additionally, we observed an increased variability of airlines used. The solution uses five different airlines almost evenly, whereas the original routes choose one distinguished airline 92% of the times. In addition, different airlines are used in combination for single routes: this shows that using a more diverse portfolio of air carriers in combination can be advantageous for a route's general performance.

Although we obtained encouraging results, we must acknowledge that the reported KPI values suffer from variability, which results mainly from pessimistic simulation outcomes. As emerges from the stochastic analysis, the distributions of the transit times and delays are mostly skewed, showing significant disparity between the average and mode values. Furthermore, we observe an elevated number of outliers. Having performed simulations of 15000 iterations we feel confident in concluding that this is a consequence of the (too) pessimistic nature of the utilized probability distributions rather than the result of randomness in general. By comparing the simulation results with real data, we find confirmation to this affirmation. Whereas the time values tend to be overestimated, costs seem to be underestimated. Since the costing data used in the model are not accurate, the latter observation is trivial. Therefore, comparisons in terms of cost-efficiency can only be made with fair confidence between the generated solutions, rather than with the current situation.

In conclusion, this research yields multiple contributions. Starting with an academical viewpoint, we contribute to existing literature by providing a comprehensive classification scheme for the MRCP (summarized in Figure 14), which includes all features extracted from related works collected during our literature review. This scheme can be used by researchers as framework to translate a real-life problem to a MRCP model. Furthermore, we enrich the MRCP formulation by Lei et al. (2014) with several additions. First, we incorporate the use of hard time-windows to model departures of scheduled transportations services. Secondly, we provide with a simple, yet effective method to estimate the risk factors used in authors' model, as they do not provide with one. This allows future researchers wanting to use their model for practical research to empirically quantify risk. Third, we correct a mistake by the authors as they do not include (explicit) flow constraints in their model formulation. Additionally, we demonstrate that tabu search can effectively solve the MRCP, which to the best of our knowledge, has not been done before. On top of that, we provide with two equally good methods for using memory. Finally, we add onto the literature on hybrid heuristics, as we demonstrate that our MNTS can effectively solve a MRCP. Our major practical contribution is to provide SSLC with insights into how certain routing choices influence relevant objectives of time, risk and cost. With these insights, the company can improve their operations and provide an even more tailored service to their customers. For example, if a customer requires a faster route, agents can consider offering direct options between two minor hubs, bearing in mind this brings additional risk and significant costs. On the other hand, routes for regular shipments can be steered more towards routing choices from the weighted model, hence departing from minor/medium hubs with transit and arrival at a major one. On a higher level, this research serves as a blueprint for freight forwarders specialized in express logistics who wish to optimize their routes, as it effectively demonstrates a method to do so.

6.2 Recommendations

In this section we provide SSLC with practical recommendations based on the research's results. We start off with the operational ones. One of the most interesting findings of this research is related to the usage of smaller hubs as starting airport to reduce the shipment's drive time and export service time. Whereas we consider this plausible, we also acknowledge that the available timestamps for those airports are significantly less than those for bigger hubs, as the former are operated seldomly in the network. Therefore, the estimated service times might be too optimistic compared to reality. It would be therefore interesting to investigate whether the performance of those minor hubs coincides with what we input to the model in this research. In a similar way, our solution uses different (unrelated) airlines in combination for a single route. SSLC currently avoids this, as it allegedly exposes a route to an increased risk of irregularities. Because of this, we have no data to prove with certainty whether combining airlines can be better or worse. Therefore, we recommend investigating this further by using test shipments with different operating airlines. In the time-greedy solutions, we propose the use of vehicles for mitigating delays caused by shipments missing their fight from a minor (and thus limitedly connected) hub. It is questionable whether the costs of such options would be worth the benefit, besides the fact that SSLC might not have sufficient partner capacity to use this option. Nevertheless, it could be an interesting option to explore for short-haul routes like the one examined within our research. Finally, we recommend trying to relax the $depwithin$ constrain in the API request the system currently sends to the flights scheduler and see whether this indeed benefits the overall quality of results.

Besides providing recommendations for the short term, we also have suggestions for the medium-to-long term on how to improve the process in general. The first one concerns improving the current data quality, as it has a considerable influence on the presented results. Indeed, we recommend to research ways to improve the data collection process, for both sales and tracking datasets. As we show in the stochastic analysis, the estimated probability distributions generate too many outliers, making the average simheuristic results yield overall pessimistic estimations for the service and travel times. These rather extreme values are the result of outliers present in the collected data. To get a more realistic solution, a fraction of those outliers needs to be removed. This is however not possible in the current setup, as we cannot distinguish which ones are caused by exceptional circumstances (e.g. adverse weather conditions, strikes), miscommunication accidents with the customer or measurement errors (caused by either partners or CS agents). Additionally, potentially useful data is not being collected in the current setup. For example, the number of capacity warnings displayed to agents for fully loaded flights is not being stored. This type of data would have been very useful, as it would have allowed us to consider capacity limitations in our model design and hence make it more realistic.

Furthermore, SSLC should also consider acquiring flight schedules data. The carriers used as input to our model all belong to the set of flights which have been used by SSLC in the past. This restricts our results significantly, as there might be more options offering better routes. Most importantly, acquiring the data would enable SSLC to implement its own routing algorithm, following the approach proposed in our research. Besides the benefits demonstrated by our solution, doing so would enable the company to redesign the process we describe in Chapter 2. The validation of results could be done upstream, thus reducing the system's computational inefficiency. Naturally, it is up to the company to investigate whether the costs of such investment would be paid-off by the mentioned benefits.

If SSLC wishes to implement our proposed solution, we recommend to regularly estimate and update the model parameters, using a direct pipeline to the sales and tracking datasets. One of the strengths of the proposed solution is that it uses parameters estimated from actual data, making its outcomes more realistic. To keep this advantage, data must be regularly collected, and datasets updated, ensuring parameters estimations can be kept up to date. Another important step would be to test the hybrid tabu search on a larger problem instance first, and next using the entire Express network. If the algorithm proves to be too computationally demanding, the classic tabu search variant can be considered alternatively, as it still provides with good results in significantly less time. Before implementing the solution, we also recommend observing how the results differ if solutions are compared and selected based on their mode instead of average. A potential outcome could be obtaining better routes at the price of more sensitivity to randomness. It is again up to SSLC to evaluate which approach is preferred.

6.3 Limitations and further research

We conclude by addressing the limitations of our research and giving suggestions for future work. First, our model design is incomplete concerning the minimization of the risk objective. As we explain in

Chapter 5, the likelihood of incurring a delay is not only influenced by the individual carriers or visited airports, but also by the tightness of scheduled transits. Therefore, a first suggestion to improve our work would be to include it in the model design.

We also make several assumptions to simplify the mathematical formulation of the studied problem. For example, we assume an uncapacitated network, whereas both vehicles as flights are subject to capacity restrictions. We also do not distinguish the shipments' goods nature, thereby assuming all goods can be transported without restrictions. In reality, the nature of goods can limit the number of available options: for example, products containing specific lithium batteries can only be transported with freighter aircraft. Moreover, some shipments might need additional assets in place, like cool boxes for transportation, which also need to be arranged beforehand and have influence on the route in general. In our model, we account only for the simplest customs clearance scenario, namely performing the export clearance at the origin airport and the import clearance at destination. This is rather simplistic, as customs can be performed in many other ways. For example, for shipments with a T1 status, the import clearance can be postponed until delivery. Another possibility would be to do the export clearance at a transit airport. This is not uncommon, as certain airports have convenient clearance procedures for certain goods. Furthermore, we specify that customs clearance needs to be done after export preparation at the origin and prior to import preparation at destination. Although this is the most used order of operations, it is not mandatory. For example, a shipment arriving past closing time at the airport handler at the origin could still be cleared for customs, then wait until the handler opens again and be serviced for departure. Our approach dictates it must wait until reopening of the handler office, prepared for export and only then customs cleared, taking more time than necessary in practice. Altogether, incorporating these considerations in the current model would make it more realistic.

Finally, there are aspects that are yet not considered by SSLC in their operations but would be interesting to study. As discussed in Chapter 2, environmental metrics are not considered actively for the route calculation. Nevertheless, we must acknowledge the increased relevance of sustainability in the field of logistics. Therefore, it would be interesting to observe how routing decisions are influenced when taking green KPIs into account. For the same reason, it would also be interesting to explore the possibilities of consolidation within this context. The benefit of consolidating shipments on one route could be potentially twofold: first, it would reduce the transportation costs per shipment; secondly, it would also reduce the carbon footprint per shipment. Lastly, we would recommend exploring the possibilities of route re-optimization, following disruptions like major delays. In the current design, if a shipment misses a flight, it waits until the best next option to the subsequent airport. This is however not optimal, as agents would normally look for alternative routes in the system. Therefore, we argue that designing a solution capable of optimally recalculating a route following such disruptive event would yield better results, thus making this an interesting topic for future work.

References

Amine, K. 2019. "Multiobjective Simulated Annealing: Principles and Algorithm Variants." *Advances in Operations Research.*

Angelelli, E., C. Archetti, and L. Peirano. 2020. "A Methauristic for the Air Transportation Freight Forwarder Service Problem."

Archetti, A., and L. Peirano. 2019. "Air Intermodal Freight Transportation: The Freight Forwarder Service Problem." *Omega.*

Archetti, C, L. Peirano, and M.G Speranza. 2021. "Optimization in Multimodal Freight Transportation Problems: A Survey." *European Journal of Operational Research.*

Archetti, C., A. Hertz, and M.G. Speranza. 2005. "Metaheuristics for the Team Orienteering Problem."

Arostegui, M.A., S.N. Atadeniz, and B.M. Khumawala. 2006. "An Empirical Comparison of Tabu Search, SImulated Annealing, and Genetic Algorithm for Facilities Location Problems." *International Journal of Production Economics.*

Ayar, B., and H. Yaman. 2012. "An Intermodal Multicommodity Routing Problem with Scheduled Services."

Ayed, H., C. Galvez-Fernandez, Z. Habbas, and D. Khadraoui. 2010. "Solving Time-Dependent Multimodal Transport Problems Using a Transfer Graph Model." *Computers & Industrial Engineering.*

Belhaiza, S., P. Hansen, and G. Laporte. 2013. "A Hybrid Variable Neighborhood Tabu Search Heuristic for the Vehicle Routing Problem with Multiple Time Windows." *Computers & Operations Research.*

Bhandari, P. 2020. *Central Tendency: Understanding the Mean, Median & Mode.* July 30. https://www.scribbr.com/statistics/central-tendency.

Brandão, J., and R. Eglese. 2008. "A Deterministic Tabu Search Algorithm for the Capacitated Arc Routing Problem." *Computers & Operations Research.*

Brock, T. 2022. *Pareto Principle.* April 7. https://www.investopedia.com/terms/p/paretoprinciple.asp.

Censor, Y. 1977. "Pareto Optimality in Multiobjective Problems." *Applied Mathematics and Optimization.*

Chabini, I. 1997. "Discrete Dynamic Shortest Path Problems in Transportation Applications: Complexity and Algorithms with Optimal Run Time." *Transportation Research.*

Chang, T.S. 2008. "Best Routes Selection in International Intermodal Networks." *Computers & Operations Research.*

Chen, C.C., and Y.J. Lai. 2015. "Modeling and Optimizing Intermodal Logistics Route Across the Taiwan Straits." *Journal of the Eastern Asia Society for Transportation Studies.*

Crainic, T.G., M. Gendreau, P. Soriano, and M. Toulouse. 1993. "A Tabu Search Procedure for Multicommodity Location/Allocation with Balancing Requirements." *Operations Research.*

Deb, K. 2011. "Multiobjective Optimization Using Evolutionary Algorithms."

Delignette-Muller, M.L. 2014. "fitdistrplus: An R Package for Fitting Distributions."

Domuta, C., N.S. Tipi, D. Radu, M. Hulea, and A. Astilean. 2012. "Finding Intermodal and Multi-Objective Time-Dependent Best Routes With Exact Algorithm."

Dong, X., W. Li, and L. Zheng. 2013. "Ant Colony Optimization For a Resource-Constrained Shortest Path Problem with Applications in Multimodal Transport."

Dua, Aman, and Deepankar Sinha. 2019. "Assessment of Quality of Multimodal Transportation for Containerized Exports." *IIM Kozhikode Society & Management Review.*

Eldrandaly, K.A., A.H.N. Ahmed, and A.M.F. AbdAllah. 2008. "Routing Problems: A Survey."

Forman, Ernest, and Kirti Peniwati. 1997. "Aggregating individual judgments and priorities with the Analytic Hierarchy Process." *European Journal of Operational Research .*

Fu, L. 2000. "An Adaptive Routing Algorithm for In-Vehicle Route Guidance Systems with Real-Time Information."

Gendreau, M., G. Ghiani, and E. Guerriero. 2015. "Time-Dependent Routing Problems: A Review." *Computers & Operations Research.*

Ghiani, Giampaolo, Gilbert Laporte, and Roberto Musmanno. 2013. *Introduction to Logistcis Systems Management.*

Glover, F. 1990. "Tabu search - part II." *Informs Journal on Computing.*

Glover, F., and E. Taillard. 1993. "A User's Guide To Tabu Search." *Annalis of Operational Research.*

Gozacan, Nazlican, and Cisem Lafci. 2020. "Evaluation of Key Performance Indicators of Logistics Firms." *Logistics & Sustainable Transport.*

Hansen, P., Mladenovic N., and J. Brimberg. 2010. "Variable Neighborhood Search." In *Handbook of Metaheuristics*, by M. Gendreau and J. Potvin.

Heerkens, H., and A. van Winden. 2017. *Solving Managerial Problems Systematically.*

Hong, W., K. Tang, A. Zhou, H. Ishibuchi, and X. Yao. 2018. "A Scalable Indicator-Based Evolutionary Algorithm for Large-Scale Multi-Objective Optimization."

Hrusovsky, M., E. Demir, W. Jammernegg, and T. van Woensel. 2016. "Hybrid Simulation and Optimization Approach for Green Intermodal Transportation Problem With Travel Time Uncertainty."

IATA. 2021. "2021 Lithium Battery Guidance."

Isermann, H. 1982. "Linear Lexicographic Optimization."

Jiang, Z., Y. Cheen, X. Li, and B. Li. 2021. "A Heuristic Optimization Approach for Multi-Vehicle and One-Cargo Green Transportation Scheduling in Shipbuilding." *Advanced Engineering Informatics.*

Juan, A.A., J. Faulin, S.E. Grasman, M. Rabe, and G. Figureira. 2015. "A Review of Simheuristics: Extending Metaheuristics to Deal With Stochastic Combinatorial Optimization Problems." *Operations Research Perspectives.*

Kaewfak, K., V. Ammarapala, and V.N. Huynh. 2021. "Multi-objective Optimization of Freight Route Choices in Multimodal Transportation." *International Journal of Computational Intelligence Systems.*

Kengpol, A. S., and M.T. Tuammee. 2014. "The Development of a Framework for Route Selection in Multimodal Transportation." *The International Journal of Logistics Management.*

Koohathongsumrit, N., and W. Meethom. 2020. "Route Selection in Multimodal Transportation Networks: a Hybrid Multiple Criteria Decision-Making Approach."

Kumar, R.A., P. Mohapatra, W.K. Yew, L. Benyoucef, and M.K. Tiwari. 2014. "Route Selection and Consolidation in International Intermodal Freight Transportation."

Lalla-Ruiz, E.A. 2020. "Selecting suppliers with AHP."

Lei, K., W. Zhu, J Hou, and W. Huang. 2014. "Decision of Multimodal Transporation Scheme Based on Swarm Intelligence." *Mathematical Problems in Engineering.*

Lenze, Wiebke, and Nils Niessen. 2021. "Quality in rail timetabling - A detailed review of stakeholders' interests comapred with current practices in Germany." *Journal of Rail Transport Planning & Management.*

Li, L., R.R. Negenborn, and B. De Schutter. 2013. "A Sequential Linear Programming Approach for Flow Assignment in Intermodal Freight Transport."

Li, Y., S. Tong, M. Li, and J. Zhao. 2011. "Solving Mode Selection Problem in Multimodal Transportation Considering Transportation Cost Structure."

Liu, X., and D. Zhang. 2019. "An Improved SPEA2 Algorithm with Local Search for Multi-Objective Investment Decision Making."

Macharis, C., and Y.M. Bontekoning. 2004. "Opportunities for or in Intermodal Freight Transport Research." *European Journal of Operations Research.*

Marett, R., and M. Wright. 1996. "A Comparison of Neighborhood Search Techniques for Multi-Objective Combinatorial Problems."

Michiels, W., E. Aarts, and J. Korst. 2007. "Theoretical Aspects of Local Search."

Miller-Hooks, E.D., and H.S. Mahmassani. 1998a. "Least Expected Time Paths in Stochastic, Time-Varying Transportation Networks."

Miller-Hooks, E.D., and H.S. Mahmassani. 1998b. "Least Possible Time Paths in Stochastic Time-Varying Networks."

Mnif, M., and S. Bouamama. 2017a. "A Multi-Objective Formulation for Multimodal Transportation Network´s Planning Problems."

Mnif, M., and S. Bouamama. 2017b. "A Multi-objective Mathematical Model for Problems Optimization in Multi-modal Transportation Network."

Mutlu, Aysun, Yasanur Kayikci, and Bulent Catay. 2017. "Planning Multimodal Freight Transport Operations: A Literature Review."

Nielsen, L.R., K.A. Andersen, and D. Pertolani. 2013. "Ranking Paths in Stochastic Time-Dependent Networks."

Nydick, Robert L., and Ronald Paul Hill. 1992. "Using the Analytic Hierarchy Process to Structure the Supllier Selection Procedure ." *International Journal of Purchasing and Materials Management.*

Peng, Y., P. Yong, and Y. Luo. 2020. "The Route Problem of Multimodal Transportation With Timetable Under Uncertainty: Multi-Objective Robust Optimization Model and Heuristic Approach."

Pirim, H., E. Bayraktar, and B. Eksioglu. 2008. "Tabu Search: A Comparative Study."

Pisinger, D., and S. Ropke. 2010. "Large Neighborhood Search." In *Handbook of Metaheuristics*, by M. Gendreau and J. Potvin.

Potti, S., and C. Chinnasamy. 2011. "Strength Pareto Evolutionary Algorithm based Multi-Objective Optimization for Shortest Path Routing Problem in Computer Networks." *Journal of Computer Science.*

Ragheb, M. 2020. "Risk quantification." *Nature.*

Solien, Emma, Gemma Nicholson, and Anders Peterson. 2017. "A microscopic evaluation of railway timetable robustness and critical points." *Journal of Rail Transport Planning & Management.*

Song, H., and G. Chen. 2007. "Minimum Cost Delivery Problem in Intermodal Transportation Networks."

SteadieSeifi, M., N. Dellaert, W. Nuijten, T. Van Woensel, and R. Raoufi. 2014. "Multimodal Freight Transportation Planning: A Literature Review."

Stephens, M.A. 1974. "EDF Statistics for Goodness of Fit and Some Comparisons." *Journal of the American Statistical Association.*

Sun, Y., and M. Lang. 2015. "Modeling the Multicommodity Multimodal Routing Problem with Schedule-Based Services and Carbon Dioxide Emission Costs." *Mathematical Problems in Engineering.*

Sun, Y., and X. Li. 2019. "Fuzzy Programming Approaches for Modeling a Customer-Centred Freight Routing Problem in the Road-Rail Intermodal Hub-and-Spoke Network with Fuzzy Soft Time Windows and Multiple Sources of Time Uncertainty."

Sun, Y., M. Lang, and D. Wang. 2015. "Optimization Models and Solution Algorithms for Freight Routing Planning Problem in the Multi-Modal Transportation Networks: A Review of the State-of-the-Art." *The Open Civil Engineering Journal.*

Talbi, E.G. 2009. *Metaheuristics: From Design to Implementation.*

Teng, S., H.L. Ong, and H.C. Huang. 2003. "A Comparative Study of Metaheuristics for Vehicle Routing Problem With Stochastic Demands." *Asia Pacific Journal of Operations Research.*

Teunter, R. H., M. Zied Babai, and A. A. Syntetos. 2009. "ABC Classification: Service Levels and Inventory Costs." *Production and Operations Management Society.*

Treacy, M., and F. Wiersma. 1997. *The Discipline of Market Leaders.*

Votov, A.V., and K. Miettien. 2008. "Visualizing the Pareto Frontier."

Walck, C. 2007. "Hand-book on Statistical Distributions for Experimentalists."

Wellman, P., M. Ford, and K. Larson. 1995. "Path Planning Under Time-Dependent Uncertainty."

Xia, Y., and Z. Fu. 2018. "Improved Tabu Search for the Open Vehicle Routing Problem With Soft Time Windows and Satisfaction Rate." *Cluster Computing.*

Xiong, G, and Y. Wang. 2014. "Best Routes Selection in Multimodal Networks Using Multi-Objective Genetic Algorithm."

Zhang, L., and Z. Peng. 2009. "Optimization Model for Multi-modal transport of Full Loads with Time Windows."

Zitzler, E., and S. Künzuli. 2004. "Indicator-Based Selection in Multiobjective Search."

Zografos, K.G., and K.N. Androutsopoulos. 2008. "Algorithms for Itinerary Planning in Multimodal Transportation Networks." *Transactions On Intelligent Transportation Systems.*

Appendices

A. Product portfolio

SSLC offers a wide range of products. This appendix paragraph describes them in short:

- Express Air: this is one of the most popular products offered. Shipments have higher loading priority than express cargo and can be almost always loaded on board in either compartment five or the aircraft's belly. This is the product that can be (and usually is) booked with the shortest preadvice, with an average LAT of 45 minutes before the flight. It is also usually the fastest type of shipment. Figure 27 shows which hubs globally can be served with this product.
- Express train: transportation jobs are executed by train. Train options are restricted to the European market, more specifically to Germany, Paris, Vienna, Basel and Amsterdam.
- Global Express Air Freight: this is also a form of prioritized air cargo product. Key differences with Express Air are the package dimensions and weight; global express has no restrictions, and destination type (global express covers almost all possible destinations globally, whereas express air covers only specific hubs).
- Spare Parts & Service Logistics: as the name suggests, this product focuses on the flow of spare parts within a company-operated (European-based) network. Flight schedules are fixed, therefore only volumes of transported parts vary within this product.
- SSLC Warehouse: this service grants direct access to apron areas at the airport to ensure the lowest handling times possible. Within the warehouse shipments are prepared for transport, customs cleared, and brought directly to the aircraft (either from the warehouse or tail-to-tail).
- On-board Courier: with this type of shipment, transport is executed and monitored by a traveling courier. Customers often choose this type of solution for temperature-controlled goods, e.g. medical products.
- Tailor-made solutions: this includes booking designated flights (charters) or dedicated truck transports (Direct Deliveries) to guarantee nonstop, high-speed solutions. This is often the most expensive option.

On top of the above options, each transportation job can vary in the combination of origin and destination points. Shipments can be door2door, airport2airport, door2airport, door2station, etc.

Figure 27: Global overview of express flight hubs

B. Problem cluster

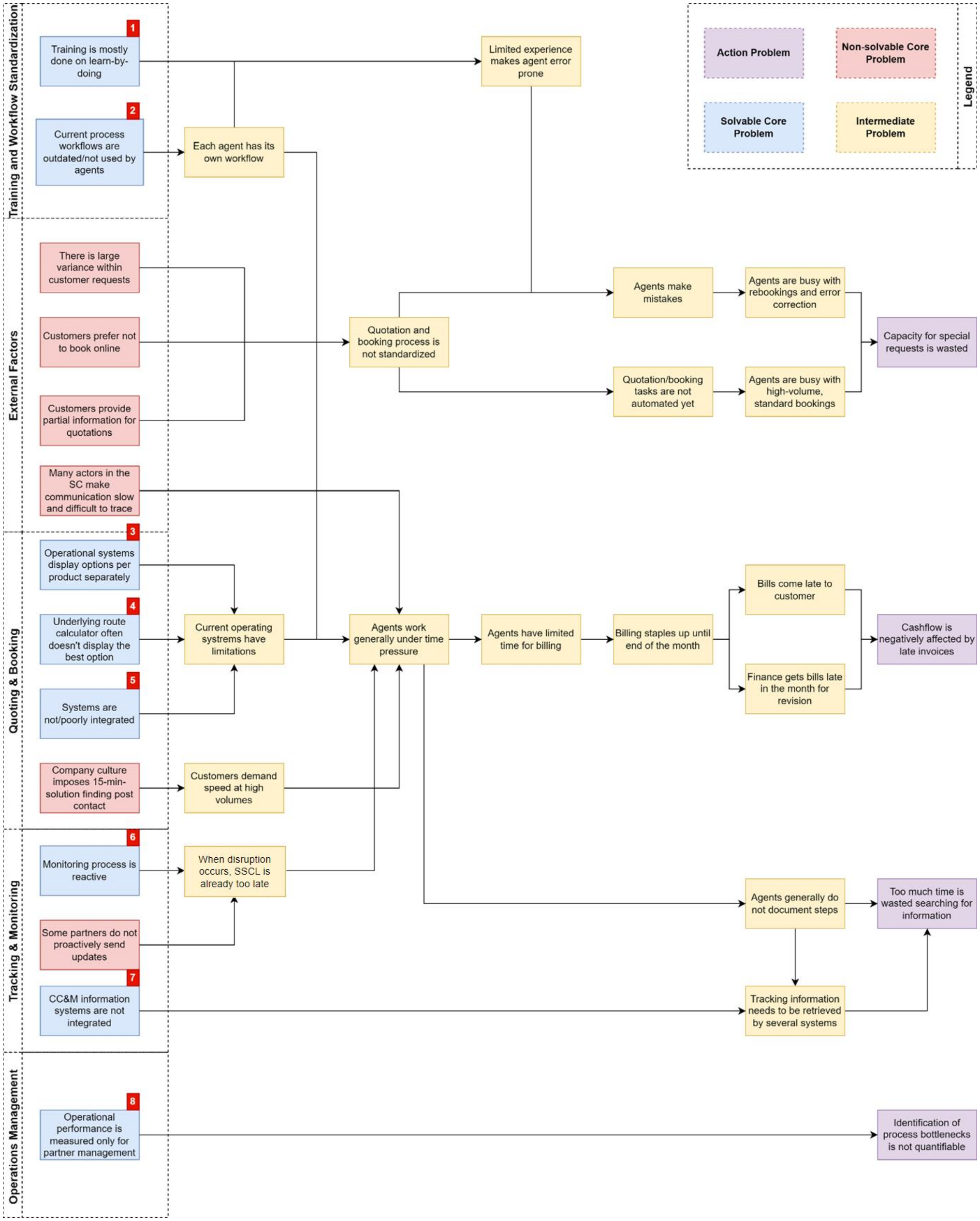


Figure 28: Problem cluster. Action problems are displayed on the right and highlighted in purple. Core problems are depicted on the left, we distinguish between solvable (blue) and non-solvable (red).

Table 30: Solvable core problems (Blue= already being solved, Green = yet to be solved)

Number	Problem	Explanation
1	Training is mostly done on learn-by-doing.	Agents are currently trained with basic online materials; the real learning takes place when they are put on operations. Agents learn therefore slowly and overall make more mistakes within the first year.
2	Current process workflows are outdated/not used by agents.	Agents approach the same processes each with their own workflow, thus causing human errors.
3	Operational systems display options per product category separately.	Shipment requests can be potentially served by multiple product options: the lack of a cross-product comparison slows down the quotation process.
4	Underlying route calculator often does not display the best option	Route calculation does not explicitly account for relevant objectives like minimalization of transit time, delays, costs, etc. Customers therefore do not always get the fastest option and rebooking/delays add work for the CS team.
5	Systems are not/poorly integrated	The operational systems are not well connected, leading to manual work and human errors.
6	Monitoring process is reactive	The CC&M team intervenes only when an irregularity takes place. Hence, they are by default too late when a disruption takes place.
7	CC&M information systems are not integrated	Besides causing unnecessary manual work, the system disaggregation makes it challenging for the CC&M team to retrieve information when disruptions take place.
8	Operational performance is measured only for partner management	Only irregularities during tracking are used for post-hoc analysis; this leads to an incomplete overview of operations performance

This Appendix discusses the identification of problems within the company which lead to selecting the core problem introduced in Section 1.2.2 . In their method, Heerkens and van Winden (2017) distinguish two types of managerial problems: action problems and core problems. The former represent an evident discrepancy between a desired and actual situation, whereas the latter are their root causes. Action problems are usually noticed first; root-cause analysis leads consequently to core problems. In order to solve an action problem, researchers should focus on solving core problems. Figure 28 shows, by means of a problem cluster, the existing relations between observed action problems and core problems within the Customer Service. During our preliminary research, we identified fourteen core problems in total, of which we deemed eight to be solvable. Table 30 shows these eight problems in more detail together with their description. Of the eight problems, four are currently being solved. SSLC Netherlands is focusing on setting up a novel training program (addressing 1 and 2) and the GOP is in an advanced stage of solving 3 and 5. This leaves us with 4, 6, 7 and 8 as candidate core problems to tackle. Ultimately, we choose to solve problem 4.

C. KPI selection list

Table 31 Displays the list of KPIs considered to measure the routing quality. The KPIs are either derived from the answers of the panel of experts (PoE) or literature.

Table 31: KPI selection on route quality

KPI	Objective	Description	Source	Selected (Y/N)	Comment
Transit time	Speed	Difference between arrival and departure time.	PoE, (Lenze and Niessen 2021), (Solien, Nicholson and Peterson 2017)	Yes	-
Transit time quotient	Speed	Ratio between actual transit time and longest possible transit time on a route.	PoE, (Lenze and Niessen 2021)	Yes	-
Occupancy rate	Speed	Compressed time (i.e., frequency of departing flights/hour) divided by the transportation time window.	(Lenze and Niessen 2021), (Solien, Nicholson and Peterson 2017)	No	Impossible to retrieve information from historical data; overall complex to measure.
Delivery delay	Risk	Time difference between actual and scheduled delivery at destination.	PoE, (Dua and Sinha 2019), (Lenze and Niessen 2021), (Solien, Nicholson and Peterson 2017), (Ghiani, Laporte and Musmanno 2013), (Gozacan and Lafci 2020), (SteadieSeifi, et al. 2014)	Yes	-
On-time deliveries	Risk	Percentage of on-time deliveries.	PoE, (Dua and Sinha 2019), (Ghiani, Laporte and Musmanno 2013), (Gozacan and Lafci 2020)	No	Less informative than delivery delay; a delay of 5 minutes is not as bad as a delay of 2 hours.
(Average) number of missed transits	Risk	Average number of missed transits.	PoE	No	Not retrievable from data.

Reroute probability	Risk	Probability of missing flight connection.	PoE	No	See number of missed transits.
Partner reliability	Risk	Combined metric of delays caused by partner, partner timeliness of communication, accuracy of information provided.	PoE	No	The indicators composing this metric are mutually dependent, making it tricky to interpret it objectively.
Total transit delay	Risk	Sum of delays per leg throughout the entire route.	PoE, (Lenze and Niessen 2021), (Solien, Nicholson and Peterson 2017)	Yes	This type of delay doesn't affect the customer but has influence on workload for SSLC agents and costs, and is therefore relevant.
Resilience	Risk	Ratio of differences between recovered and disrupted state, and disrupted and initial state.	(SteadieSeifi, et al. 2014)	No	Not retrievable in the data.
Total operating costs	Costs	Total operating costs.	PoE, (Ghiani, Laporte and Musmanno 2013), (Gozacan and Lafci 2020), (Mutlu, Kayikci and Catay 2017)	Yes	-
CO_2 emissions	Other	CO_2 emissions.	PoE, (Dua and Sinha 2019), (Gozacan and Lafci 2020), (Mutlu, Kayikci and Catay 2017)	No	-

D. AIJ method calculations

This appendix section explains how we performed the AIJ method in order to obtain weights for the KPIs used to assess route performance. Figure 29 shows the judgments filled by the respondents. The first step in the AIJ is to aggregate the judgements by taking the geometric mean of each distinct criteria combination. The geometric mean can be described by the formula:

$$\left(\prod_{i=1}^{n} x_i\right)^{\frac{1}{n}} = \sqrt[n]{x_1 x_2 \ldots x_n} \ \ with\ n = number\ of\ values; x_i = values\ to\ average$$

Figure 30 shows the geometric means within the aggregated (first) matrix. The next step consists of normalizing the judgment values by the column sums. These values are visible in the normalized matrix. By averaging the rows, we obtain the priorities (weights). The last step consists of checking the latter values for consistency. First, we build a new matrix, inserting the weights in row 0 and the aggregated judgements below (see the first consistency check matrix). Next, we multiply each judgment value in a column by its corresponding column weight. We sum the obtained values by each row and calculate alpha by dividing each sum by the corresponding weight values. These calculations are applied to the second consistency check matrix in Figure 30. With the alpha values, we calculate:

$$\lambda_{max} = \sum_{i=1}^{n} \alpha_n$$

With λ_{max} we then calculate the consistency index (C.I.), defined as:

$$C.I. = \frac{\lambda_{max} - n}{n - 1}\ with\ n = number\ of\ criteria$$

Finally, we calculate the consistency ratio (C.R.). For the obtained weights to be consistent, the C.R. should be less than 0.10. The C.R. is calculated with:

$$C.R. = \frac{C.I.}{R.I.}\ with\ R.I. = random\ index$$

For a value of n = 5 (as we compare five KPIs), the R.I. has value 1.12 (Lalla-Ruiz 2020). We thus obtain a value for the C.R. of 0.04 confirming our weights are consistent.

Matrix 1	Transit time	Transit time quotient	Delivery delay	Total transit delay	Total operating costs
Transit time	1	7	0.25	0.2	8
Transit time quotient	0.14285714	1	0.142857143	0.142857143	0.5
Delivery delay	4	7	1	0.2	7
Total transit delay	5	7	5	1	8
Total operating costs	0.125	2	0.142857143	0.125	1

Matrix 2	Transit time	Transit time quotient	Delivery delay	Total transit delay	Total operating costs
Transit time	1	5	1	1	5
Transit time quotient	0.2	1	0.2	0.333333333	0.2
Delivery delay	1	5	1	7	3
Total transit delay	1	3	0.142857143	1	1
Total operating costs	0.2	5	0.333333333	1	1

Matrix 3	Transit time	Transit time quotient	Delivery delay	Total transit delay	Total operating costs
Transit time	1	0.166666667	8	6	3
Transit time quotient	6	1	1	1	4
Delivery delay	0.125	1	1	1	5
Total transit delay	0.16666667	1	1	1	5
Total operating costs	0.33333333	0.25	0.2	0.2	1

Matrix 4	Transit time	Transit time quotient	Delivery delay	Total transit delay	Total operating costs
Transit time	1	0.142857143	0.2	0.125	5
Transit time quotient	7	1	0.2	0.166666667	6
Delivery delay	5	5	1	5	6
Total transit delay	8	6	0.2	1	4
Total operating costs	0.2	0.166666667	0.166666667	0.25	1

Matrix 5	Transit time	Transit time quotient	Delivery delay	Total transit delay	Total operating costs
Transit time	1	0.166666667	7	7	1
Transit time quotient	6	1	7	7	4
Delivery delay	0.14285714	0.142857143	1	0.166666667	1
Total transit delay	0.14285714	0.142857143	6	1	1
Total operating costs	1	0.25	1	1	1

Figure 29: Individual expert judgments

Aggregated matrix	Transit time	Transit time quotient	Delivery delay	Total transit delay	Total operating costs
Transit time	1.000	0.674	1.229	1.010	3.594
Transit time quotient	1.484	1.000	0.525	0.561	1.572
Delivery delay	0.814	1.904	1.000	1.031	3.630
Total transit delay	0.990	1.783	0.970	1.000	2.759
Total operating costs	0.278	0.636	0.276	0.362	1.000
Column total	4.567	5.996	3.999	3.964	12.556

Normalized matrix	Transit time	Transit time quotient	Delivery delay	Total transit delay	Total operating costs	Priorities
Transit time	0.219	0.112	0.307	0.255	0.286	0.236
Transit time quotient	0.325	0.167	0.131	0.142	0.125	0.178
Delivery delay	0.178	0.317	0.250	0.260	0.289	0.259
Total transit delay	0.217	0.297	0.242	0.252	0.220	0.246
Total operating costs	0.061	0.106	0.069	0.091	0.080	0.081
	1.000	1.000	1.000	1.000	1.000	

Consistency check	Transit time	Transit time quotient	Delivery delay	Total transit delay	Total operating costs
Weights	0.236	0.178	0.259	0.246	0.081
Transit time	1.000	0.674	1.229	1.010	3.594
Transit time quotient	1.484	1.000	0.525	0.561	1.572
Delivery delay	0.814	1.904	1.000	1.031	3.630
Total transit delay	0.990	1.783	0.970	1.000	2.759
Total operating costs	0.278	0.636	0.276	0.362	1.000

Consistency check	Transit time	Transit time quotient	Delivery delay	Total transit delay	Total operating costs	Sum	Alpha
Transit time	0.236	0.120	0.318	0.248	0.293	1.215	5.149
Transit time quotient	0.350	0.178	0.136	0.138	0.128	0.930	5.225
Delivery delay	0.192	0.339	0.259	0.253	0.295	1.339	5.169
Total transit delay	0.234	0.317	0.251	0.246	0.225	1.272	5.178
Total operating costs	0.066	0.113	0.071	0.089	0.081	0.421	5.168

Lambda max	5.177774035
C.I	0.044443509
R.I	1.12
C.R.	0.039681704

Figure 30: Aggregated matrix and consistency check

E. Auxiliary procedures pseudocode

PROCEDURE 1: OFFICEHOURSORIGIN

```
Input: arrivalTime, o, cu
Output: readyTime
if ((arrivalTime mod 24 ≥ opsopen_o) and ((arrivalTime + exim_o) mod 24 ≤
opsclosed_o)) then
    | readyTime = arrivalTime + exim_o
else
    | readyTime = arrivalTime + 24 + exim_o
end if
if (cu == 1) then
    | if ((readyTime mod 24 ≥ customsopen_o) and ((readyTime + cs_o) mod 24 ≤
    | customsclosed_o)) then
    |     | readyTime = readyTime + cs_o
    | else
    |     | readyTime = readyTime + 24 + cs_o
    | end if
end if
return readyTime
```

Figure 31: Pseudocode for calculating the ready time at the origin airport o.

PROCEDURE 2: OFFICEHOURSTRANSIT

```
Input: arrivalTime, a
Output: readyTime
if ((arrivalTime mod 24 ≥ opsopen_a) and ((arrivalTime + s_a) mod 24 ≤ opsclosed_a))
then
    | readyTime = arrivalTime + s_a
else
    | readyTime = arrivalTime + 24 + s_a
end if
return readyTime
```

Figure 32: Pseudocode for calculating the ready time at a transiting airport a.

PROCEDURE 3: OFFICEHOURSENDING

```
Input: arrivalTime, e, cu
Output: readyTime
if (cu == 1) then
    | if ((arrivalTime mod 24 ≥ customsopen_e) and ((arrivalTime + cs_e)
    | mod 24 ≤ customsclosed_e)) then
    |     | readyTime = arrivalTime + cs_e
    | else
    |     | readyTime = arrivalTime + 24 + cs_e
    | end if
end if
if ((readyTime mod 24 ≥ opsopen_e) and ((readyTime + exim_e) mod 24 ≤
opsclosed_e)) then
    | readyTime = readyTime + exim_e
else
    | readyTime = readyTime + 24 + exim_e
end if
return readyTime
```

Figure 33: Pseudocode for calculating the ready time at the ending airport e.

PROCEDURE 4: REOPTIMIZEROUTE

1 ***Input:*** *airports*
2 ***Output:*** *solution*
3 ***Initialize:*** *carriers[], objValues[], time, risk, cost, carrierTime, carrierRisk, carrierCost, balScore, balBest, bestTime, bestRisk, bestCost, carBest, ready*
4 ***Set*** *balbest = M, o = first element in airports, e = last element in airports*
5 *Calculate arrival time at* $o \rightarrow tarr_o^k = tarr_p + t_{po}^k$, *calculate ready time at* $o{:}\ ready = OfficeHoursOrigin(tarr_o^k, o, cu)$
6 $cost = c_{po}^k + c_{ed}^k + cu * (cs_o + cs_e) + \sum_{i\ in\ airports} c_i$, $risk = r_{po}^k + r_{ed}^k + cu * (cr_o + cr_e) + \sum_{i\ in\ airports} r_i$
7 ***for*** *(all airports i except the last element in airports)* ***do***
8   ***set*** *departure time for vehicle carrier from i to i+1 to ready time* $\rightarrow dep_{i_{i+1}}^{v} = ready$
9   ***for*** *(all carriers k between i and i+1)* ***do***
10   ***if*** *(i is the origin airport o)* ***then***
11   *Calculate arrival time at i+1*$\rightarrow tarr_{i+1}^k = dep_{i(i+1)}^k + t_{i(i+1)}^k$ *given* $dep_{i(i+1)}^k \leq tarr_p + 24$ *and* $dep_{i(i+1)}^k \geq ready$
12   ***if*** *(i+1 = e)* ***then***
13   *Calculate arrival time at ending airport* → *carrierTime = OfficeHoursEnding(*$tarr^k{}_{i+1}$*, i+1, cu)*
14   ***else***
15   *Calculate arrival time at transit airport* → *carrierTime = OfficeSHoursTransit(*$tarr^k{}_{i+1}$*, i+1)*
16   ***end if***
17   *Store carrier specific risk and cost carrierRisk =* $r_{i\,i+1}{}^k$ *, carrierCost =* $c_{i\,i+1}{}^k$
18   *Balance parameters balScore =* α ** carrierTime +* β ** carrierRisk +* γ ** carrierCost*
19   ***if*** *(balScore < balBest)* ***then*** *select carrier k as best option between i and i+1*
20   *balBest = balScore, carBest = k, bestTime = carrierTime, bestRisk = carrierRisk, bestCost = carrierCost*
21   ***end if***
22   ***else*** *(i is a transit airport)*
23   *Follow calculate time, risk and cost of using k for transportation between i and i+1, store k values if k has the best balanced score (steps 11-20); given* $dep_{i(i+1)}^k \geq ready$
24   ***end if***
25   ***end for***
26   *Reset balBest = M; time = time + bestTime, risk = risk + bestRisk, cost = cost + bestCost, add carBest to carriers[]*
27 ***end for***
28 *objValues = [time, risk, cost]*
29 *solution = [carriers, objValues]*
30 ***return*** *solution*

Figure 34: Pseudocode for finding the optimal carrier combination given a sequence of input airports

CHOOSEBESTNEIGHBOR

Input: *Neighborhood, TabuList, currentBest*
Output: *BestNeighbor*
Set *tabuMatch = false, bestFound = false, bestNeighbor = 0*
While *(bestFound == false)* ***do***
    for *(all moves j in TabuList)* ***do***
        if *(Neihborhood[bestNeighbor][move] == j)* ***then*** *(match found in Tabu list)*
            if *(Neighborhood[BestNeighbor] ≥ currentBest)* ***then*** *(aspiration criterion)*
                tabuMatch = true; exit for
            end if
        end if
    end for
    if *(tabuMatch == false)* ***then***
        bestFound = true
    else
        bestNeighbor +=1
        tabuMatch = false (reset)
    end if
end while
return *Neighborhood[bestNeighbor]*

Figure 35: Pseudocode for selecting the best neighbor solution from a generated neighborhood. The neighbor solution is chosen if no match in the tabu list is found or if it satisfies the aspiration criterion. Otherwise, the next best neighbor is chosen.

F. Example plots of visual goodness-of-fit tests

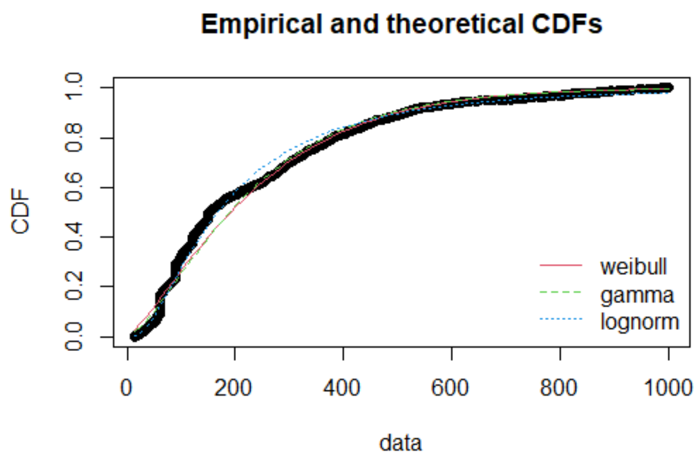
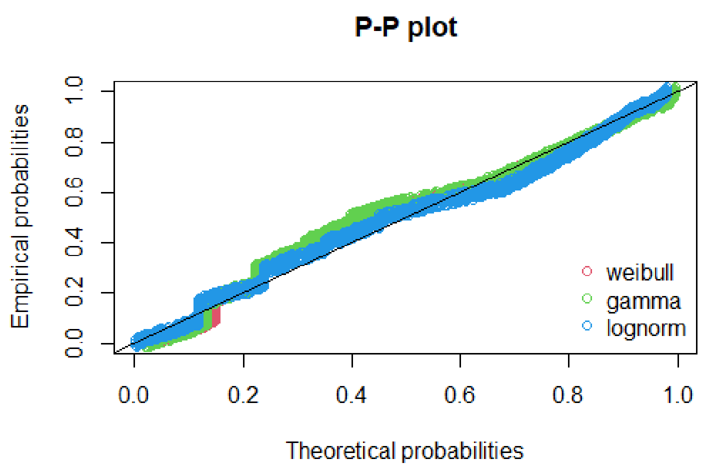


Figure 36: Example plot of the service times distributions at airport a. The plot compares the observed data with the cumulative distributions of the Weibull, Gamma, exponential and lognormal distributions.



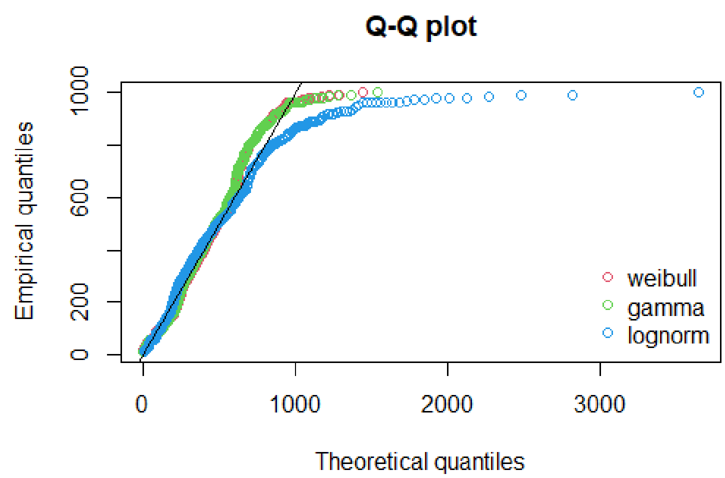


Figure 37: P-P (left) and Q-Q plot for visually inspecting the goodness-of-fit of the Weibull, Gamma exponential and lognormal distributions on service times data.

G. Experimental outcomes of the tuning experiments

This appendix section presents the measured model performance with the selected settings as presented in Section 5.2.1 . All experiments are performed on the same problem instance.

In Table 32 we present the results of tuning the simheuristic parameters. We tune them using the balanced model configuration and TS-M1 and extend the found settings to the remaining models.

Table 32:Tuning outcomes for the simheuristic parameters $iter_i$ and $iter_f$. The highlighted rows indicate the best results obtained for either parameter.

Simulation param.		*Tabu search param.*		*Results coefficients of variation*			*Computational performance*
$\boldsymbol{iter_i}$	$iter_f$	T_{length}	T_{iter}	Time	Risk	Cost	Run time (s)
100	1000	5	100	0.08	0.08	0.04	81
500	1000	5	100	0.01	0.08	0.02	258
1000	1000	5	100	0.003	0.02	0.006	480.5
1500	**1000**	**5**	**100**	**0.003**	**0.02**	**0.005**	**788**
5000	1000	5	100	0.015	0.07	0.001	1082.3
1500	5000	5	100	0.002	0.015	0.002	426.5
1500	10000	5	100	0.002	0.010	0.003	425
1500	**15000**	**5**	**100**	**0.002**	**0.009**	**0.002**	**503.8**
1500	30000	5	100	0.002	0.013	0.003	1056.5

Following the first batch of experiments, we find $iter_i = 1500$ and $iter_f = 15000$ yield good results in acceptable time. We use these values for the remainder of the experiments. Below we present the performance of each solution, using the found optimal parameter configuration. A comprehensive overview of the tuning experiments is omitted for conciseness.

Table 33: Outcomes of the tuning experiments for the Tabu search M1.

TS – M1							
Model	T_{length}	T_{iter}	$depwithin$	Transit time (h)	Total risk	Total cost (euro)	Run time (s)
1	5	100	∞	30.9	7.9	1888	419.6
2	5	100	∞	47.6	8.1	2632.5	525.7
3	5	100	∞	47.6	6.6	1052.4	513.5
4	7	100	∞	47.7	6.4	1044.5	479.1
5	7	100	∞	47.6	6.6	1052.6	565.6

Table 34: Outcomes of the tuning experiments for the Tabu search M2.

TS – M1							
Model	T_{length}	T_{iter}	$depwithin$	Transit time (h)	Total risk	Total cost (euro)	Run time (s)
1	5	100	∞	31.2	8.4	1895	442.7
2	5	100	∞	47.7	8	2629.8	554.6
3	5	100	∞	47.6	6.6	1048.6	541.7
4	7	100	∞	47.6	6.6	1047.4	481.5
5	6	100	∞	47.6	6.6	1046.6	596.7

Table 35: Outcomes of the tuning experiments for the Hybrid Tabu search.

MNTS							
Model	T_{length}	T_{iter}	$depwithin$	Transit time (h)	Total risk	Total cost (euro)	Run time (s)
1	6	1000	∞	31.2	8.4	1896	1792.9
2	6	1000	∞	47.7	8	2629	1941.1
3	6	1000	∞	70.8	5	893.1	1896
4	7	1000	∞	47.6	6.5	1046.9	1927.3
5	7	1000	∞	47.6	6.6	1047.9	2386.8

H. Experimental outcomes of the deterministic analysis

This appendix section presents all outcomes of the experiments used to evaluate the proposed solutions deterministically. Each table below presents the integral results of solving 49 scenarios generated with historical order data, given a particular model setup. Within each table we compare the solutions' performance with the generated initial solution[8], their run time and indicate whether they change the route of the initial solution.

Table 36: Experimental results of deterministically solving the test instances with the time-greedy ($\alpha = 1$) model.

	Initial solution			Tabu search M1					Tabu search M2					MNTS				
Experiment	Transit time	Total risk	Total cost	% TT	%TR	%TC	Run time	Change	% TT	%TR	%TC	Run time	Change	% TT	%TR	%TC	Run time	Change
1	40.2	0.63	1291	-54.73%	-31.75%	50.66%	32.48	Yes	-54.73%	-31.75%	50.66%	32.63	Yes	-59.70%	-70.76%	24.71%	148.72	Yes
2	15	0.25	808	0.00%	0.00%	0.00%	33.89	No	0.00%	0.00%	0.00%	39.31	No	-17.33%	-33.15%	3.22%	154.81	Yes
3	10.6	0.18	631	0.00%	0.00%	0.00%	38.18	No	0.00%	0.00%	0.00%	43.59	No	0.00%	0.00%	0.00%	154.87	No
4	14.1	0.21	677	0.00%	0.00%	0.00%	42.77	No	0.00%	0.00%	0.00%	44.05	No	0.00%	0.00%	0.00%	155.87	No
5	17.5	0.19	270	-36.57%	12.15%	74.44%	42.73	Yes	-36.57%	12.15%	74.44%	45.02	Yes	-36.57%	12.15%	74.44%	155.07	Yes
6	10.1	0.21	677	0.00%	0.00%	0.00%	42.26	No	0.00%	0.00%	0.00%	44.22	No	0.00%	0.00%	0.00%	154.67	No
7	11.3	0.30	547	0.00%	0.00%	0.00%	45.81	No	0.00%	0.00%	0.00%	46.53	No	-4.42%	-29.16%	47.35%	154.39	Yes
8	11.3	0.30	547	0.00%	0.00%	0.00%	44.87	No	0.00%	0.00%	0.00%	47.85	No	-4.42%	-29.16%	47.35%	154.51	Yes
9	11.3	0.30	547	0.00%	0.00%	0.00%	44.51	No	0.00%	0.00%	0.00%	47.20	No	-4.42%	-29.16%	47.35%	154.49	Yes
10	10.1	0.21	677	0.00%	0.00%	0.00%	43.48	No	0.00%	0.00%	0.00%	50.60	No	0.00%	0.00%	0.00%	154.11	No
11	10.1	0.21	677	0.00%	0.00%	0.00%	43.25	No	0.00%	0.00%	0.00%	46.09	No	0.00%	0.00%	0.00%	152.86	No
12	9.1	0.21	677	0.00%	0.00%	0.00%	43.84	No	0.00%	0.00%	0.00%	52.12	No	0.00%	0.00%	0.00%	152.47	No
13	15.2	0.47	534	-19.74%	-54.54%	31.46%	33.52	Yes	-19.74%	-54.54%	31.46%	41.84	Yes	-19.74%	-54.54%	31.46%	143.74	Yes
14	9.1	0.21	677	0.00%	0.00%	0.00%	43.74	No	0.00%	0.00%	0.00%	44.15	No	0.00%	0.00%	0.00%	156.27	No
15	23.5	0.63	948	-26.81%	-46.89%	-24.58%	39.67	Yes	-26.81%	-46.89%	-24.58%	39.99	Yes	-48.94%	-55.13%	2.32%	152.89	Yes
16	16.9	0.48	1237	-23.08%	-55.39%	-43.98%	32.83	Yes	-23.08%	-55.39%	-43.98%	33.74	Yes	-23.08%	-55.39%	-43.98%	147.12	Yes
17	17.4	0.48	1237	-22.41%	-55.10%	-43.98%	32.82	Yes	-22.41%	-55.10%	-43.98%	33.76	Yes	-22.41%	-55.10%	-43.98%	145.78	Yes
18	37.6	0.48	888	-48.94%	-41.40%	95.50%	32.53	Yes	-48.94%	-41.40%	95.50%	33.54	Yes	-50.53%	-19.24%	122.86%	147.66	Yes
19	16.3	0.47	538	-24.54%	-54.04%	31.23%	32.90	Yes	-24.54%	-54.04%	31.23%	34.92	Yes	-24.54%	-54.04%	31.23%	145.81	Yes
20	11.5	0.47	525	0.00%	0.00%	0.00%	33.23	No	0.00%	0.00%	0.00%	35.05	No	0.00%	0.00%	0.00%	145.98	No
21	88.5	0.47	538	-83.84%	-54.22%	31.23%	33.35	Yes	-83.84%	-54.22%	31.23%	34.25	Yes	-83.84%	-54.22%	31.23%	149.10	Yes
22	12.8	0.26	506	-10.94%	36.91%	108.89%	39.87	Yes	-10.94%	36.91%	108.89%	41.26	Yes	-10.94%	36.91%	108.89%	152.42	Yes
23	15.3	0.47	525	-19.61%	-54.99%	32.00%	32.84	Yes	-19.61%	-54.99%	32.00%	33.63	Yes	-19.61%	-54.99%	32.00%	144.86	Yes
24	15.2	0.47	536	-14.47%	-38.43%	94.78%	33.10	Yes	-14.47%	-38.43%	94.78%	34.11	Yes	-14.47%	-38.43%	94.78%	147.50	Yes
25	11.5	0.31	430	0.00%	0.00%	0.00%	41.00	No	0.00%	0.00%	0.00%	42.47	No	0.00%	0.00%	0.00%	151.20	No
26	19.9	0.48	380	-33.17%	-35.72%	138.68%	37.95	Yes	-33.17%	-35.72%	138.68%	39.34	Yes	-33.17%	-58.15%	107.63%	151.24	Yes
27	34.3	0.47	525	-50.73%	-41.94%	161.52%	32.75	Yes	-50.73%	-41.94%	161.52%	34.16	Yes	-52.48%	-19.50%	207.81%	146.73	Yes
28	35.3	0.47	525	-52.12%	-41.72%	161.52%	32.65	Yes	-52.12%	-41.72%	161.52%	45.85	Yes	-53.82%	-19.40%	207.81%	145.40	Yes
29	13.4	0.34	649	0.00%	0.00%	0.00%	41.00	No	0.00%	0.00%	0.00%	42.94	No	-5.22%	-14.16%	130.51%	155.79	Yes
30	18.1	0.50	1390	-21.55%	-53.30%	-39.14%	32.98	Yes	-21.55%	-53.30%	-39.14%	33.60	Yes	-21.55%	-53.30%	-39.14%	146.31	Yes
31	13.1	0.26	1142	-6.11%	12.08%	28.11%	41.05	Yes	-6.11%	12.08%	28.11%	41.22	Yes	-6.11%	12.08%	28.11%	151.64	Yes
32	13.5	0.25	379	-16.30%	-11.40%	126.12%	40.46	Yes	-16.30%	-11.40%	126.12%	42.64	Yes	-16.30%	-11.40%	126.12%	151.06	Yes
33	16.2	0.25	795	0.00%	0.00%	0.00%	37.77	No	0.00%	0.00%	0.00%	40.40	No	-16.67%	-32.76%	2.01%	150.36	Yes
34	33.7	0.47	525	-48.66%	-41.67%	161.52%	33.38	Yes	-48.66%	-41.67%	161.52%	34.29	Yes	-50.45%	-19.37%	207.81%	148.77	Yes
35	15.2	0.49	678	-13.16%	-52.63%	24.78%	33.29	Yes	-13.16%	-52.63%	24.78%	34.49	Yes	-13.16%	-52.63%	24.78%	146.70	Yes
36	27.2	0.63	1210	-39.34%	-38.45%	9.83%	38.22	Yes	-39.34%	-38.45%	9.83%	39.54	Yes	-43.38%	-39.36%	58.84%	149.97	Yes
37	131.4	0.47	534	-87.06%	-41.53%	158.80%	32.54	Yes	-87.06%	-41.53%	158.80%	33.87	Yes	-87.52%	-19.31%	204.31%	144.81	Yes
38	35.4	0.49	678	-49.44%	-58.06%	119.47%	33.12	Yes	-49.44%	-58.06%	119.47%	34.18	Yes	-50.85%	-18.73%	160.91%	147.92	Yes
39	15	0.47	525	-14.67%	-38.77%	96.76%	32.73	Yes	-14.67%	-38.77%	96.76%	34.77	Yes	-14.67%	-38.77%	96.76%	148.71	Yes
40	128.4	0.49	708	-86.76%	-62.76%	115.25%	33.50	Yes	-86.76%	-62.76%	115.25%	35.43	Yes	-86.76%	-62.76%	115.25%	149.70	Yes
41	133.9	0.63	981	-85.59%	-28.50%	37.21%	38.52	Yes	-85.59%	-28.50%	37.21%	39.50	Yes	-88.20%	-70.65%	32.52%	154.12	Yes
42	16.4	0.32	1375	0.00%	0.00%	0.00%	37.37	No	0.00%	0.00%	0.00%	38.80	No	-16.46%	-46.92%	-42.84%	151.33	Yes
43	31.7	0.47	525	-47.32%	-38.38%	96.76%	33.34	Yes	-47.32%	-38.38%	96.76%	34.82	Yes	-47.95%	-31.58%	52.57%	146.54	Yes
44	31.4	0.20	342	-47.77%	49.05%	264.04%	38.57	Yes	-47.77%	49.05%	264.04%	40.01	Yes	-48.41%	40.19%	164.91%	155.55	Yes
45	15.5	0.49	1362	0.00%	0.00%	0.00%	33.62	No	0.00%	0.00%	0.00%	34.17	No	0.00%	0.00%	0.00%	143.18	No
46	13.2	0.25	1155	-9.09%	9.92%	25.11%	40.82	Yes	-9.09%	9.92%	25.11%	42.22	Yes	-9.09%	9.92%	25.11%	153.85	Yes
47	16.1	0.47	525	-24.84%	-54.28%	32.00%	32.79	Yes	-24.84%	-54.28%	32.00%	34.24	Yes	-24.84%	-54.28%	32.00%	147.78	Yes
48	15.2	0.48	1237	0.00%	0.00%	0.00%	33.15	No	0.00%	0.00%	0.00%	34.02	No	0.00%	0.00%	0.00%	141.74	No
49	15	0.47	520	-14.67%	-38.58%	97.12%	33.01	Yes	-14.67%	-38.58%	97.12%	34.37	Yes	-14.67%	-38.58%	97.12%	147.38	Yes

[8] TT = percentual difference in transit time; TR = percentual difference in risk, TC = percentual difference in cost

Table 37:Experimental results of deterministically solving the test instances with the risk-greedy ($\beta = 1$) model.

	Initial solution			Tabu search M1					Tabu search M2					MNTS				
Experiment	Transit time	Total risk	Total cost	% TT	%TR	%TC	Run time	Change	% TT	%TR	%TC	Run time	Change	% TT	%TR	%TC	Run time	Change
1	40.2	0.63	1291	233.58%	-77.22%	327.11%	38.91	Yes	233.58%	-77.22%	327.11%	43.86	Yes	42.04%	-83.50%	422.77%	156.58	Yes
2	15	0.25	808	218.00%	-48.35%	138.24%	39.00	No	218.00%	-48.35%	138.24%	40.39	No	304.00%	-56.89%	782.43%	154.68	Yes
3	10.6	0.18	631	508.49%	-39.99%	1140.57%	41.63	No	508.49%	-39.99%	1140.57%	46.31	No	508.49%	-39.99%	1140.57%	155.02	Yes
4	14.1	0.21	677	333.33%	-49.75%	967.36%	42.71	No	333.33%	-49.75%	967.36%	48.05	No	333.33%	-49.75%	967.36%	155.88	Yes
5	17.5	0.19	270	248.57%	-42.32%	2564.44%	43.77	Yes	248.57%	-42.32%	2564.44%	48.66	Yes	248.57%	-42.32%	2564.44%	158.33	Yes
6	10.1	0.21	677	504.95%	-49.75%	967.36%	42.98	No	504.95%	-49.75%	967.36%	44.16	No	504.95%	-49.75%	967.36%	156.16	Yes
7	11.3	0.30	547	362.83%	-56.86%	355.21%	43.27	No	362.83%	-56.86%	355.21%	44.74	No	409.73%	-65.16%	1105.12%	159.89	Yes
8	11.3	0.30	547	362.83%	-56.86%	355.21%	44.42	No	362.83%	-56.86%	355.21%	46.98	No	409.73%	-65.16%	1105.12%	159.58	Yes
9	11.3	0.30	547	362.83%	-56.86%	355.21%	43.48	No	362.83%	-56.86%	355.21%	49.88	No	409.73%	-65.16%	1105.12%	161.04	Yes
10	10.1	0.21	677	504.95%	-49.75%	967.36%	42.06	No	504.95%	-49.75%	967.36%	44.74	No	504.95%	-49.75%	967.36%	155.72	Yes
11	10.1	0.21	677	504.95%	-49.75%	967.36%	42.20	No	504.95%	-49.75%	967.36%	48.96	No	504.95%	-49.75%	967.36%	156.13	Yes
12	9.1	0.21	677	571.43%	-49.75%	967.36%	42.85	No	571.43%	-49.75%	967.36%	46.19	No	571.43%	-49.75%	967.36%	158.70	Yes
13	15.2	0.47	534	277.63%	-77.99%	1174.72%	32.79	Yes	277.63%	-77.99%	1174.72%	34.17	Yes	277.63%	-77.99%	1174.72%	147.47	Yes
14	9.1	0.21	677	571.43%	-49.75%	967.36%	43.74	No	571.43%	-49.75%	967.36%	48.57	No	571.43%	-49.75%	967.36%	157.13	Yes
15	23.5	0.63	948	413.62%	-76.98%	510.23%	38.51	Yes	413.62%	-76.98%	510.23%	44.65	Yes	149.79%	-83.31%	640.51%	153.64	Yes
16	16.9	0.48	1237	239.64%	-78.13%	449.64%	32.84	Yes	239.64%	-78.13%	449.64%	37.36	Yes	239.64%	-78.13%	449.64%	148.64	Yes
17	17.4	0.48	1237	229.89%	-78.26%	445.11%	32.62	Yes	229.89%	-78.26%	445.11%	36.60	Yes	229.89%	-78.26%	445.11%	147.02	Yes
18	37.6	0.48	888	56.38%	-77.99%	689.64%	32.90	Yes	56.38%	-77.99%	689.64%	36.58	Yes	56.38%	-77.99%	689.64%	147.12	Yes
19	16.3	0.47	538	250.92%	-78.02%	1158.18%	33.73	Yes	250.92%	-78.02%	1158.18%	38.83	Yes	250.92%	-78.02%	1158.18%	148.02	Yes
20	11.5	0.47	525	399.13%	-77.94%	1184.38%	33.09	No	399.13%	-77.94%	1184.38%	39.41	No	399.13%	-77.94%	1184.38%	148.65	Yes
21	88.5	0.47	538	-35.37%	-78.02%	1158.18%	33.42	Yes	-35.37%	-78.02%	1158.18%	41.88	Yes	-35.37%	-78.02%	1158.18%	143.87	Yes
22	12.8	0.26	506	332.81%	-60.12%	1154.55%	39.98	Yes	332.81%	-60.12%	1154.55%	48.19	Yes	332.81%	-60.12%	1154.55%	151.38	Yes
23	15.3	0.47	525	274.51%	-77.83%	1185.52%	32.94	Yes	274.51%	-77.83%	1185.52%	39.30	Yes	274.51%	-77.83%	1185.52%	143.16	Yes
24	15.2	0.47	536	274.34%	-78.02%	1156.16%	33.08	Yes	274.34%	-78.02%	1156.16%	37.43	Yes	274.34%	-78.02%	1156.16%	143.28	Yes
25	11.5	0.31	430	503.48%	-59.85%	858.37%	41.33	No	503.48%	-59.85%	858.37%	50.22	No	400.87%	-65.94%	1418.14%	152.60	Yes
26	19.9	0.48	380	242.21%	-74.37%	1133.95%	38.27	Yes	242.21%	-74.37%	1133.95%	44.43	Yes	197.49%	-78.30%	1767.37%	149.51	Yes
27	34.3	0.47	525	67.93%	-77.77%	1196.19%	32.88	Yes	67.93%	-77.77%	1196.19%	37.38	Yes	67.93%	-77.77%	1196.19%	144.93	Yes
28	35.3	0.47	525	62.04%	-77.90%	1189.90%	33.11	Yes	62.04%	-77.90%	1189.90%	37.30	Yes	62.04%	-77.90%	1189.90%	144.95	Yes
29	13.4	0.34	649	448.51%	-63.01%	547.30%	41.17	No	448.51%	-63.01%	547.30%	50.04	No	330.60%	-68.61%	918.18%	153.38	Yes
30	18.1	0.50	1390	218.78%	-78.94%	392.59%	33.15	Yes	218.78%	-78.94%	392.59%	39.45	Yes	218.78%	-78.94%	392.59%	144.08	Yes
31	13.1	0.26	1142	335.88%	-58.84%	471.37%	40.41	Yes	335.88%	-58.84%	471.37%	46.43	Yes	335.88%	-58.84%	471.37%	151.52	Yes
32	13.5	0.25	379	323.70%	-57.90%	1609.50%	40.89	Yes	323.70%	-57.90%	1609.50%	45.72	Yes	323.70%	-57.90%	1609.50%	152.20	Yes
33	16.2	0.25	795	645.68%	-48.07%	139.25%	38.27	No	645.68%	-48.07%	139.25%	46.78	No	275.31%	-56.58%	793.58%	152.45	Yes
34	33.7	0.47	525	70.33%	-77.94%	1184.38%	33.77	Yes	70.33%	-77.94%	1184.38%	37.94	Yes	70.33%	-77.94%	1184.38%	145.46	Yes
35	15.2	0.49	678	279.61%	-78.63%	909.88%	33.58	Yes	279.61%	-78.63%	909.88%	44.78	Yes	279.61%	-78.63%	909.88%	145.64	Yes
36	27.2	0.63	1210	545.96%	-76.89%	437.52%	38.33	Yes	545.96%	-76.89%	437.52%	47.51	Yes	134.19%	-83.19%	539.59%	148.87	Yes
37	131.4	0.47	534	-56.32%	-77.99%	1174.72%	32.99	Yes	-56.32%	-77.99%	1174.72%	41.08	Yes	-56.32%	-77.99%	1174.72%	148.61	Yes
38	35.4	0.49	678	62.99%	-78.63%	909.88%	33.27	Yes	62.99%	-78.63%	909.88%	40.65	Yes	62.99%	-78.63%	909.88%	147.09	Yes
39	15	0.47	525	282.67%	-77.80%	1195.05%	32.99	Yes	282.67%	-77.80%	1195.05%	40.87	Yes	282.67%	-77.80%	1195.05%	146.16	Yes
40	128.4	0.49	708	-55.14%	-78.77%	864.69%	33.35	Yes	-55.14%	-78.77%	864.69%	37.90	Yes	-55.14%	-78.77%	864.69%	146.54	Yes
41	133.9	0.63	981	-17.55%	-77.17%	491.74%	38.38	Yes	-17.55%	-77.17%	491.74%	45.15	Yes	-55.64%	-83.44%	617.64%	151.92	Yes
42	16.4	0.32	1375	444.51%	-60.37%	256.80%	37.60	No	444.51%	-60.37%	256.80%	42.99	No	276.22%	-66.34%	431.85%	150.01	Yes
43	31.7	0.47	525	81.07%	-77.94%	1184.38%	33.66	Yes	81.07%	-77.94%	1184.38%	38.64	Yes	81.07%	-77.94%	1184.38%	147.29	Yes
44	31.4	0.20	342	93.63%	-46.37%	1995.91%	39.17	Yes	93.63%	-46.37%	1995.91%	48.09	Yes	93.63%	-46.37%	1995.91%	150.92	Yes
45	15.5	0.49	1362	274.19%	-78.68%	405.43%	33.38	No	274.19%	-78.68%	405.43%	36.95	No	274.19%	-78.68%	405.43%	147.92	Yes
46	13.2	0.25	1155	333.33%	-58.10%	465.19%	41.56	Yes	333.33%	-58.10%	465.19%	51.65	Yes	333.33%	-58.10%	465.19%	153.84	Yes
47	16.1	0.47	525	257.14%	-77.90%	1199.43%	33.29	Yes	257.14%	-77.90%	1199.43%	43.32	Yes	257.14%	-77.90%	1199.43%	145.71	Yes
48	15.2	0.48	1237	277.63%	-78.26%	445.11%	34.02	No	277.63%	-78.26%	445.11%	42.56	No	277.63%	-78.26%	445.11%	146.46	Yes
49	15	0.47	520	282.67%	-77.71%	1205.77%	33.24	Yes	282.67%	-77.71%	1205.77%	39.32	Yes	282.67%	-77.71%	1205.77%	152.30	Yes

Table 38: Experimental results of deterministically solving the test instances with the cost-greedy ($\gamma = 1$) model.

	Initial solution			Tabu search M1					Tabu search M2					MNTS				
Experiment	Transit time	Total risk	Total cost	% TT	%TR	%TC	Run time	Change	% TT	%TR	%TC	Run time	Change	% TT	%TR	%TC	Run time	Change
1	40.2	0.63	1291	73.13%	-50.09%	-54.76%	40.29	Yes	73.13%	-50.09%	-54.76%	44.37	Yes	-20.40%	-57.77%	-62.59%	170.46	Yes
2	15	0.25	808	206.00%	-9.18%	-50.62%	40.61	Yes	206.00%	-9.18%	-50.62%	45.90	Yes	75.33%	-20.92%	-57.18%	159.84	Yes
3	10.6	0.18	631	0.00%	0.00%	0.00%	42.83	No	0.00%	0.00%	0.00%	47.21	No	0.00%	0.00%	0.00%	161.73	No
4	14.1	0.21	677	0.00%	0.00%	0.00%	42.69	No	0.00%	0.00%	0.00%	48.55	No	0.00%	0.00%	0.00%	161.68	No
5	17.5	0.19	270	0.00%	0.00%	0.00%	43.48	No	0.00%	0.00%	0.00%	48.99	No	0.00%	0.00%	0.00%	163.33	No
6	10.1	0.21	677	0.00%	0.00%	0.00%	42.71	No	0.00%	0.00%	0.00%	47.83	No	0.00%	0.00%	0.00%	160.19	No
7	11.3	0.30	547	828.32%	-2.72%	-32.91%	46.30	Yes	828.32%	-2.72%	-32.91%	50.82	Yes	124.78%	-24.01%	-42.23%	167.41	Yes
8	11.3	0.30	547	828.32%	-2.72%	-32.91%	46.00	Yes	828.32%	-2.72%	-32.91%	51.78	Yes	124.78%	-24.01%	-42.23%	166.69	Yes
9	11.3	0.30	547	828.32%	-2.72%	-32.91%	46.64	Yes	828.32%	-2.72%	-32.91%	53.54	Yes	124.78%	-24.01%	-42.23%	166.39	Yes
10	10.1	0.21	677	0.00%	0.00%	0.00%	42.49	No	0.00%	0.00%	0.00%	49.73	No	0.00%	0.00%	0.00%	161.25	No
11	10.1	0.21	677	0.00%	0.00%	0.00%	42.24	No	0.00%	0.00%	0.00%	50.01	No	0.00%	0.00%	0.00%	163.47	No
12	9.1	0.21	677	0.00%	0.00%	0.00%	42.69	No	0.00%	0.00%	0.00%	45.03	No	0.00%	0.00%	0.00%	161.46	No
13	15.2	0.47	534	13.16%	-46.63%	-35.02%	33.67	Yes	13.16%	-46.63%	-35.02%	33.87	Yes	13.16%	-46.63%	-35.02%	153.76	Yes
14	9.1	0.21	677	0.00%	0.00%	0.00%	43.74	No	0.00%	0.00%	0.00%	43.91	No	0.00%	0.00%	0.00%	162.63	No
15	23.5	0.63	948	208.51%	-49.68%	-60.34%	39.31	Yes	208.51%	-49.68%	-60.34%	40.27	Yes	20.85%	-61.66%	-70.99%	156.86	Yes
16	16.9	0.48	1237	0.59%	-47.81%	-72.68%	33.68	Yes	0.59%	-47.81%	-72.68%	34.71	Yes	0.59%	-47.81%	-72.68%	154.57	Yes
17	17.4	0.48	1237	0.57%	-47.56%	-72.68%	33.50	Yes	0.57%	-47.56%	-72.68%	34.42	Yes	0.57%	-47.56%	-72.68%	154.89	Yes
18	37.6	0.48	888	91.76%	-48.31%	-24.89%	33.66	Yes	91.76%	-48.31%	-24.89%	35.33	Yes	91.76%	-48.31%	-24.89%	161.05	Yes
19	16.3	0.47	538	12.27%	-46.59%	-34.76%	33.54	Yes	12.27%	-46.59%	-34.76%	34.25	Yes	12.27%	-46.59%	-34.76%	154.17	Yes
20	11.5	0.47	525	226.09%	-46.78%	-35.62%	33.69	Yes	226.09%	-46.78%	-35.62%	34.81	Yes	226.09%	-46.78%	-35.62%	154.97	Yes
21	88.5	0.47	538	-29.60%	-46.59%	-34.76%	34.03	Yes	-29.60%	-46.59%	-34.76%	37.56	Yes	-29.60%	-46.59%	-34.76%	154.98	Yes
22	12.8	0.26	506	0.00%	0.00%	0.00%	40.46	No	0.00%	0.00%	0.00%	45.68	No	0.00%	0.00%	0.00%	164.88	No
23	15.3	0.47	525	13.07%	-47.01%	-35.62%	33.54	Yes	13.07%	-47.01%	-35.62%	37.29	Yes	13.07%	-47.01%	-35.62%	154.63	Yes
24	15.2	0.47	536	13.16%	-46.63%	-34.89%	33.97	Yes	13.16%	-46.63%	-34.89%	38.60	Yes	13.16%	-46.63%	-34.89%	155.26	Yes
25	11.5	0.31	430	0.00%	0.00%	0.00%	42.84	No	0.00%	0.00%	0.00%	48.72	No	26.09%	-18.74%	-11.86%	163.29	Yes
26	19.9	0.48	380	0.00%	0.00%	0.00%	39.72	No	0.00%	0.00%	0.00%	43.75	No	0.00%	0.00%	0.00%	161.45	No
27	34.3	0.47	525	-6.41%	-47.08%	-35.62%	34.43	Yes	-6.41%	-47.08%	-35.62%	42.24	Yes	-6.41%	-47.08%	-35.62%	155.48	Yes
28	35.3	0.47	525	-6.23%	-46.84%	-35.62%	33.73	Yes	-6.23%	-46.84%	-35.62%	45.06	Yes	-6.23%	-46.84%	-35.62%	155.24	Yes
29	13.4	0.34	649	0.00%	0.00%	0.00%	42.32	No	0.00%	0.00%	0.00%	48.58	No	44.78%	-18.46%	-7.86%	165.22	Yes
30	18.1	0.50	1390	0.55%	-46.01%	-64.68%	33.56	Yes	0.55%	-46.01%	-64.68%	38.01	Yes	0.55%	-46.01%	-64.68%	155.14	Yes
31	13.1	0.26	1142	29.77%	-3.82%	-66.81%	40.24	Yes	29.77%	-3.82%	-66.81%	47.49	Yes	29.77%	-3.82%	-66.81%	162.97	Yes
32	13.5	0.25	379	0.00%	0.00%	0.00%	40.40	No	0.00%	0.00%	0.00%	51.05	No	0.00%	0.00%	0.00%	163.99	No
33	16.2	0.25	795	190.74%	-9.25%	-51.45%	39.93	Yes	190.74%	-9.25%	-51.45%	47.10	Yes	69.75%	-21.08%	-58.11%	162.96	Yes
34	33.7	0.47	525	-6.53%	-46.78%	-35.62%	33.46	Yes	-6.53%	-46.78%	-35.62%	37.23	Yes	-6.53%	-46.78%	-35.62%	156.62	Yes
35	15.2	0.49	678	13.16%	-45.23%	-27.58%	33.60	Yes	13.16%	-45.23%	-27.58%	38.22	Yes	13.16%	-45.23%	-27.58%	156.35	Yes
36	27.2	0.63	1210	356.62%	-49.46%	-47.27%	39.27	Yes	356.62%	-49.46%	-47.27%	46.30	Yes	106.25%	-61.38%	-55.62%	157.56	Yes
37	131.4	0.47	534	-38.20%	-46.63%	-35.02%	33.50	Yes	-38.20%	-46.63%	-35.02%	36.22	Yes	-38.20%	-46.63%	-35.02%	159.86	Yes
38	35.4	0.49	678	-6.21%	-45.23%	-27.58%	33.89	Yes	-6.21%	-45.23%	-27.58%	38.76	Yes	-6.21%	-45.23%	-27.58%	156.87	Yes
39	15	0.47	525	13.33%	-47.03%	-35.62%	33.88	Yes	13.33%	-47.03%	-35.62%	37.76	Yes	13.33%	-47.03%	-35.62%	156.15	Yes
40	128.4	0.49	708	-39.10%	-44.96%	-26.41%	33.86	Yes	-39.10%	-44.96%	-26.41%	37.41	Yes	-39.10%	-44.96%	-26.41%	160.12	Yes
41	133.9	0.63	981	0.00%	-49.26%	-58.31%	39.59	Yes	0.00%	-49.26%	-58.31%	42.58	Yes	20.84%	-61.13%	-68.60%	166.00	Yes
42	16.4	0.32	1375	190.85%	-24.74%	-60.65%	40.26	Yes	190.85%	-24.74%	-60.65%	42.91	Yes	71.34%	-33.88%	-64.51%	160.73	Yes
43	31.7	0.47	525	-6.94%	-46.78%	-35.62%	33.84	Yes	-6.94%	-46.78%	-35.62%	34.46	Yes	-6.94%	-46.78%	-35.62%	156.38	Yes
44	31.4	0.20	342	0.00%	0.00%	0.00%	38.80	No	0.00%	0.00%	0.00%	39.65	No	0.00%	0.00%	0.00%	163.27	No
45	15.5	0.49	1362	273.55%	-1.44%	-52.28%	34.02	Yes	273.55%	-1.44%	-52.28%	34.72	Yes	219.35%	3.01%	-58.15%	157.59	Yes
46	13.2	0.25	1155	23.48%	-3.89%	-66.06%	41.34	Yes	23.48%	-3.89%	-66.06%	42.12	Yes	23.48%	-3.89%	-66.06%	164.73	Yes
47	16.1	0.47	525	12.42%	-46.80%	-35.62%	33.31	Yes	12.42%	-46.80%	-35.62%	33.76	Yes	12.42%	-46.80%	-35.62%	154.40	Yes
48	15.2	0.48	1237	133.55%	-47.56%	-72.68%	33.64	Yes	133.55%	-47.56%	-72.68%	38.52	Yes	133.55%	-47.56%	-72.68%	157.57	Yes
49	15	0.47	520	0.00%	0.00%	0.00%	33.70	No	0.00%	0.00%	0.00%	35.91	No	264.00%	4.66%	-15.38%	155.16	Yes

Table 39: Experimental results of deterministically solving the test instances with the balanced model.

	Initial solution			Tabu search M1					Tabu search M2					MNTS				
Experiment	Transit time	Total risk	Total cost	% TT	%TR	%TC	Run time	Change	% TT	%TR	%TC	Run time	Change	% TT	%TR	%TC	Run time	Change
1	40.2	0.63	1291	-29.60%	-43.27%	-54.53%	40.38	Yes	-29.60%	-43.27%	-54.53%	41.67	Yes	-20.40%	-57.77%	-62.59%	164.92	Yes
2	15	0.25	808	46.00%	-11.25%	-50.62%	41.30	Yes	46.00%	-11.25%	-50.62%	41.64	Yes	28.67%	-20.86%	-57.18%	163.09	Yes
3	10.6	0.18	631	0.00%	0.00%	0.00%	41.99	No	0.00%	0.00%	0.00%	44.30	No	0.00%	0.00%	0.00%	162.60	No
4	14.1	0.21	677	0.00%	0.00%	0.00%	42.34	No	0.00%	0.00%	0.00%	48.97	No	0.00%	0.00%	0.00%	163.91	No
5	17.5	0.19	270	0.00%	0.00%	0.00%	43.84	No	0.00%	0.00%	0.00%	48.65	No	0.00%	0.00%	0.00%	166.13	No
6	10.1	0.21	677	0.00%	0.00%	0.00%	42.68	No	0.00%	0.00%	0.00%	48.04	No	0.00%	0.00%	0.00%	165.08	No
7	11.3	0.30	547	151.33%	-3.40%	-32.91%	45.87	Yes	151.33%	-3.40%	-32.91%	51.13	Yes	124.78%	-24.01%	-42.23%	173.49	Yes
8	11.3	0.30	547	151.33%	-3.40%	-32.91%	46.13	Yes	151.33%	-3.40%	-32.91%	50.86	Yes	124.78%	-24.01%	-42.23%	169.94	Yes
9	11.3	0.30	547	151.33%	-3.40%	-32.91%	46.60	Yes	151.33%	-3.40%	-32.91%	49.98	Yes	124.78%	-24.01%	-42.23%	170.20	Yes
10	10.1	0.21	677	0.00%	0.00%	0.00%	43.69	No	0.00%	0.00%	0.00%	48.56	No	0.00%	0.00%	0.00%	165.26	No
11	10.1	0.21	677	0.00%	0.00%	0.00%	42.74	No	0.00%	0.00%	0.00%	47.49	No	0.00%	0.00%	0.00%	164.19	No
12	9.1	0.21	677	0.00%	0.00%	0.00%	42.62	No	0.00%	0.00%	0.00%	46.57	No	0.00%	0.00%	0.00%	164.43	No
13	15.2	0.47	534	13.16%	-46.63%	-35.02%	33.31	Yes	13.16%	-46.63%	-35.02%	35.14	Yes	13.16%	-46.63%	-35.02%	157.66	Yes
14	9.1	0.21	677	0.00%	0.00%	0.00%	42.41	No	0.00%	0.00%	0.00%	46.48	No	0.00%	0.00%	0.00%	175.12	No
15	23.5	0.63	948	4.26%	-47.54%	-60.02%	39.68	Yes	4.26%	-47.54%	-60.02%	44.80	Yes	20.85%	-61.66%	-70.99%	159.71	Yes
16	16.9	0.48	1237	0.59%	-47.81%	-72.68%	33.05	Yes	0.59%	-47.81%	-72.68%	36.47	Yes	0.59%	-47.81%	-72.68%	157.47	Yes
17	17.4	0.48	1237	0.57%	-47.56%	-72.68%	33.37	Yes	0.57%	-47.56%	-72.68%	37.50	Yes	0.57%	-47.56%	-72.68%	159.80	Yes
18	37.6	0.48	888	-5.85%	-46.47%	-21.06%	33.40	Yes	-5.85%	-46.47%	-21.06%	39.96	Yes	17.55%	-41.18%	-23.54%	158.49	Yes
19	16.3	0.47	538	12.27%	-46.59%	-34.76%	33.30	Yes	12.27%	-46.59%	-34.76%	40.23	Yes	12.27%	-46.59%	-34.76%	159.42	Yes
20	11.5	0.47	525	226.09%	-46.78%	-35.62%	33.67	Yes	226.09%	-46.78%	-35.62%	34.70	Yes	226.09%	-46.78%	-35.62%	159.07	Yes
21	88.5	0.47	538	-29.60%	-46.59%	-34.76%	33.38	Yes	-29.60%	-46.59%	-34.76%	37.47	Yes	-29.60%	-46.59%	-34.76%	159.40	Yes
22	12.8	0.26	506	0.00%	0.00%	0.00%	40.20	No	0.00%	0.00%	0.00%	58.60	No	0.00%	0.00%	0.00%	165.90	No
23	15.3	0.47	525	13.07%	-47.01%	-35.62%	33.59	Yes	13.07%	-47.01%	-35.62%	42.57	Yes	13.07%	-47.01%	-35.62%	157.69	Yes
24	15.2	0.47	536	13.16%	-46.63%	-34.89%	33.88	Yes	13.16%	-46.63%	-34.89%	38.28	Yes	13.16%	-46.63%	-34.89%	158.71	Yes
25	11.5	0.31	430	0.00%	0.00%	0.00%	42.55	No	0.00%	0.00%	0.00%	47.46	No	26.09%	-18.74%	-11.86%	165.93	Yes
26	19.9	0.48	380	0.00%	0.00%	0.00%	39.37	No	0.00%	0.00%	0.00%	43.66	No	0.00%	0.00%	0.00%	161.63	No
27	34.3	0.47	525	-6.41%	-47.08%	-35.62%	33.52	Yes	-6.41%	-47.08%	-35.62%	36.19	Yes	-6.41%	-47.08%	-35.62%	158.69	Yes
28	35.3	0.47	525	-6.23%	-46.84%	-35.62%	33.87	Yes	-6.23%	-46.84%	-35.62%	36.96	Yes	-6.23%	-46.84%	-35.62%	159.62	Yes
29	13.4	0.34	649	0.00%	0.00%	0.00%	42.63	No	0.00%	0.00%	0.00%	46.98	No	44.78%	-18.46%	-7.86%	167.28	Yes
30	18.1	0.50	1390	0.55%	-46.01%	-64.68%	33.14	Yes	0.55%	-46.01%	-64.68%	37.01	Yes	0.55%	-46.01%	-64.68%	159.10	Yes
31	13.1	0.26	1142	29.77%	-3.82%	-66.81%	40.05	Yes	29.77%	-3.82%	-66.81%	45.35	Yes	29.77%	-3.82%	-66.81%	165.83	Yes
32	13.5	0.25	379	0.00%	0.00%	0.00%	40.31	No	0.00%	0.00%	0.00%	48.54	No	0.00%	0.00%	0.00%	167.38	No
33	16.2	0.25	795	42.59%	-11.33%	-51.45%	40.04	Yes	42.59%	-11.33%	-51.45%	43.41	Yes	26.54%	-21.02%	-58.11%	162.50	Yes
34	33.7	0.47	525	-6.53%	-46.78%	-35.62%	33.52	Yes	-6.53%	-46.78%	-35.62%	34.91	Yes	-6.53%	-46.78%	-35.62%	157.66	Yes
35	15.2	0.49	678	13.16%	-45.23%	-27.58%	33.95	Yes	13.16%	-45.23%	-27.58%	35.65	Yes	13.16%	-45.23%	-27.58%	159.35	Yes
36	27.2	0.63	1210	3.68%	-47.32%	-47.02%	39.82	Yes	3.68%	-47.32%	-47.02%	43.84	Yes	16.54%	-55.89%	-54.63%	161.17	Yes
37	131.4	0.47	534	-38.20%	-46.63%	-35.02%	33.46	Yes	-38.20%	-46.63%	-35.02%	37.05	Yes	-38.20%	-46.63%	-35.02%	158.15	Yes
38	35.4	0.49	678	-6.21%	-45.23%	-27.58%	33.99	Yes	-6.21%	-45.23%	-27.58%	36.01	Yes	-6.21%	-45.23%	-27.58%	158.78	Yes
39	15	0.47	525	13.33%	-47.03%	-35.62%	33.85	Yes	13.33%	-47.03%	-35.62%	36.83	Yes	13.33%	-47.03%	-35.62%	157.67	Yes
40	128.4	0.49	708	-39.10%	-44.96%	-26.41%	33.51	Yes	-39.10%	-44.96%	-26.41%	38.22	Yes	-39.10%	-44.96%	-26.41%	161.65	Yes
41	133.9	0.63	981	-36.59%	-47.05%	-58.00%	40.30	Yes	-36.59%	-47.05%	-58.00%	44.67	Yes	-35.85%	-55.77%	-67.38%	164.29	Yes
42	16.4	0.32	1375	44.51%	-26.35%	-60.65%	39.59	Yes	44.51%	-26.35%	-60.65%	44.32	Yes	28.66%	-33.84%	-64.51%	161.31	Yes
43	31.7	0.47	525	-6.94%	-46.78%	-35.62%	35.22	Yes	-6.94%	-46.78%	-35.62%	38.52	Yes	-6.94%	-46.78%	-35.62%	160.63	Yes
44	31.4	0.20	342	-45.22%	0.00%	0.00%	39.55	Yes	-45.22%	0.00%	0.00%	42.99	Yes	-45.22%	0.00%	0.00%	162.73	Yes
45	15.5	0.49	1362	72.26%	-1.76%	-52.28%	33.86	Yes	72.26%	-1.76%	-52.28%	38.21	Yes	64.52%	2.88%	-58.15%	156.86	Yes
46	13.2	0.25	1155	23.48%	-3.89%	-66.06%	41.10	Yes	23.48%	-3.89%	-66.06%	46.72	Yes	23.48%	-3.89%	-66.06%	164.36	Yes
47	16.1	0.47	525	12.42%	-46.80%	-35.62%	33.38	Yes	12.42%	-46.80%	-35.62%	36.87	Yes	12.42%	-46.80%	-35.62%	160.06	Yes
48	15.2	0.48	1237	133.55%	-47.56%	-72.68%	34.87	Yes	133.55%	-47.56%	-72.68%	38.19	Yes	133.55%	-47.56%	-72.68%	158.06	Yes
49	15	0.47	520	0.00%	0.00%	0.00%	33.96	No	0.00%	0.00%	0.00%	37.70	No	14.00%	4.45%	-15.38%	155.61	Yes

Table 40: Experimental results of deterministically solving the test instances with the weighted model.

	Initial solution			Tabu search M1					Tabu search M2					MNTS				
Experiment	Transit time	Total risk	Total cost	% TT	%TR	%TC	Run time	Change	% TT	%TR	%TC	Run time	Change	% TT	%TR	%TC	Run time	Change
1	40.2	0.63	1291	-29.60%	-43.27%	-54.53%	41.12	Yes	-29.60%	-43.27%	-54.53%	44.17	Yes	-29.60%	-52.60%	-61.66%	163.03	Yes
2	15	0.25	808	46.00%	-11.25%	-50.62%	40.55	Yes	46.00%	-11.25%	-50.62%	47.06	Yes	28.67%	-20.86%	-57.18%	165.63	Yes
3	10.6	0.18	631	0.00%	0.00%	0.00%	42.38	No	0.00%	0.00%	0.00%	62.56	No	0.00%	0.00%	0.00%	163.27	No
4	14.1	0.21	677	0.00%	0.00%	0.00%	42.66	No	0.00%	0.00%	0.00%	57.37	No	0.00%	0.00%	0.00%	165.56	No
5	17.5	0.19	270	0.00%	0.00%	0.00%	43.60	No	0.00%	0.00%	0.00%	50.65	No	0.00%	0.00%	0.00%	168.10	No
6	10.1	0.21	677	0.00%	0.00%	0.00%	43.45	No	0.00%	0.00%	0.00%	47.20	No	0.00%	0.00%	0.00%	164.91	No
7	11.3	0.30	547	121.24%	-11.06%	-40.04%	46.21	Yes	121.24%	-11.06%	-40.04%	50.72	Yes	107.00%	-24.01%	-42.23%	171.17	Yes
8	11.3	0.30	547	121.24%	-11.06%	-40.04%	46.90	Yes	121.24%	-11.06%	-40.04%	52.01	Yes	124.78%	-24.01%	-42.23%	172.16	Yes
9	11.3	0.30	547	121.24%	-11.06%	-40.04%	46.29	Yes	121.24%	-11.06%	-40.04%	49.77	Yes	124.78%	-24.01%	-42.23%	172.79	Yes
10	10.1	0.21	677	0.00%	0.00%	0.00%	42.98	No	0.00%	0.00%	0.00%	49.91	No	0.00%	0.00%	0.00%	164.59	No
11	10.1	0.21	677	0.00%	0.00%	0.00%	42.45	No	0.00%	0.00%	0.00%	50.19	No	0.00%	0.00%	0.00%	164.28	No
12	9.1	0.21	677	0.00%	0.00%	0.00%	43.44	No	0.00%	0.00%	0.00%	48.67	No	0.00%	0.00%	0.00%	169.93	No
13	15.2	0.47	534	13.16%	-46.63%	-35.02%	33.27	Yes	13.16%	-46.63%	-35.02%	38.26	Yes	13.16%	-46.63%	-35.02%	156.10	Yes
14	9.1	0.21	677	0.00%	0.00%	0.00%	43.03	No	0.00%	0.00%	0.00%	49.20	No	0.00%	0.00%	0.00%	164.73	No
15	23.5	0.63	948	33.62%	-47.89%	-60.02%	39.40	Yes	33.62%	-47.89%	-60.02%	46.97	Yes	20.85%	-61.66%	-70.99%	161.68	Yes
16	16.9	0.48	1237	0.59%	-47.81%	-72.68%	33.53	Yes	0.59%	-47.81%	-72.68%	38.49	Yes	0.59%	-47.81%	-72.68%	157.62	Yes
17	17.4	0.48	1237	0.57%	-47.56%	-72.68%	33.71	Yes	0.57%	-47.56%	-72.68%	35.18	Yes	0.57%	-47.56%	-72.68%	155.58	Yes
18	37.6	0.48	888	-5.85%	-46.47%	-21.06%	33.42	Yes	-5.85%	-46.47%	-21.06%	35.16	Yes	-15.69%	-39.61%	-19.71%	158.07	Yes
19	16.3	0.47	538	12.27%	-46.59%	-34.76%	33.42	Yes	12.27%	-46.59%	-34.76%	34.71	Yes	12.27%	-46.59%	-34.76%	158.40	Yes
20	11.5	0.47	525	226.09%	-46.78%	-35.62%	33.94	Yes	226.09%	-46.78%	-35.62%	35.93	Yes	193.91%	-39.87%	-33.33%	158.88	Yes
21	88.5	0.47	538	-29.60%	-46.59%	-34.76%	33.67	Yes	-29.60%	-46.59%	-34.76%	37.30	Yes	-33.79%	-39.71%	-32.53%	158.65	Yes
22	12.8	0.26	506	0.00%	0.00%	0.00%	41.50	No	0.00%	0.00%	0.00%	41.86	No	0.00%	0.00%	0.00%	164.10	No
23	15.3	0.47	525	13.07%	-47.01%	-35.62%	33.36	Yes	13.07%	-47.01%	-35.62%	34.51	Yes	13.07%	-47.01%	-35.62%	158.14	Yes
24	15.2	0.47	536	13.16%	-46.63%	-34.89%	33.81	Yes	13.16%	-46.63%	-34.89%	39.34	Yes	13.16%	-46.63%	-34.89%	158.48	Yes
25	11.5	0.31	430	0.00%	0.00%	0.00%	43.02	No	0.00%	0.00%	0.00%	52.60	No	26.09%	-18.74%	-11.86%	164.55	Yes
26	19.9	0.48	380	0.00%	0.00%	0.00%	39.60	No	0.00%	0.00%	0.00%	46.70	No	0.00%	0.00%	0.00%	161.17	No
27	34.3	0.47	525	-6.41%	-47.08%	-35.62%	33.55	Yes	-6.41%	-47.08%	-35.62%	38.13	Yes	-17.20%	-40.13%	-33.33%	159.10	Yes
28	35.3	0.47	525	-6.23%	-46.84%	-35.62%	33.50	Yes	-6.23%	-46.84%	-35.62%	36.94	Yes	-16.71%	-39.92%	-33.33%	158.23	Yes
29	13.4	0.34	649	0.00%	0.00%	0.00%	42.38	No	0.00%	0.00%	0.00%	47.74	No	17.16%	-8.82%	-6.01%	164.65	Yes
30	18.1	0.50	1390	0.55%	-46.01%	-64.68%	33.21	Yes	0.55%	-46.01%	-64.68%	37.11	Yes	0.55%	-46.01%	-64.68%	157.10	Yes
31	13.1	0.26	1142	29.77%	-3.82%	-66.81%	40.27	Yes	29.77%	-3.82%	-66.81%	47.53	Yes	29.77%	-3.82%	-66.81%	165.00	Yes
32	13.5	0.25	379	0.00%	0.00%	0.00%	40.73	No	0.00%	0.00%	0.00%	47.32	No	0.00%	0.00%	0.00%	165.64	No
33	16.2	0.25	795	42.59%	-11.33%	-51.45%	39.70	Yes	42.59%	-11.33%	-51.45%	43.59	Yes	26.54%	-21.02%	-58.11%	160.78	Yes
34	33.7	0.47	525	-6.53%	-46.78%	-35.62%	33.57	Yes	-6.53%	-46.78%	-35.62%	38.68	Yes	-17.51%	-39.87%	-33.33%	158.03	Yes
35	15.2	0.49	678	13.16%	-45.23%	-27.58%	34.33	Yes	13.16%	-45.23%	-27.58%	40.35	Yes	13.16%	-45.23%	-27.58%	161.59	Yes
36	27.2	0.63	1210	29.04%	-47.96%	-47.02%	40.17	Yes	29.04%	-47.96%	-47.02%	41.65	Yes	16.54%	-55.89%	-54.63%	160.06	Yes
37	131.4	0.47	534	-83.64%	-44.14%	18.91%	33.79	Yes	-83.64%	-44.14%	18.91%	34.56	Yes	-80.67%	-39.11%	3.93%	155.53	Yes
38	35.4	0.49	678	-6.21%	-45.23%	-27.58%	33.80	Yes	-6.21%	-45.23%	-27.58%	34.71	Yes	-16.67%	-38.55%	-25.81%	158.22	Yes
39	15	0.47	525	13.33%	-47.03%	-35.62%	33.90	Yes	13.33%	-47.03%	-35.62%	34.80	Yes	13.33%	-47.03%	-35.62%	159.87	Yes
40	128.4	0.49	708	-85.59%	-42.56%	14.27%	33.60	Yes	-85.59%	-42.56%	14.27%	34.29	Yes	-82.55%	-37.71%	2.97%	157.66	Yes
41	133.9	0.63	981	-84.47%	-41.23%	-27.32%	40.10	Yes	-84.47%	-41.23%	-27.32%	40.92	Yes	-35.85%	-55.77%	-67.38%	162.86	Yes
42	16.4	0.32	1375	44.51%	-26.35%	-60.65%	40.41	Yes	44.51%	-26.35%	-60.65%	42.71	Yes	28.66%	-33.84%	-64.51%	162.50	Yes
43	31.7	0.47	525	-6.94%	-46.78%	-35.62%	33.79	Yes	-6.94%	-46.78%	-35.62%	34.03	Yes	-18.61%	-39.87%	-33.33%	158.07	Yes
44	31.4	0.20	342	-45.22%	0.00%	0.00%	39.25	Yes	-45.22%	0.00%	0.00%	45.22	Yes	-45.22%	0.00%	0.00%	162.27	Yes
45	15.5	0.49	1362	72.26%	-1.76%	-52.28%	33.97	Yes	72.26%	-1.76%	-52.28%	34.48	Yes	64.52%	2.88%	-58.15%	155.86	Yes
46	13.2	0.25	1155	23.48%	-3.89%	-66.06%	41.52	Yes	23.48%	-3.89%	-66.06%	41.36	Yes	23.48%	-3.89%	-66.06%	168.21	Yes
47	16.1	0.47	525	12.42%	-46.80%	-35.62%	33.66	Yes	12.42%	-46.80%	-35.62%	33.92	Yes	12.42%	-46.80%	-35.62%	159.35	Yes
48	15.2	0.48	1237	133.55%	-47.56%	-72.68%	33.39	Yes	133.55%	-47.56%	-72.68%	33.89	Yes	109.21%	-40.75%	-71.71%	158.74	Yes
49	15	0.47	520	0.00%	0.00%	0.00%	33.32	No	0.00%	0.00%	0.00%	34.53	No	14.00%	4.45%	-15.38%	156.92	Yes

I. Experimental outcomes of the simheuristic

This appendix presents all outcomes of the experiments carried out to evaluate the simheuristic, using both normal and multiple neighborhoods tabu search variants. Each table presents a comparison in performance between the two solutions for each problem instance. We thereby show the airports used by each route, being either major (MA), medium (ME) or minor (MI) hubs, the preferred carrier type, the average KPI values and run time.

Table 41: Simheuristic results for the time-greedy ($\alpha = 1$) model.

	Tabu search									Multiple neighborhoods tabu search								
Experiment	Origin type	Transit type(s)	Destination type	Carriers	Transit time	Delivery delay	Total transit delay	Total cost	Run time	Origin type	Transit type(s)	Destination type	Carriers	Transit time	Delivery delay	Total transit delay	Total cost	Run time
1	MI	ME, MA	MI	Mixed	43.72	25.52	82.96	3972.39	853.19	MI		MA	Vehicle	30.41	0.36	0.46	493.80	2526.85
2	MI	ME	MA	Mixed	20.02	2.93	14.75	1971.08	749.41	MA		ME	Flight	29.25	2.16	14.04	514.60	2451.39
3	MI		ME	Vehicle	25.71	4.02	12.60	2823.06	584.87	MA		ME	Flight	34.59	1.84	9.72	1036.40	2411.38
4	MA		ME	Flight	16.75	2.70	7.48	813.59	609.25	MA		ME	Flight	19.47	1.21	5.79	807.83	2409.49
5	MI		MI	Vehicle	23.49	4.89	12.83	1422.51	616.73	MA	ME	MI	Mixed	35.18	3.49	22.46	466.47	2440.51
6	MA		ME	Flight	16.91	6.82	23.38	810.26	649.23	MA		ME	Flight	29.65	2.67	30.35	1147.82	2435.43
7	MA	ME	MI	Mixed	37.14	15.65	58.50	1671.57	788.24	MA		ME	Flight	40.73	2.70	8.69	1145.66	2525.76
8	MA	ME	MI	Mixed	37.18	15.68	58.51	1673.55	890.05	MA		ME	Flight	97.88	13.34	91.83	79.03	2538.71
9	MA	ME	MI	Mixed	37.22	15.72	58.55	1677.47	869.55	MA		ME	Flight	99.75	13.67	94.47	79.56	2538.62
10	MA		ME	Flight	16.92	6.83	23.44	811.01	725.56	MA		ME	Flight	29.42	2.63	29.64	1137.44	2473.09
11	MA		ME	Flight	16.90	6.82	23.37	810.32	715.90	MA		ME	Flight	29.92	2.71	30.69	1151.98	2471.58
12	MA		ME	Flight	17.35	8.26	28.99	853.94	676.53	MA		ME	Flight	32.62	3.35	39.29	1272.59	2420.56
13	MA		ME	Flight	20.68	5.60	18.21	765.81	694.01	MA		ME	Flight	34.88	1.35	14.63	575.44	2350.25
14	MA		ME	Flight	17.32	8.23	28.84	852.59	743.16	MI	ME	MA	Mixed	32.26	3.29	38.85	1265.93	2465.25
15	MA	ME, MA	MI	Flight	48.30	24.84	184.03	1729.90	936.51	ME	MA	MI	Mixed	60.03	5.67	64.16	545.20	2474.40
16	ME		MA	Vehicle	15.91	2.87	9.76	1493.58	648.36	MI	ME	MA	Mixed	32.04	3.00	37.78	719.75	2367.56
17	ME		MA	Vehicle	16.60	3.04	10.05	1505.76	641.41	ME		MI	Vehicle	32.83	3.04	37.68	730.83	2444.14
18	MI		MA	Vehicle	22.26	3.08	9.45	2013.56	642.10	MI	ME	MA	Mixed	33.97	0.42	0.44	716.29	2479.96
19	MI		MA	Vehicle	19.77	2.71	9.08	1561.14	611.35	MI	ME, ME	MA	Mixed	24.19	1.44	8.71	460.90	2382.53
20	ME		MA	Vehicle	18.97	5.41	19.24	1203.91	651.89	MI	ME, ME	MA	Mixed	32.84	0.50	2.65	371.70	2428.60
21	MA		MA	Flight	21.04	6.91	25.09	856.79	689.48	MI	ME, MA	MA	Mixed	57.41	0.33	0.36	359.38	2467.40
22	MA		MA	Flight	30.99	1.36	5.29	603.87	753.71	MI	ME	MI	Mixed	49.63	0.66	0.67	767.28	2487.51
23	ME		MA	Vehicle	16.30	2.95	9.95	1501.42	731.43	ME		MA	Vehicle	25.73	1.87	11.80	479.21	2356.58
24	MA		MA	Flight	21.15	8.20	30.20	1341.22	749.99	MI	ME, MA	MA	Mixed	43.02	2.80	49.57	722.16	2405.46
25	MA	MA	MA	Flight	19.13	7.70	48.32	542.71	798.85	MA		MA	Vehicle	25.96	2.33	14.95	578.21	2478.18
26	MA	MA	MI	Mixed	35.74	12.40	42.78	1608.54	804.63	ME		MI	Vehicle	37.35	2.38	8.00	821.42	2442.34
27	MI		MA	Vehicle	19.50	2.65	9.03	1541.32	664.86	MI	ME	MA	Mixed	30.17	0.34	0.40	345.63	2455.99
28	MI		MA	Vehicle	19.57	2.72	9.12	1546.88	653.06	MI	ME	MA	Mixed	30.99	0.31	0.41	346.61	2425.70
29	MI	ME	MA	Mixed	23.09	5.85	25.15	1159.79	783.64	MA		MA	Vehicle	22.08	0.87	4.06	686.85	2383.15
30	MI		ME	Vehicle	22.38	3.19	9.38	2227.72	706.38	ME		MI	Vehicle	33.31	3.01	36.84	972.77	2394.37
31	ME		MA	Vehicle	16.71	3.26	11.42	1644.92	712.59	MA	MA	MA	Flight	22.67	1.73	20.67	543.16	2418.61
32	MA		MA	Flight	19.12	5.81	20.56	360.79	718.14	MI	ME	MA	Mixed	26.85	2.64	17.30	608.18	2417.76
33	MI	ME	MA	Mixed	20.33	3.04	14.84	1973.54	826.56	MA		ME	Flight	29.19	1.95	12.20	474.39	2466.96
34	MI		MA	Vehicle	20.08	2.81	9.16	1558.52	730.56	MI	ME, MA	MA	Mixed	27.89	0.59	3.88	379.37	2492.14
35	MA		MA	Flight	21.94	8.77	31.96	1103.42	676.66	MI	ME, MA	MA	Mixed	35.09	1.01	5.14	986.60	2480.74
36	MA	ME, MA	MI	Flight	35.12	8.09	53.70	1346.95	870.95	ME		MI	Vehicle	57.79	3.94	22.61	938.24	2363.08
37	MI		MA	Vehicle	19.65	2.69	9.04	1556.72	641.14	MI	ME	MA	Mixed	24.56	0.40	1.59	581.68	2512.62
38	MI		MA	Vehicle	20.84	2.87	9.19	1738.70	667.73	MI	MA	MA	Mixed	31.60	0.38	0.42	503.87	2387.50
39	MA		MA	Flight	20.98	8.23	30.38	1329.03	706.15	MI	ME, MA	MA	Mixed	41.51	2.56	46.24	693.98	2404.33
40	MA		MA	Flight	23.03	5.82	20.63	1153.66	745.87	MI	ME, MA	MA	Mixed	23.73	0.71	5.35	816.12	2464.97
41	MI	ME, MA	MI	Flight	30.57	9.84	41.82	1265.24	866.98	MI		MI	Vehicle	80.43	0.58	2.20	334.03	2557.32
42	MI	ME	MA	Mixed	20.95	3.06	14.98	2163.77	707.20	MA		MA	Vehicle	29.93	1.99	12.40	699.18	2424.97
43	MA		MA	Flight	22.34	5.83	20.74	981.63	658.33	MI	ME, MA	MA	Mixed	26.88	0.72	5.45	392.15	2534.32
44	MA		MA	Flight	24.24	5.38	16.41	1325.89	687.36	MI	MA	MA	Mixed	31.50	0.99	7.17	428.31	2572.71
45	MI		MA	Vehicle	20.48	2.82	9.20	1709.32	617.39	ME	ME	MA	Mixed	71.43	8.64	112.83	996.27	2477.90
46	MA		MA	Vehicle	21.44	7.27	24.64	789.35	667.93	MA		MA	Vehicle	32.19	3.65	45.69	632.34	2444.35
47	MI		MA	Vehicle	19.52	2.67	9.05	1538.28	600.15	MI	ME, ME	MA	Mixed	24.17	1.47	9.15	444.82	2341.72
48	MI		MA	Vehicle	20.04	2.78	9.11	1554.39	597.34	ME	ME	MA	Mixed	33.31	0.32	0.35	344.11	2418.13
49	MA		MA	Flight	20.90	8.16	30.13	1316.30	638.60	MI	MA	MA	Mixed	31.21	0.34	0.52	555.73	2437.20

Table 42: Simheuristic results for the risk-greedy ($\beta = 1$) model.

	Tabu search									Multiple neighborhoods tabu search								
Experiment	Origin type	Transit type(s)	Destination type	Carriers	Transit time	Delivery delay	Total transit delay	Total cost	Run time	Origin type	Transit type(s)	Destination type	Carriers	Transit time	Delivery delay	Total transit delay	Total cost	Run time
1	ME	ME, MA	MI	Flight	136.98	2.80	2.84	2199.29	853.19	ME	MA	MI	Flight	112.07	1.82	1.90	1061.42	2324.11
2	MA	ME	MA	Vehicle	53.10	10.20	33.75	5698.43	749.41	ME		MA	Vehicle	29.53	4.94	13.62	2741.48	2317.30
3	MA		ME	Vehicle	49.63	10.43	21.56	5579.94	584.87	MA	MA	ME	Mixed	54.71	6.61	21.07	3605.23	2265.26
4	ME		MI	Vehicle	32.46	3.51	9.41	2030.04	609.25	MA	MA	ME	Flight	128.98	1.45	16.49	2371.91	2322.71
5	MA		MI	Vehicle	48.89	6.41	16.14	4274.39	616.73	MA	MA	MI	Mixed	50.67	8.17	27.87	3117.72	2343.09
6	MA		ME	Vehicle	56.33	12.23	25.79	6918.37	649.23	MI	MA, MA	ME	Flight	81.35	9.45	45.23	4146.71	2358.75
7	MA	ME	MI	Vehicle	72.75	3.30	31.17	4888.23	788.24	ME		MI	Vehicle	26.08	4.59	10.51	1502.45	2462.75
8	MA	ME	MI	Vehicle	72.73	3.27	31.02	4889.38	890.05	ME		MI	Vehicle	26.21	4.72	10.68	1513.32	2451.41
9	MA	ME	MI	Vehicle	72.73	3.28	31.10	4891.66	869.55	ME		MI	Vehicle	25.98	4.49	10.43	1493.51	2432.52
10	MA		ME	Vehicle	56.40	12.30	25.89	6927.87	725.56	MI	MA, MA	ME	Flight	77.68	5.79	19.27	5358.90	2363.17
11	MA		ME	Vehicle	56.34	12.24	25.80	6920.36	715.90	MI	MA, MA	ME	Flight	77.68	5.78	18.92	5382.01	2366.49
12	MA		ME	Vehicle	56.38	12.28	25.88	6930.29	676.53	MI	MA, MA	ME	Flight	81.10	10.20	50.51	4129.35	2389.61
13	MA		MA	Flight	36.29	7.93	16.93	2440.27	694.01	ME	MA	MA	Flight	35.90	7.46	34.56	756.94	2329.00
14	MA		ME	Vehicle	56.36	12.26	25.83	6918.27	743.16	MI	MA, MA	ME	Flight	171.62	4.72	11.48	5094.25	2369.15
15	ME	ME, MA	MI	Flight	119.97	1.83	1.87	1834.11	936.51	ME	MA	MI	Flight	98.77	4.25	30.37	886.56	2392.95
16	MA		MA	Flight	36.06	7.90	16.92	2426.39	648.36	MI	MA	MI	Flight	75.97	3.59	6.32	4584.87	2275.85
17	MA		MA	Flight	36.93	8.19	17.28	2440.60	641.41	MI	MA	MI	Flight	75.50	3.12	5.80	4477.12	2385.49
18	MA		MA	Flight	48.60	2.17	2.54	2222.98	642.10	ME	MI, MA	MA	Mixed	72.52	1.85	22.63	4886.65	2295.09
19	MA		MA	Flight	35.88	6.45	13.34	2296.19	611.35	ME	MI, MA, MA	MA	Mixed	78.45	1.18	9.50	2085.40	2345.17
20	MA		MA	Flight	39.40	14.52	32.29	2705.35	651.89	ME	MI, MA, MA	MA	Mixed	74.78	2.15	16.33	2222.78	2348.05
21	ME		MA	Flight	68.36	5.44	12.55	653.73	689.48	ME	MI, MA, MA	MA	Mixed	98.31	1.03	8.41	2061.53	2346.85
22	MA		MA	Flight	49.00	1.79	2.16	2276.54	753.71	MA	MA, MA	MA	Mixed	73.08	1.16	1.51	785.51	2312.10
23	MA		MA	Flight	36.47	7.99	17.01	2430.07	731.43	ME	MI, MA, MA	MA	Mixed	77.55	1.26	9.80	2081.99	2348.66
24	MA		MA	Flight	36.37	7.98	16.96	2443.05	749.99	ME	MI, MA, MA	MA	Mixed	77.48	1.26	9.91	2092.44	2340.71
25	ME	MA	MA	Flight	47.27	11.97	40.55	1923.08	798.85	MA		MA	Flight	37.13	2.08	4.13	1010.53	2384.44
26	ME	MA	MI	Flight	93.90	2.20	2.24	1485.04	804.63	ME		MI	Vehicle	46.21	26.32	77.86	2505.15	2379.27
27	MA		MA	Flight	44.66	1.61	1.98	1819.80	664.86	ME	MI, MA	MA	Mixed	68.78	1.70	23.03	4556.56	2340.02
28	MA		MA	Flight	45.46	1.60	1.96	1832.68	653.06	ME	MI, MA	MA	Mixed	69.57	1.48	22.19	4475.78	2282.64
29	ME	MA	MA	Flight	44.88	4.81	13.47	1871.82	783.64	MA		MA	Flight	27.56	11.23	32.17	1554.29	2229.76
30	MA		MA	Flight	37.79	8.29	17.37	2651.47	706.38	MI	MA	MI	Flight	76.25	3.57	6.42	4643.05	2278.82
31	MI		ME	Vehicle	25.57	6.08	10.95	2224.88	712.59	MA	MA	MA	Flight	54.99	2.98	11.39	534.84	2345.92
32	MA		ME	Vehicle	57.42	12.02	24.48	7149.07	718.14	MA	MA	MA	Flight	57.24	8.40	38.78	576.84	2320.58
33	MA	MA	MA	Vehicle	53.54	10.44	33.98	5736.09	826.56	ME		MA	Vehicle	29.84	5.05	13.72	2752.30	2319.85
34	MA		MA	Flight	44.43	1.82	2.24	1826.01	730.56	ME	MA, MA	MA	Flight	91.75	1.15	1.50	1814.68	2287.59
35	MA		MA	Flight	38.42	9.90	21.12	2561.57	676.66	ME	MA	MA	Flight	42.51	13.95	66.63	952.94	2303.54
36	ME	ME, MA	MI	Flight	174.13	3.77	3.81	2149.45	870.95	ME	MA	MI	Flight	54.51	4.03	10.47	1116.57	2317.18
37	ME		MA	Flight	83.83	5.23	11.92	674.98	641.14	ME		MA	Flight	82.10	3.57	10.56	674.93	2351.31
38	MA		MA	Flight	46.29	1.92	2.31	1981.77	667.73	ME	MI, MA	MA	Mixed	70.21	1.81	23.17	4716.77	2302.19
39	MA		MA	Flight	36.14	7.99	17.11	2434.43	706.15	ME	MA	MA	Flight	39.57	11.35	54.26	695.89	2308.86
40	ME		MA	Flight	84.32	5.31	12.19	847.38	745.87	ME		MA	Flight	82.58	3.61	10.70	839.36	2310.94
41	ME	ME, MA	MI	Flight	105.20	2.91	10.31	1125.73	866.98	ME	ME, MA	MI	Flight	104.56	1.48	8.12	1135.89	2362.49
42	ME	ME	MA	Flight	60.34	1.92	2.89	2410.35	707.20	MA		MA	Flight	32.49	1.85	3.53	531.02	2320.90
43	MA		MA	Flight	42.43	1.84	2.30	1828.26	658.33	ME	MA, MA	MA	Flight	89.82	1.21	1.71	1821.85	2394.71
44	MA		MA	Flight	41.81	1.64	1.99	1415.72	687.36	ME	MA, MA	MI	Flight	134.92	2.56	3.04	4979.03	2413.14
45	MA		MA	Flight	40.81	17.45	39.18	3123.11	617.39	ME	MA	MA	Flight	66.67	19.80	96.83	1161.47	2314.34
46	MA		MA	Flight	41.69	14.04	31.40	3054.36	667.93	MA	MA	MA	Flight	60.09	8.71	40.44	449.74	2289.04
47	MA		MA	Flight	35.49	6.30	13.13	2273.79	600.15	ME	MI, MA, MA	MA	Flight	78.20	1.15	9.33	2067.41	2325.74
48	MA		ME	Vehicle	55.17	11.98	24.39	6781.48	597.34	ME	MA	MA	Flight	50.00	3.75	14.93	776.97	2284.35
49	MA		MA	Flight	36.86	8.68	18.74	2494.47	638.60	ME	MA	MA	Flight	39.79	11.56	55.56	690.64	2312.07

Table 43: Simheuristic results for the cost-greedy ($\gamma = 1$) model.

	Tabu search									Multiple neighborhoods tabu search								
Experiment	Origin type	Transit type(s)	Destination type	Carriers	Transit time	Delivery delay	Total transit delay	Total cost	Run time	Origin type	Transit type(s)	Destination type	Carriers	Transit time	Delivery delay	Total transit delay	Total cost	Run time
1	MA	MA,MA	MA	Flight	70.51	1.37	1.95	601.88	949.40	MA		MA	Flight	32.49	0.92	1.16	499.49	2697.74
2	MA	MA	MA	Flight	48.75	3.37	11.48	470.67	841.48	MA		MA	Flight	27.81	2.00	5.82	403.07	2767.91
3	MA		ME	Flight	22.38	6.13	13.81	827.67	637.97	MA	MA	ME	Flight	76.58	1.04	2.52	685.26	2553.22
4	MA		ME	Flight	33.61	2.09	2.80	708.82	606.97	MA		ME	Flight	32.83	1.31	2.03	707.39	2518.66
5	MA		MI	Flight	58.23	4.42	4.47	337.09	683.59	MA		MI	Flight	56.70	2.85	2.86	341.05	2632.48
6	MA		ME	Flight	31.02	3.18	5.75	751.63	659.19	MA	MA	ME	Flight	124.63	1.08	14.56	690.96	2636.47
7	MA	MA	MI	Flight	109.28	4.43	45.96	458.96	848.08	MA		MI	Flight	57.35	31.97	116.08	589.40	2738.07
8	MA	MA	MI	Flight	109.28	4.43	45.14	458.66	837.95	MA		MI	Flight	58.56	33.18	120.60	610.75	2730.64
9	MA	MA	MI	Flight	109.30	4.45	45.62	460.04	839.32	MA		MI	Flight	56.86	31.47	114.14	580.46	2731.02
10	MA		ME	Flight	31.00	3.16	5.69	750.64	652.32	MA		ME	Flight	29.83	1.99	4.48	750.07	2677.66
11	MA		ME	Flight	31.06	3.22	5.88	753.91	654.39	MA	MA	ME	Flight	72.09	20.53	88.37	1001.16	2648.92
12	MA		ME	Flight	30.48	3.64	6.98	773.92	673.94	MA	MA	ME	Flight	123.59	1.05	18.87	691.94	2684.20
13	MA		MA	Flight	23.71	6.87	17.14	495.46	618.36	MA	MA	MA	Flight	150.87	0.69	0.74	457.36	2491.45
14	MA		ME	Flight	30.49	3.64	7.00	773.59	664.47	MA	MA	ME	Flight	99.53	1.00	19.08	691.02	2655.95
15	MA	MA, MA	MI	Flight	127.21	19.38	114.54	566.29	905.06	MA		MI	Flight	60.24	31.91	117.17	515.37	2616.98
16	MA		MA	Flight	23.49	6.86	17.13	482.38	624.82	MA	MA	MA	Flight	126.69	0.77	0.82	452.62	2556.10
17	MA		MA	Flight	24.25	7.03	17.35	414.81	624.41	MA	MA	MA	Flight	127.32	0.75	0.80	449.16	2483.89
18	MA		MI	Flight	74.39	2.36	2.37	689.08	694.20	MA		MI	Flight	73.59	1.55	1.56	696.25	2698.89
19	MA		MA	Flight	23.13	5.25	12.59	460.06	620.62	MA		MA	Flight	21.31	3.44	11.04	466.08	2540.50
20	MA		MA	Flight	37.71	1.21	1.25	352.58	666.10	MA		MA	Flight	37.15	0.70	0.72	350.33	2622.52
21	MA		MA	Flight	62.86	1.16	1.19	358.93	652.89	MA		MA	Flight	62.31	0.66	0.69	357.14	2620.18
22	MA		MA	Flight	24.66	11.96	30.90	723.03	666.35	MA	MA	MA	Flight	98.73	0.78	0.83	622.55	2638.94
23	MA		MA	Flight	23.88	6.93	17.30	415.38	623.05	MA		MA	Flight	21.15	4.17	14.15	478.19	2532.10
24	MA		MA	Flight	35.36	18.50	48.99	443.49	643.88	ME	MA	MA	Flight	59.15	4.82	27.89	344.72	2512.64
25	MA	MA	MA	Flight	40.34	8.09	32.49	546.72	783.83	MA		MA	Flight	19.03	4.78	16.39	360.88	2683.77
26	MI	MA	MI	Flight	70.49	2.78	6.64	502.97	821.41	MI	MA	MI	Flight	69.38	1.69	4.75	505.63	2720.97
27	MA		MA	Flight	32.55	1.12	1.19	345.58	751.91	MA		MA	Flight	32.23	0.79	0.89	346.94	2661.55
28	MA		MA	Flight	33.61	1.14	1.19	345.88	779.97	MA		MA	Flight	33.11	0.68	0.70	345.53	2695.33
29	MA	MA	MI	Flight	56.84	4.00	10.29	818.99	852.04	MA		MA	Flight	21.00	1.93	5.03	685.49	2521.32
30	MA		MA	Flight	25.15	7.17	17.52	603.49	675.75	MA		MI	Flight	80.01	1.30	1.31	510.69	2522.69
31	MA		MA	Flight	19.76	3.36	7.44	447.92	672.15	MA	MA	MA	Flight	56.16	2.11	10.12	460.70	2572.61
32	MA		MA	Flight	22.93	9.62	24.58	506.10	748.65	MA	MA	MA	Flight	99.39	0.77	0.82	491.47	2658.30
33	MA	MA	MA	Flight	49.77	3.10	9.68	445.41	935.26	MA		MA	Flight	28.76	1.75	4.44	376.45	2699.02
34	MA		MA	Flight	32.25	1.24	1.31	346.01	774.97	MA		MA	Flight	31.84	0.81	0.96	345.62	2703.54
35	MA		MA	Flight	41.79	24.77	65.78	787.60	704.76	ME	MA	MA	Flight	61.27	6.80	40.67	535.21	2622.96
36	MA	MA, MA	MI	Flight	165.10	5.52	13.03	831.24	1000.36	MA		MI	Flight	58.36	2.28	2.29	567.38	2478.69
37	MA		MA	Flight	81.70	1.14	1.16	354.37	842.21	MA		MA	Flight	81.21	0.63	0.65	352.06	2743.81
38	MA		MA	Flight	34.14	1.34	1.39	503.32	804.82	MA		MA	Flight	33.63	0.84	0.91	501.71	2652.52
39	MA		MA	Flight	35.62	18.99	50.34	437.70	727.85	ME	MA	MA	Flight	59.30	5.26	31.44	330.07	2499.27
40	MA		MA	Flight	78.97	1.23	1.25	532.25	779.72	MA		MA	Flight	78.55	0.81	0.83	532.28	2743.94
41	MA	MA, MA	MI	Flight	148.92	3.80	3.84	491.97	985.36	MA		MI	Flight	162.79	1.17	1.19	319.20	2764.37
42	MA	MA	MA	Flight	50.42	3.14	9.98	625.75	804.40	MA		MA	Flight	29.53	1.85	5.01	557.67	2656.55
43	MA		MA	Flight	30.33	1.33	1.49	348.06	787.96	MA		MA	Flight	29.72	0.72	0.89	344.96	2698.69
44	MA		MA	Flight	26.96	4.26	9.88	425.06	810.32	MA		MA	Flight	25.55	2.82	8.97	432.20	2729.54
45	MA		MA	Flight	84.66	1.24	1.26	749.82	825.42	ME	MA	MA	Flight	67.61	18.24	115.86	844.19	2664.32
46	MA		MA	Flight	26.13	10.31	26.43	397.64	749.33	MA	MA	MA	Flight	59.77	6.33	37.90	364.42	2573.40
47	MA		MA	Flight	22.87	5.23	12.68	444.39	738.90	MA		MA	Flight	20.84	3.18	10.19	438.30	2564.05
48	MA		MA	Flight	36.23	1.23	1.27	345.71	798.69	MA		MA	Flight	35.80	0.77	0.81	345.76	2605.38
49	ME		MA	Flight	28.69	14.11	36.91	566.00	793.12	ME	MA	MA	Flight	59.42	5.36	31.36	326.03	2536.43

Table 44: Simheuristic results for the balanced model.

	Tabu search								
Experiment	Origin type	Transit type(s)	Destination type	Carriers	Transit time	Delivery delay	Total transit delay	Total cost	Run time
1	MA	MA, MA	MA	Flight	37.29	5.11	29.54	450.03	1046.93
2	MA	MA	MA	Flight	26.47	4.90	27.42	379.78	916.40
3	MA		MI	Flight	31.50	4.56	12.75	697.36	701.24
4	MA		ME	Flight	17.04	2.99	7.89	817.06	653.15
5	MA		MI	Flight	25.27	8.06	26.51	328.46	783.04
6	MA		MI	Flight	78.34	1.52	1.53	957.86	736.13
7	MA	MA	MI	Flight	46.47	18.09	101.18	521.71	890.07
8	MA	MA	MI	Flight	46.31	17.94	100.02	517.63	881.14
9	MA	MA	MI	Flight	46.40	18.02	100.77	519.97	954.02
10	MA		MI	Flight	102.36	1.53	1.54	958.21	753.35
11	MA		MI	Flight	102.35	1.52	1.53	957.70	779.99
12	MA		MI	Flight	77.35	1.53	1.54	959.79	769.50
13	MA		MA	Flight	21.31	4.48	14.64	493.94	759.44
14	MA		MI	Flight	53.33	1.51	1.52	956.58	814.22
15	MA	MA, MA	MI	Flight	82.88	11.45	159.79	507.54	955.75
16	MA		MA	Flight	21.15	4.53	14.77	482.23	616.36
17	MA		MA	Flight	21.85	4.64	14.97	321.13	619.66
18	MA		MA	Flight	36.16	1.03	1.08	719.99	682.47
19	MA		MA	Flight	21.27	3.40	10.58	457.69	614.46
20	MA		MA	Flight	37.32	0.80	0.83	352.04	661.04
21	MA		MA	Flight	62.41	0.73	0.75	358.08	646.01
22	MA		MA	Flight	20.69	7.98	27.11	642.36	671.28
23	MA		MA	Flight	21.50	4.54	14.79	482.46	628.59
24	MA		MA	Flight	29.19	12.33	43.08	379.85	703.44
25	MA	MA	MA	Flight	19.83	8.40	49.32	555.09	904.86
26	MI	MA	MI	Flight	68.82	1.12	1.95	497.68	995.94
27	MA		MA	Flight	32.16	0.73	0.78	345.24	833.14
28	MA		MA	Flight	33.24	0.76	0.80	345.62	740.04
29	ME	MA	MA	Flight	40.24	3.14	15.33	1161.76	852.70
30	MA		MA	Flight	22.63	4.66	14.84	464.25	718.44
31	MA		MA	Flight	18.62	2.21	6.30	447.77	648.13
32	MA		MA	Flight	26.22	2.35	8.33	485.88	707.01
33	MA	MA	MA	Flight	27.11	4.32	23.27	538.25	849.56
34	MA		MA	Flight	31.83	0.82	0.90	346.14	677.82
35	MA		MA	Flight	33.33	16.31	57.21	650.35	663.84
36	ME	MA, MA	MI	Flight	55.76	3.59	13.74	866.69	869.26
37	MA		MA	Flight	81.32	0.75	0.77	354.49	718.62
38	MA		MA	Flight	33.66	0.86	0.94	502.91	691.35
39	MA		MA	Flight	28.94	12.31	43.09	368.07	636.75
40	MA		MA	Flight	78.57	0.84	0.86	533.07	737.32
41	MA	MA, MA	MI	Flight	84.95	1.30	4.40	432.62	1010.31
42	MA	MA	MA	Flight	27.71	4.31	23.04	752.12	799.46
43	MA		MA	Flight	29.87	0.85	1.01	347.30	678.14
44	MA		MA	Flight	31.16	2.12	7.61	433.55	709.18
45	MI		MA	Flight	24.90	1.16	2.12	791.91	695.19
46	MA		MA	Flight	22.43	6.62	22.28	356.94	623.70
47	MA		MA	Flight	21.10	3.46	10.90	444.75	616.92
48	MA		MA	Flight	35.82	0.81	0.84	345.27	661.18
49	ME		MA	Flight	23.75	9.17	31.59	490.68	643.41

	Multiple neighborhoods tabu search								
Experiment	Origin type	Transit type(s)	Destination type	Carriers	Transit time	Delivery delay	Total transit delay	Total cost	Run time
1	MA		MA	Flight	32.14	0.56	0.67	495.61	2765.08
2	MA		MA	Flight	22.13	3.15	12.54	327.86	2752.11
3	MA	MA, MA	ME	Flight	77.20	1.65	37.73	792.62	2516.57
4	MA		ME	Flight	15.83	1.78	5.25	806.86	2606.59
5	MA		MI	Flight	22.23	5.01	19.97	244.85	2745.81
6	MA	MA	ME	Flight	41.25	13.66	69.90	855.83	2638.19
7	MA		MI	Flight	45.39	20.02	84.60	459.29	2782.28
8	MA		MI	Flight	45.52	20.13	85.54	458.72	2777.82
9	MA		MI	Flight	45.69	20.31	86.05	464.84	2780.09
10	MA	MA	ME	Flight	52.27	0.72	11.73	692.69	2622.81
11	MA	MA	ME	Flight	52.23	0.69	12.17	690.13	2612.75
12	MA	MA, MA	ME	Flight	63.89	0.65	30.17	830.73	2650.88
13	ME	MA	MA	Flight	32.47	2.12	14.15	587.42	2595.42
14	MA	MA	ME	Flight	99.22	0.69	15.63	691.64	2704.72
15	ME	MA	MI	Flight	40.13	8.88	61.04	386.93	2563.82
16	ME	MA	MA	Flight	21.33	4.47	33.10	402.84	2593.49
17	ME	MA	MA	Flight	22.05	4.61	34.22	403.71	2596.25
18	MA	MA	MI	Flight	45.38	1.24	3.48	736.35	2734.63
19	MA		MA	Flight	20.02	2.13	7.98	460.76	2510.65
20	MA		MA	Flight	36.89	0.47	0.41	350.80	2662.69
21	MA		MA	Flight	62.16	0.48	0.42	360.37	2616.14
22	MA		MI	Flight	51.97	0.93	0.79	762.67	2703.81
23	ME	MA	MA	Flight	21.44	4.25	31.30	405.30	2525.38
24	MA		MA	Flight	24.66	7.80	33.29	342.45	2579.24
25	MA		MA	Flight	17.46	3.20	12.53	359.38	2658.34
26	MI	MA	MI	Flight	68.42	0.71	1.35	500.45	2729.96
27	MA		MA	Flight	31.95	0.46	0.49	345.58	2690.34
28	MA		MA	Flight	32.99	0.48	0.42	347.05	2682.81
29	MA		MA	Flight	20.24	1.20	3.38	680.56	2513.37
30	MA		MI	Flight	79.59	0.89	0.76	514.45	2559.77
31	MA	MA	MA	Flight	16.93	2.59	18.12	547.12	2536.73
32	MA		MA	Flight	17.33	4.00	16.20	360.53	2637.48
33	MA		MA	Flight	22.90	2.71	10.46	466.61	2655.59
34	MA		MA	Flight	31.52	0.52	0.50	344.57	2689.70
35	ME	MA	MI	Flight	42.33	9.60	66.47	609.64	2530.14
36	MA		MI	Flight	57.58	1.51	1.27	572.09	2488.51
37	MA		MA	Flight	81.06	0.48	0.42	354.37	2707.11
38	MA		MA	Flight	33.35	0.54	0.57	503.95	2640.12
39	MA		MA	Flight	23.95	7.31	31.01	320.32	2539.84
40	MA		MA	Flight	78.26	0.49	0.43	532.00	2726.07
41	MA	MA	MI	Flight	86.69	0.97	2.81	339.57	2818.41
42	MA		MA	Flight	23.54	2.72	10.42	682.74	2608.32
43	MA		MA	Flight	29.55	0.54	0.63	348.64	2623.76
44	MA		MA	Flight	29.19	1.52	6.56	435.94	2648.32
45	ME	MA	MA	Flight	38.17	12.78	99.71	772.55	2668.89
46	MA	MA	MA	Flight	34.99	5.38	40.38	301.49	2633.78
47	MA		MA	Flight	19.65	2.01	7.37	436.12	2455.27
48	MA		MA	Flight	35.53	0.52	0.47	348.23	2598.11
49	ME	MA	MA	Flight	23.78	6.92	53.53	472.89	2542.41

Table 45: Simheuristic results for the weighted model.

	Tabu search									Multiple neighborhoods tabu search								
Experiment	Origin type	Transit type(s)	Destination type	Carriers	Transit time	Delivery delay	Total transit delay	Total cost	Run time	Origin type	Transit type(s)	Destination type	Carriers	Transit time	Delivery delay	Total transit delay	Total cost	Run time
1	MA	MA, MA	MA	Flight	32.44	1.09	8.35	654.82	860.81	MA		MA	Flight	31.98	0.45	0.55	493.80	2370.42
2	MI	MA	MA	Flight	27.24	0.80	3.72	705.85	771.31	MA		MA	Flight	21.69	2.70	14.58	329.21	2441.76
3	MA		MI	Flight	39.43	3.00	11.73	1054.49	601.24	MA		MI	Flight	29.24	2.30	10.18	1036.40	2339.23
4	MA		ME	Flight	21.82	1.86	6.70	815.26	574.27	MA		ME	Flight	15.57	1.52	6.10	807.83	2277.20
5	MA		MI	Flight	38.51	5.11	23.66	463.56	661.12	MA		MI	Flight	21.62	4.37	23.34	244.30	2325.01
6	ME		ME	Flight	30.66	0.73	0.79	1119.25	620.49	MA	MA	ME	Flight	19.61	3.33	31.02	677.45	2334.67
7	MA	MA	MA	Flight	59.32	10.75	96.67	1097.00	751.23	MA		MI	Flight	31.73	3.38	9.37	672.26	2431.25
8	MA	MA	MA	Flight	58.85	10.64	95.60	1095.31	768.67	MA		MI	Flight	42.05	16.68	95.17	418.55	2370.63
9	MA	MA	MA	Flight	70.87	10.41	93.40	839.15	756.01	MA		MI	Flight	42.47	17.09	97.89	425.65	2395.56
10	ME		ME	Flight	30.67	0.73	0.79	1118.53	620.81	MA	MA	ME	Flight	19.55	3.28	30.30	675.38	2380.62
11	ME		ME	Flight	30.64	0.72	0.80	1118.30	622.75	MA	MA	ME	Flight	19.67	3.39	31.37	674.90	2381.15
12	ME		ME	Flight	29.67	0.73	0.84	1120.76	634.14	MA	MA	ME	Flight	19.48	4.19	40.13	679.49	2429.06
13	MA		MA	Flight	28.74	2.86	13.11	494.91	585.80	ME	MA	MA	Flight	31.99	1.69	14.97	575.44	2303.78
14	ME		ME	Flight	29.67	0.73	0.81	1118.76	639.45	MA	MA	ME	Flight	19.40	4.11	39.67	681.73	2382.17
15	MA	ME, MA	MI	Flight	94.34	17.01	177.48	407.82	830.29	ME	MA	MI	Flight	38.34	7.09	65.58	353.13	2329.16
16	MA		MA	Flight	28.38	2.83	12.98	480.39	578.95	ME	MA	MA	Flight	20.61	3.75	38.53	404.16	2259.21
17	MA		MA	Flight	29.45	2.94	13.24	484.21	583.18	ME	MA	MA	Flight	21.25	3.81	38.44	407.62	2262.27
18	MA		MA	Flight	37.84	0.65	0.70	720.20	654.53	MA		MA	Flight	35.66	0.52	0.54	716.29	2446.42
19	MA		MA	Flight	27.09	2.21	9.68	462.15	572.25	MA		MA	Flight	19.66	1.80	9.07	460.90	2235.61
20	MA		MA	Flight	38.60	0.51	0.56	353.51	630.66	MA	MA	MA	Flight	33.97	0.63	2.77	371.70	2352.78
21	MA		MA	Flight	63.62	0.47	0.49	357.81	610.29	MA		MA	Flight	62.11	0.42	0.44	359.38	2334.34
22	MA		MA	Flight	33.53	5.00	24.09	903.03	637.86	MA		MI	Flight	51.86	0.82	0.83	767.28	2459.38
23	MA		MA	Flight	28.90	2.87	13.11	482.40	589.81	MA		MA	Flight	19.26	2.33	12.27	479.21	2353.46
24	MA		MA	Flight	49.06	7.73	38.18	500.81	606.03	ME	MA, MA	MA	Flight	33.82	3.49	50.26	417.89	2316.20
25	MA	MA	MA	Flight	33.45	5.29	46.16	789.17	713.34	MA		MA	Flight	17.18	2.92	15.54	359.66	2307.01
26	MI	MA	MI	Flight	70.69	0.72	1.56	498.03	777.05	MA		MI	Flight	29.63	2.97	8.59	507.50	2438.75
27	MA		MA	Flight	33.34	0.46	0.52	345.18	642.63	MA		MA	Flight	31.85	0.42	0.49	345.63	2445.27
28	MA		MA	Flight	34.48	0.48	0.54	346.12	635.11	MA		MA	Flight	32.86	0.39	0.49	346.61	2449.49
29	ME	MA	MA	Flight	45.50	2.01	14.46	1166.43	721.62	MA		MA	Flight	20.17	1.09	4.28	686.85	2241.60
30	MA		MA	Flight	30.20	2.94	13.05	700.98	592.97	ME	MA	MA	Flight	21.93	3.77	37.59	544.95	2269.68
31	MA		MA	Flight	22.34	1.42	5.61	449.31	570.41	MA	MA	MA	Flight	16.53	2.17	21.11	543.16	2255.02
32	MA		MA	Flight	30.13	4.04	19.04	613.91	633.22	MA		MA	Flight	16.61	3.30	17.97	360.19	2407.30
33	MA	MA	MA	Flight	34.19	2.74	21.66	537.52	778.75	MA		MA	Flight	22.66	2.44	12.69	314.38	2374.51
34	MA		MA	Flight	33.22	0.53	0.60	346.45	637.71	MA	MA	MA	Flight	28.06	0.73	4.03	379.37	2383.73
35	MA		MA	Flight	60.33	10.39	51.66	777.65	623.84	MI	ME, MA	MI	Flight	33.95	1.26	5.39	986.60	2256.66
36	MA	ME, MA	MI	Flight	50.11	5.54	50.75	1977.46	817.65	ME	MA	MI	Flight	44.51	4.93	23.60	469.12	2246.53
37	MI		MA	Flight	24.63	0.90	2.74	671.54	632.20	MI	MA	MA	Flight	25.27	0.50	1.69	581.68	2370.25
38	MA		MA	Flight	35.14	0.56	0.63	504.01	657.60	MA		MA	Flight	33.23	0.47	0.51	503.87	2424.51
39	MA		MA	Flight	48.56	7.66	37.92	484.13	599.43	ME	MA, MA	MA	Flight	33.32	3.20	46.88	416.30	2267.57
40	MI		MA	Flight	26.64	2.05	8.51	933.61	654.69	MI	MA	MA	Flight	22.81	0.89	5.53	816.12	2405.59
41	MA	MA, MA	MI	Flight	87.08	0.84	4.12	433.76	915.02	MA	MA	MI	Flight	86.45	0.73	2.35	334.03	2500.96
42	MI	MA	MA	Flight	29.02	0.76	2.99	848.34	777.67	MA		MA	Flight	23.32	2.48	12.90	460.72	2424.60
43	MA		MA	Flight	31.25	0.54	0.71	347.45	643.99	MA	MA	MA	Flight	26.26	0.90	5.64	392.15	2408.93
44	MA		MA	Flight	34.54	1.29	6.61	436.82	674.61	MA		MA	Flight	30.05	1.24	7.41	428.31	2469.41
45	MI		MA	Flight	26.90	0.76	1.78	794.26	644.65	ME	MA	MA	Flight	36.17	10.80	114.99	722.27	2368.37
46	MA		MA	Flight	33.59	4.26	20.24	484.84	615.05	MA	MA	MA	Flight	17.52	4.56	46.60	324.23	2313.96
47	MA		MA	Flight	26.75	2.19	9.62	444.16	583.97	MA		MA	Flight	19.50	1.84	9.52	444.82	2215.34
48	MA		MA	Flight	37.11	0.51	0.53	345.16	630.45	MA		MA	Flight	35.39	0.40	0.43	344.11	2390.50
49	ME		MA	Flight	39.15	5.89	28.76	711.73	602.41	MI	MA	MA	Flight	32.96	0.43	0.61	555.73	2362.76