Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Economics and Management
Dept. of Industrial Economics and Technology Management

Arne Thien An Bui

Simulation of a Dial-a-Ride Service with Autonomous Ferries in the Kiel Fjord

Master's thesis in Industrial Economics and Technology Management
Supervisor: Prof. Kjetil Fagerholt
August 2020

NTNU
Norwegian University of Science and Technology

Preface

This master's thesis concludes my Master of Science in Industrial Economics and Technology Management at the Norwegian University of Science and Technology, Department of Industrial Economics and Technology Management. The work is done within the field of Managerial Economics and Operations Research.

The thesis is a continuation of the project report completed together with fellow student Jennifer Nguyen in the fall of 2019. The work is written as part of a larger project related to the ongoing initiative, *CAPTin Kiel*, and is a collaborative effort with the Christian-Albrecht University of Kiel.

I want to express my sincere gratitude to my supervisors Prof. Dr. Kjetil Fagerholt, Prof. Dr. Frank Meisel, Dr. Lennart Johnsen, and Dr. Mohamed Kais Msakni, for valuable discussions, guidance, and constructive feedback. I would also like to extend my appreciation to family and friends for their endless support and encouragement in completing this master's thesis.

Trondheim, August 2020

Arne T. A. Bui

Abstract

Recent technological developments have enabled us to explore new modes of operations. It is believed that the changed cost structures due to the utilization of a fully autonomous fleet, allows for the design of flexible, cost-efficient, and climate-friendly mobility services. Through the project CAPTin Kiel, this thesis aims to study the opportunity to provide an on-demand ferry service with autonomous ferries in the Kiel Fjord. The intended fleet will utilize smaller autonomous ferries for replacing conventional large ferries in order to achieve greater flexibility.

In this thesis, the Dynamic Dial-a-Ride Problem with Autonomous Ferries (DDARP-AF) is studied through simulation. The problem concerns the design of an on-demand ferry service, where incoming requests with potentially very short call ahead times are made known to the service provider after the initial routing and scheduling of ferries. The highly dynamic nature entails that the operational planning procedure needs to efficiently determine if the requests can feasibly be served in an online manner and update the ferry schedules accordingly. It is essential that the new ferry service must be designed in a way that can meet the expected demand while being able to maintain an adequate level of service perceived by the passengers. The main challenge is being able to balance the desired service level against the cost of operating the service.

A simulation model is developed to evaluate the effects of changing various characteristics of the service. An insertion heuristic is chosen to solve the operational planning, as it is essential that the operational planning problem can be solved efficiently. Feasible insertions are determined by pickup time window constraints, maximum ride time

constraints, and ferry capacity constraints. The performance of different services can be measured through key performance indicators defined from the simulation output, reflecting both the perspective of both the operator and passengers. The efficiency of various configurations can further provide insight and decision support in recommending the fleet configuration and overall design of the ferry service.

The simulation model is implemented in Python with the process-oriented discrete-event simulation framework Simpy. Test instances were characterized by parameter values for pickup time window widths, maximum ride time coefficient, fleet configurations, and the choice in the objective function. The test instances were simulated for three different demand scenarios over a planning horizon of 500 hours to assure long-term steady-state performance.

The results indicate that the insertion heuristic is not able to exploit increased planning flexibility through wider time window widths efficiently. The average excess ride time is significantly affected by the value of the maximum ride time coefficient, regardless of other service-related parameters. Generally, the combinations of service-related parameters performed similarly for different fleets and demand scenarios. A wider time window width combined with a high maximum ride time coefficient provided the overall best performance with regards to demand met at the expense of a higher, but still acceptable level of average excess ride time. The change of objective function in favor of minimizing average distance traveled per ferry provided a positive impact of up to 39 % in the service's ability to accommodate requests. The result implies that that excessive consideration towards minimizing excessive ride times limits the overall performance.

A sensitivity analysis of the fleet size suggested an almost linearly proportional relation between increased fleet size and demand met at the peak demand scenario for a given setting. Varying fleet size for a given setting does not seem to affect other key performance indicators significantly. Given the preference of the service provider, a fleet of at least 13 ferries is found to provide a sufficiently acceptable level of service in the peak demand scenario.

Sammendrag

Nylige teknologiske framskritt har muliggjort nye operasjonsmoduser. Man ser for seg at en endret kostnadsstruktur gjennom bruk av en helautonom flåte, gjør det mulig for å designe fleksible, kostnadseffektive, og klimavennlige transporttjenester. Gjennom prosjektet CAPTin Kiel, sikter denne masteroppgaven på å utforske muligheten til å tilby et on-demand fergetilbud med autonome ferger i Kielfjorden. Den tiltenkte flåten skal benytte mindre autonome ferger for å erstatte konvensjonelle større ferger, og dermed oppnå større fleksibilitet.

I denne masteroppgaven vil det Dynamiske Dial-a-Ride Problemet med Autononome Ferger (DDARP-AF) studeres gjennom simuleringer. Problemstillingen tar for seg designet av et on-demand fergetilbud, hvor innkommende forespørsler med potensielt meget kort innringingstid blir gjort kjent for tjenesteleverandøren etter at den initielle rute- og tidsplanleggingen av fergene har skjedd. Det karakteristiske dynamiske trekket medfører at den operasjonelle planleggingsprosedyren må kunne effektivt avgjøre om forespørsler kan imøtekommes fortløpende, og oppdatere ruteplanleggingen deretter. Det er særlig viktig at det nye fergetilbudet blir designet på en måte som gjør at det dekker forventet etterspørsel, og samtidig opprettholder et tilstrekkelig oppfattet servicenivå for passasjerene. Hovedutfordringen er å balansere ønsket servicenivå mot kostnaden til tjenesten sett fra operatørens perspektiv.

En simuleringsmodell er utviklet for å evaluere effektene av å endre forskjellige karakteristike innstillinger for fergetilbudet. En innsetningsheuristikk er valgt for å løse operasjonsplanleggingen, ettersom det er kritisk at operasjonsplanleggingsproblemet kan løses

effektivt. Mulige innsetninger er begrenset av betjeningsvindurestriksjoner, maksimal reisetidsrestriksjoner, og fergekapasitetsrestriksjoner. Ytelsen til de ulike fergetilbudene kan måles med nøkkelindikatorer gitt av simuleringsresultat. Disse gjenspeiler både operatørens og passasjerenes perspektiv. Effektiviteten til forskjellige konfigurasjoner kan videre gi innsikt og beslutningsstøtte for anbefalt flåtekonfigurasjon og overordnet design av fergetilbudet.

Simuleringsmodellen er implementert i Python med det prosessorienterte diskret hendelsessimuleringsrammeverket Simpy. Testinstansene er karakterisert av forskjellige parameterverdier for størrelsen på betjeningsvinduet, maksimal reisetidskoeffisient, flåtesammensetning, og valg av objektivfunksjon. Testinstansene var simulert for tre forskjellige etterspørselsscenarioer med en tilsvarende planleggingshorisont på 500 timer for å forsikre en stabil ytelse og rapportert resultat.

Resultatene indikerer at innsetningsheuristikken ikke klarer å effektivt utnytte økt planleggingsfleksibilitet gjennom større betjeningsvinduer. Den gjennomsnittlige overflødige varigheten av en reise er særlig påvirket av verdien for maksimal reisetidskoeffisient, uavhengig av andre servicerelaterte parametere. Generelt var det lik ytelse for de forskjellige kombinasjonene av parameterverdier med ulike flåter og etterspørselsscenarioer. Større betjeningsvindu kombinert med en høy maksimal reisetidskoeffisient ga den generelt beste ytelsen med hensyn på møtt etterspørsel til tross for en høyere, men fortsatt akseptabelt nivå på gjennomsnittlig overflødig reisetid. Endringen i objektivfunksjonen i favør av å minimere gjennomsnittlig distanse seilet per ferge ga en positiv innvirkning på opp til 39 % på tjenestens evne til å imøtekomme forespørsler. Resultatet indikerer at overveldende betraktning til å minimere overflødig reisetid begrenser den generelle ytelsen til hele fergetilbudet.

En sensitivitetsanalyse av flåtestørrelsen antyder en nesten lineær proporsjonal relasjon mellom økt flåtestørrelse og møtt etterspørsel ved det høyeste etterspørselsscenarioet for en gitt konfigurasjon. Variasjon av flåtestørrelsen for en gitt konfigurasjon påvirker tilsynelatende ikke andre nøkkelindikatorer signifikant. Gitt preferansen til tilbyder av tjenesten, ble det funnet at en flåtestørrelse på minst 13 ferger kan tilby et tilstrekkelig akseptabelt fergetilbud ved det høyeste etterspørselsscenarioet.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The majority of the world's population growth is anticipated to occur in urban areas for the next 30 years. Urban areas already account for 60-80 % of the global energy consumption and greenhouse gas emissions, and a proportional increase in traffic would lead to significant increments in pollution and traffic congestion (European Commission, 2017). Recent technological developments in the fields of, e.g., artificial intelligence, machine learning, 5G, and cloud computing have enabled us to explore new modes of operations. With this in mind, it is, therefore, of high interest to study efficient and sustainable mobility systems that can cope with future growth.

It is believed that the changed cost structures due to the utilization of a fully autonomous fleet, allows for the design of high service level offerings that could change the travel pattern of day-to-day commuters (Kretschmann, Burmeister & Jahn, 2017). This is due to the fact that autonomous systems eliminate the need for careful considerations of the staff roster, fixed work-hours, and related facilities – potentially providing great flexibility in developing sustainable and energy-efficient systems. This thesis presents and discusses the possibility of introducing a demand-responsive service with autonomous ferries in the Kiel Fjord.

1.1 The Kiel Fjord

Kiel is a seaport city in the northern parts of Germany, with a population of about 250 000 people. The city is split into a western and eastern shoreline by the Kiel Fjord. After World War II, the city infrastructure was rebuilt to mainly facilitate cars and buses, while the fjord was primarily intended for transportation of industrial commodities. Recently, as issues of air pollution and environmental concerns have increased, new political incentives are focused on sustainable and efficient means for transportation.

Every day, several cruise ships, local ferries, and industrial cargo ships constitute the fjord traffic. The northern parts of the fjord consist of beaches and are popular destinations for recreational purposes during summer. In contrast, the inner parts located near the city center are crowded during commuting times. There are also several military zones and facilities along the fjord, and the Kiel Canal, which is the busiest artificial waterway in the world, is directly linked with the fjord on the western shore. This diversity in traffic and demand should be considered when conducting studies related to transportation and logistics in the Kiel Fjord.

Today, there is an existing ferry service provided by Schlepp- und Fährgesellschaft Kiel (SFK). SFK operates a regular fixed-route service with the deployment of non-autonomous ferries. The ferries are rather large, with vessel capacities of 300 passengers (SFK, 2020). The maximum capacity is rarely exploited, with the exception being during the annual sailing festival, Kieler Woche. As can be seen in Figure 1.1, there are a total of ten regular ports which is served by two routes. The Förde-Fährlinie is the main route and covers the ports from Bahnhof up to Laboe. During the summer, the route is extended to include the ports of Falckenstein, Schilksee, and Strande. The Schwentine-Fährlinie serves as a more direct line across the fjord for commuters, linking the eastern shore to the central areas of the city.

Despite having a seemingly functional ferry service, several factors make the current offering less suitable as a means for day-to-day transportation across the fjord. Firstly, as can be noted in Figure 1.1, most ports do not have a direct connection. This results in very long transit times, with the trip from Bahnhof to Laboe potentially taking up to

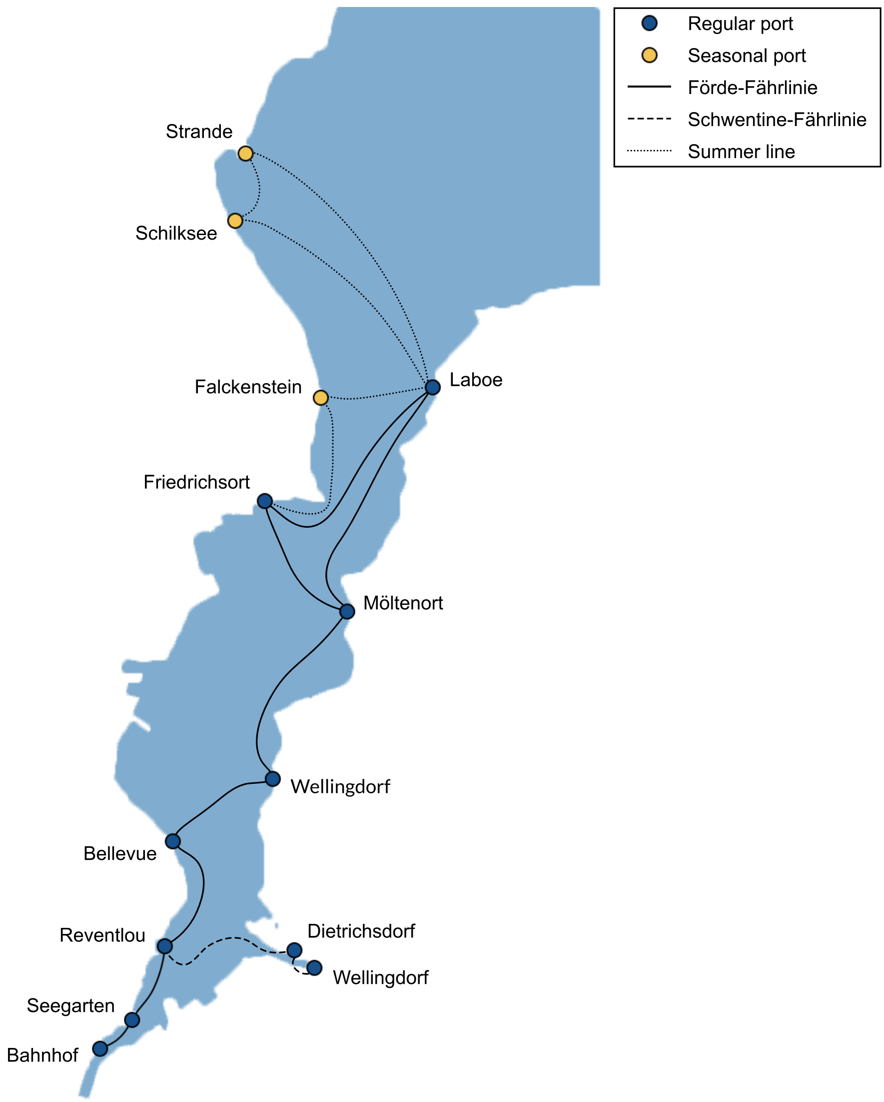


Figure 1.1: The current ferry service in the Kiel Fjord, operated by SFK.

almost an hour. Further, the schedules do not include all the possible ports in the route consistently throughout the day, resulting in varying departure times. The frequency of departures is relatively low, with at most one ferry departure every hour for all ports. The need for careful planning and adaption by the passenger in advance makes travel by ferry not a viable option for most commuters.

1.2 Clean Autonomous Public Transport in Kiel

Clean Autonomous Public Transport in Kiel (CAPTin Kiel) is a joint project initiated by the Christian-Albrecht University of Kiel (CAU) in 2017. The project constitutes a transdisciplinary innovation platform with several academic institutions, private firms, as well as local and federal governments involved. Through the project, the City of Kiel seeks to establish an innovative urban transport infrastructure, utilizing autonomous vessels to provide cost-efficient, flexible and climate-friendly mobility services in a user-friendly way (Pankratz & Müller-Lupp, 2020). The platform consists of different project groups with an emphasis on varied aspects concerning implementing and designing the new mobility system. Some are concerned with the technical implementation and testing of autonomous technology, while another group from the Muthesius University of Fine Arts and Design has been working on conceptual designs for the new autonomous ferries, as seen in Figure 1.2.

Sørensen (2017) defines a hierarchy to distinguish different levels of autonomy, but the concept of autonomy can be summarized as machines operating processes automatically without human influence (Cross & Meadow, 2017). A fully autonomous fleet without the need of crew members on-board could enable the design of more cost-efficient demand-responsive services, as staff costs can make such services up to three times more expensive than similar fixed-route services (Anderson et al., 2014). Even though a positive environmental effect cannot be guaranteed through autonomous mobility services without further insight into future technical and regulatory development, it is reasonable to believe that these aspects will be made clear with the growing adoption of the technology (Pitera & Marinelli, 2017). Several successful demonstrations and ongoing trials include the world's first autonomous ferry with passengers by Rolls-Royce and Finferries in 2018, and Yara

Birkeland as the world's first commercial autonomous container ship (Kongsberg, 2019; Rolls-Royce, 2018).

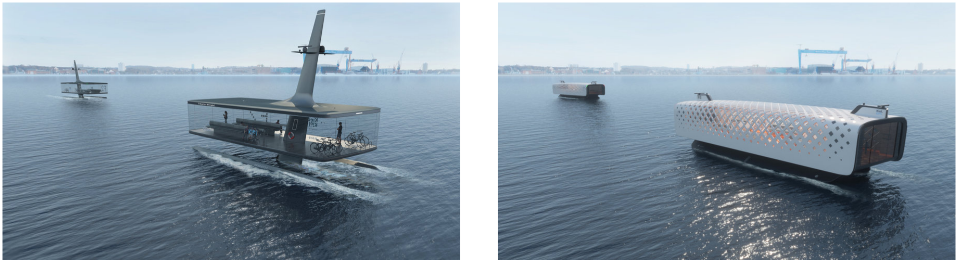


Figure 1.2: Conceptual designs of autonomous ferries (Pankratz & Müller-Lupp, 2020).

The work presented in this thesis is part of CAPTin Kiel as an R&D activity in cooperation with CAU to look into optimization and simulation of an autonomous ferry service in the Kiel Fjord. The central idea is to replace the conventional large ferries with smaller autonomous ferries in order to achieve greater flexibility. This thesis studies a demand-responsive shared-ride service with autonomous ferries. More specifically, the problem is modeled as a dynamic dial-a-ride service, where demand is unknown beforehand, and incoming customer requests need to be processed in an online manner. The specific problem of this thesis is referred to as the Dynamic Dial-a-Ride Problem with Autonomous Ferries (DDARP-AF). As previously mentioned, the current ferry service is not able to provide commuters with a sufficient and reliable means of day-to-day transportation. Thus, the new on-demand ferry service must be designed in a way that can meet the expected demand while being able to maintain an adequate level of service perceived by the passengers. As such, the typical characteristic of dial-a-ride problems (DARP) is to balance the desired service level against the cost of operating the service. Hence, the designs of the ferry service must be evaluated in terms of this. The output of a simulation model can be used as metrics to provide insight into how different system designs and ways of operation affect the overall performance and efficiency of the service under various conditions.

1.3 Contribution and Purpose

The motivation for this thesis is to study the DDARP-AF to provide decision support in designing a demand-responsive service with autonomous ferries in the Kiel Fjord. The problem is examined through an operations research point of view; thus, the legal, safety, and technical aspects regarding the use of autonomous ferries are not studied.

The service provider faces a trade-off between flexibility and costs in deciding the fleet composition. Dial-a-ride services are usually quite costly, as a higher number of vessels make the service more capable of serving requests with very short call ahead times. Overall, a larger fleet provides great flexibility in designing services that can maintain a high level of service for the passengers. However, utilizing a smaller fleet with higher capacities could provide economies of scale in terms of lower cost per passenger.

The purpose of this thesis is to provide managerial insight in determining an optimal fleet composition through a simulation study. Hence, the main contributions of the thesis are thus:

- A literature survey of relevant solution methods to evaluate the DDARP-AF.
- A mathematical model to solve the DARP-AF over a static planning horizon.
- A simulation model to evaluate the DDARP-AF through real-world performance metrics.
- A methodology to evaluate different ferry service designs, i.e., policies and fleet mix, under different demand scenarios.

1.4 Structure of Thesis

The outline of the thesis is as follows. First, in Chapter 2, a literature review concerning demand-responsive services, evaluation of dial-a-ride systems, as well as appropriate heuristic methods is conducted. Next, the problem description of the DDARP-AF is presented in Chapter 3, and the mathematical formulation of the DARP-AF is presented in Chapter 4. Chapter 5 describes the simulation model to evaluate the real-world performance of a ferry service, while Chapter 6 considers the procedure of handling incoming requests. In Chapter 7, the generation of data used in this study is described, and a computational study is conducted and presented. Lastly, concluding remarks with a discussion of further research are given in Chapter 8.

Chapter 2

Related Literature

There has been a renewed interest in dial-a-ride systems driven by environmental concerns and recent technological advances that enables new modes of operations. The various decisions related to the DDARP-AF can be categorized into different planning levels. As can be seen in Table 2.1, relevant decisions at the strategic level include the location and characteristics of ports. At the tactical level, decisions involving optimal fleet configuration and service policy are made. The operational level is concerned with the planning and scheduling of the service. This study mainly focuses on the decisions made within the tactical and operational levels of the DDARP-AF.

The chapter presents and discusses relevant literature for analyzing the DDARP-AF. First, an overview of the characteristics and applications of dial-a-ride problems is listed in Section 2.1. Section 2.2 elaborates on the literature describing solution methods for the operational planning problem. Section 2.3 presents how simulation studies have been conducted to study the effects of different changes in dial-a-ride services. Lastly, a summary of the literature survey is provided in Section 2.4. In practice, since there exist many different dial-a-ride services depending on the intended application, the literature review mainly highlights the relevant features to study the DDARP-AF further. For a more general and extensive literature survey of dial-a-ride problems, the work of Cordeau and Laporte (2007) and Ho et al. (2018) is recommended.

Table 2.1: Categorization of planning levels for the DARP with ferries.

Strategic	Number of ports
	Location of ports
	Size of ports
Tactical	Fleet size
	Ferry types
	Booking system
Operational	Ferry routing
	Scheduling
	Maintenance / Recharging

2.1 Dial-a-Ride Services

Dial-a-ride problems (DARP) are typically classified in terms of the mode in which the service is operated by. The static mode defines the case where all the requests are known, allowing for the planning of vehicle schedules before the start of the given planning horizon. In contrast, the problem is considered dynamic if the planning starts before all requests are known, and the operator can update the schedules for a fleet as the number of known requests are increasing. Ho et al. (2018) further extend the taxonomy to include the certainty of information at the time of planning, i.e., deterministic or stochastic. This taxonomy differs slightly from that of Pillac, Gendreau, Guéret and Medaglia (2013) for vehicle routing problems, by considering the possibility of imperfect information. The four categories of DARP is displayed in Table 2.2. For practical reasons, the deterministic mode is assumed when referring to the static or dynamic DARP, unless otherwise stated.

Real-world applications are always the basis for modeling DARPs, and thus varied problems are modeled explicitly to reflect the realistic features. It is typical to distinguish the literature in terms of main characteristics, such as whether the model considers a single-vehicle problem or multi-vehicle problem, and also if the fleet is homogeneous or heterogeneous. Common early studies of the DARP considers the application in paratransit of elderly or disabled people. The static mode of these services is, for instance, described by Toth and Vigo (1996), which consider a heterogeneous fleet with multiple depots and non-depot-based vehicles, while Madsen, Ravn and Rygaard (1995), Cordeau and Laporte (2007), Pillac et al. (2013) and Häll, Lundgren and Voß (2015) consider the dynamic case. Another important application includes ambulance services described by Hanne, Melo and Nickel (2009), Gendreau, Laporte and Semet (2001), and Beaudry, Laporte, Melo and Nickel (2010). The most recent studies have focused on the use of demand-responsive services in public transport. Both van Engelen, Cats, Post and Aardald (2018) and Hyland and Mahmassani (2018) suggest new methods and strategies for rerouting vehicles under a stochastic environment that can potentially reduce vehicle mileage and increase the service level, ideally aimed towards the utilization of autonomous vehicles.

Table 2.2: Taxonomy of DARP.

	Deterministic	**Stochastic**
Static	Decisions are made a priori, and the information is known with certainty.	Decisions are made a priori, but the certainty of the information is undetermined at the time of decision.
Dynamic	Decisions are made in response to new information received, and the information is known with certainty.	Decisions are made in response to new information received, but the certainty of the information is undetermined at the time of decision.

2.2 Solution Methods

Exact solution methods, primarily based upon branch-and-bound approaches, have been developed for the static DARPs. Such methods were first introduced by Cordeau (2006), which added cutting planes to a three-index mixed-integer programming formulation in a branch-and-cut algorithm. Ropke, Cordeau and Laporte (2007) provide a further tightened two-index formulation with new classes of valid inequalities. Although exact methods provide solutions of the highest quality, finding optimal solutions for extended models with large instances is very difficult due to the problem being NP-hard. For this reason, most solutions approaches are inclined towards heuristic methods.

The most common metaheuristic approach for solving DARPs is tabu search. Recent studies involving tabu search, such as Beaudry et al. (2010), are typically inspired by Cordeau and Laporte (2003), which combines simple neighborhood operators with the penalization of frequently made moves and the possibility to accept temporarily infeasible solutions. Tabu search is often used to study extensive models that incorporate complicated real-life constraints. Other metaheuristics include genetic algorithms (e.g., Uchimura, Takahashi and Saitoh, 2002), simulated annealing (e.g., Braekers, Caris and Janssens, 2014), and variable neighborhood search (e.g., Parragh, Doerner, Hartl and Gandibleux, 2009).

However, due to the nature of some dynamic systems requiring feasible solutions to be found in a very timely manner, variations of simple insertion heuristics are most often used to study these in diverse contexts. The general greedy procedure of the ADARTW heuristic developed by Jaw, Odoni, Psaraftis and Wilson (1986) is often the basis for such insertion heuristics. ADARTW considers a static DARP with a heterogeneous fleet and inserts new requests to a position in the vehicle route by the cheapest insertion criterion, i.e., minimizing the additional incremental cost caused by the insertion. Madsen et al. (1995) propose the insertion-based REBUS algorithm to solve the dynamic planning paratransit problem in Copenhagen, Denmark. Modern state-of-the-art heuristics, such as the online dynamic insertion algorithm with demand forecasts proposed by van Engelen et al. (2018), uses insertion heuristics in combination with demand forecasts to develop demand-anticipatory capabilities. Braekers et al. (2014) use an insertion heuristic to construct an initial solution for the proposed simulated annealing heuristic. Even though insertion heuristics are considered quite simple, the ability to efficiently provide feasible solutions makes these methods suitable to evaluate various operational policies and strategies (e.g., Hyland and Mahmassani, 2018).

Madsen et al. (1995) identify that the time-consuming part of insertion heuristics is to check the feasibility of an insertion. A feasible solution must be evaluated with regards to fulfilling standard flow and precedence constraints for a vehicle, as well as checking the specific modeling constraints, such as maximum ride time constraints, time window constraints, waiting time constraints, and vehicle capacity constraints. As such, efficiently determining whether a solution is feasible or not is crucial in a dynamic setting, as the schedule planner has limited time. It should be noted that this line of research is not concerned with the solution quality, but rather the worst-case complexity that defines computational time. Hunsaker and Savelsbergh (2002) propose a three-pass algorithm for a static DARP with a homogeneous fleet that determines the feasibility of a schedule in linear time. This study recognizes that service-related constraints significantly complicates the construction of high-quality schedules, and calculates the earliest feasible departure and arrival times to ensure that the constraints are fulfilled. There is, however, a known flaw with this algorithm, which for some individual cases, fails to identify a feasible solution even though it exists. This issue has been addressed among Tang, Kong, Lau and Ip (2010) and Haugland and Ho (2010). Tang et al. (2010) propose a revised algorithm with a quadratic worst-case time and define the development of efficient algorithms with linear time complexity as a promising issue for future research.

2.3 Simulation of Dial-a-Ride Services

It is essential to be able to solve DARPs efficiently, but it is also of interest to study how different ways of operating the service affect the customer and operator. The effects of how different service characteristics and policies affects the performance and efficiency of dial-a-ride services are often studied through simulation. The performance of the service is observed through key performance indicators (KPIs) that reflect both the customer's and operator's point of view. The earliest models were proposed by Heathington, Miller, Know, Hoff and Bruggeman (1968), Wilson, Sussman, Hiconnet and Goodman (1969), and Gerrard (1974) to study many-to-many DARPs. Newer simulation systems have been proposed by Deflorio (2011) and Häll, Högberg and Lundgren (2012). Deflorio (2011) accounts for possible stochastic events caused by drivers and passengers, while Häll et al. (2012) present the general-purpose simulation framework DARS to evaluate services with short call ahead times. Diana, Dessouky and Xia (2006) conduct a simulation study to determine fleet sizes given a level of service. The effects of time window settings and zoning versus no-zoning are studied through simulation in Quadrifoglio, Dessouky and Ordóñez (2008), while Häll et al. (2015) identify main parameters to consider for designing dynamic dial-a-ride services through a case study of the paratransit service in Norrköpping.

Bailey and Clark (1987) and Noda, Ohta, Shinoda, Kumada and Nakashima (2003) study the demand conditions for which dial-a-ride services perform better than fixed-route services. Bailey and Clark (1987) consider the relation between demand, service rate, and various policies in a taxi service, while Noda et al. (2003) consider alternative bus services. These studies found that the usability of the services degrades more rapidly for dial-a-ride services with an increased number of requests if the number of vehicles remained unchanged. In contrast, by increasing the number of vehicles while fixing the ratio of requests and vehicles, more possible combinations of vehicle schedules ensured that the usability increased faster for these services.

Other simulation studies have tried to define the impact of new technology on dial-a-ride services. Fu (2002) describes a model to observe how automatic vehicle locations affect the system. Hyland and Mahmassani (2018) explore various optimization-based strategies that fully autonomous vehicles allow for, and found operational efficiency gains for periods with high fleet utilization but also significantly profound effects for periods with low fleet utilization. van Engelen et al. (2018) use simulation to compare heuristic methods, and found that the proposed online insertion heuristic with demand forecasts can provide a higher level of service in terms of a reduced number of rejected requests and reduced waiting times, but with significantly increased vehicle distance driven.

2.4 Summary of Literature Review

There has been a re-emergence of studies related to dial-a-ride problems recently, especially within the area of public transportation. An overview of selected relevant simulation studies is listed in Table 2.3. As discussed, it is computationally demanding to find exact solutions to DARPs. In practical terms, very short call ahead times puts hard constraints on the computation time available for creating feasible vehicle schedules. For this reason, insertion heuristics are commonly used in simulation studies to evaluate different designs of dynamic dial-a-ride services. A general challenge with insertion heuristics is to find feasible solutions efficiently, especially with service-related constraints, such as maximum ride time constraints.

The characteristics of the DDARP-AF differ slightly from traditional problems found in the literature in that trips are not defined by a start and return to a depot, as it is assumed that the autonomous ferries can be found idle anywhere in the fjord. However, The DDARP-AF shares many of the complicating characteristics of DARPs described in the literature. The study of this problem contributes to the investigation of a dynamic dial-a-ride service with autonomous vehicles in a maritime environment. It will, therefore, comprise of describing a specific simulation model that can efficiently solve the dynamic operational planning problem. The study aims to analyze how different parameter settings affect the service level and the operational cost, to further provide decision support at the tactical level, i.e., recommendations about the fleet configuration.

Table 2.3: Overview of selected simulation studies related to DARPs.

Author	Year	Title
Bailey and Clark	1987	"A simulation analysis of demand and fleet size effects on taxicab service rates"
Deflorio	2011	"Simulation of requests in demand responsive transport systems"
Diana, Dessouky and Xia	2006	"A model for the fleet sizing of demand responsive transportation service with time windows"
van Engelen, Cats, Post and Aardald	2018	"Enhancing flexible transport services with demand-anticipatory insertion heuristics"
Fu	2002	"A simulation model for evaluating advanced dial-a-ride paratransit systems"
Gerrard	1974	"Comparison of taxi and dial-a-bus services"
Häll, Högberg and Lundgren	2012	"A modeling system for simulation of dial-a-ride services"
Häll, Lundgren and Voß	2015	"Evaluating the performance of a dial-a-ride service using simulation"
Heathington, Miller, Know, Hoff and Bruggeman	1968	"Computer simulation of a demand scheduled bus system offering door-to-door service"
Hyland and Mahmassani	2018	"Dynamic autonomous vehicle fleet operations: Optimization-based strategies to assign AVs to immediate traveler demand requests"
Noda, Ohta, Shinoda, Kumada and Nakashima	2003	"Evaluation of usability of dial-a-ride systems by social simulation"
Quadrifoglio, Dessouky and Ordóñez	2008	"A simulation study of demand responsive transit system design"
Wilson, Sussman, Hiconnet and Goodman	1969	"The use of simulation in the design of a dial-a-ride... of a computer aided routing system (CARS)"

Chapter 3

Problem Description

This chapter provides a description of the specific problem that is studied in this thesis, namely the Dynamic Dial-a-Ride problem with Autonomous Ferries in Kiel (DDARP-AF). The corresponding static mode of the problem, which was studied by Bui and Nguyen (2019), is referred to as the DARP-AF. The characteristics of the DDARP-AF is elaborated in Section 3.1. An illustrative example is presented in Section 3.2 to further assist the understanding of the reader.

3.1 Dynamic Dial-a-Ride Problem with Autonomous Ferries

The operation of DDARP-AF deals with a port-to-port ride-sharing service. More specifically, the problem deals with the assignment of a heterogeneous fleet of ferries to accommodate a set of requests in a planning period. The planning starts before all demand is known, entailing that the operator needs to update the ferry schedules in response to new incoming requests. If a request is accepted, the customer is provided with a planned pickup time, and the operator is not allowed to cancel already accepted requests. The arrival time at which a request is made known to the operator is referred to as the call-in time. A customer can book a request specifying the number of passengers to be transported from a specified pickup port to a delivery port with a time window for when the pickup is desired. Furthermore, the dynamic feature of very short call ahead times gives high flexibility for the customer, but imposes hard constraints on the available computation time for finding feasible solutions. For this reason, it is imperative that the planning procedure can determine if the request can be served or not in a fast manner while also ensure efficient deployment.

Operationally, the system aims to maximize the number of accepted requests for a planning period given that these can be feasibly served. Feasible solutions must fulfill the time window constraints and maximum ride time constraints of the requests, and ensure that the maximum ferry capacity constraints are not violated. The objective function typically reflects the perspective of the passengers or the operators. For the operator, this can be viewed as minimizing the ferry distance traveled. For the passengers, minimizing excess ride time can be considered. Overall, the design of the DDARP-AF service needs to balance the objectives of the customer and the operator. These can be understood as conflicting objectives, as improving the level of service can be directly solved with the increase in fleet size. However, this improvement in perceived service quality can be too costly for the operator. In summary, the performance of a design needs to be evaluated in terms of an acceptable level of service while limiting the costs of operation. These effects can be studied in a simulation model to provide decision support with regards to a recommended fleet configuration.

Two important considerations that can affect the provided service level is the allowed pickup time window width and allowed maximum ride time. From a practical view, a wider time window width makes it less likely that the customer is served close to the actual desired pickup time. In return, this provides greater flexibility in planning for the operator. Likewise, an increase in maximum ride time for a trip provides operational flexibility but leads to a potential increase in excess ride times. The operator can take these considerations into account through the booking system or by relaxing the routing policy. Generally, an increase in the allowed pickup time window width and maximum ride time is efficient for the operator but undesirable by the passengers. It is, therefore, essential to find reasonable values considering the aforementioned conflicting objectives.

The modeling of DDARP-AF assumes no depots, no en-route rerouting, and no transfer. The latter assumption means that it is not possible to deliver the passengers to an intermediate stop to be then served by another ferry. It is also assumed that if waiting occurs, the waiting happens at the origin node for the pair of nodes linked in a ferry route. As observed in the depicted example in Figure 3.1, the ferry will leave node 1 at the time such that it arrives at the destination node at the planned time of service for that node. Alternatively, the ferry could travel to node 4 at the first opportunity and wait for the start of service. However, the former is chosen to provide schedule flexibility if a new request could be served in-between the period that the ferry would have waited. Intuitively, the latter would result in an increased ferry mileage as it makes an unnecessary detour, e.g., the route 1-4-2-4 rather than 1-2-4.

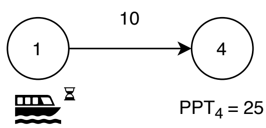


Figure 3.1: Waiting occurs at the origin node for connected nodes in a ferry route. Here, node 4 represents a pickup node with a planned pickup time at $T = 25$. The ferry waits until $T = 15$ to leave node 1.

3.2 Illustrative Example of the DDARP-AF

To further grasp the operational challenges that the operator faces when planning routes and scheduling ferry itineraries in the DDARP-AF, Figure 3.2 illustrates an example of the problem. Here, the system is represented by the two similar ferries F1 and F2 at time $T = 5$. Pickup nodes are displayed as circles, and the corresponding delivery nodes for the same requests are displayed as triangles in the same color. The brackets define the time window in which the passenger desires to be picked up. There are currently two requests known for the time being. Ferry F1 has already been assigned to serve request 1 and is en-route to the corresponding pickup node. Ferry F2 is currently idle at an arbitrary point in the fjord. At $T = 5$, request 2 is called in, and both ferries can feasibly serve it. For the remaining example outline, it is assumed that the ferry capacity constraints are not violated at any given point.

Since both ferries can feasibly serve the new request, the solution choice is based on the operator's routing policy reflected in the objective function. As previously stated, the objective could typically be to minimize the excess ride time for each request. With this in mind, the preferred solution would be that the idle ferry F2 serves request 2. If otherwise, the passengers associated with request 1 would experience an increase in excess ride time due to visiting the pickup node (2) before the delivery node (4). The route for ferry F2 is updated accordingly, as illustrated in Figure 3.3.

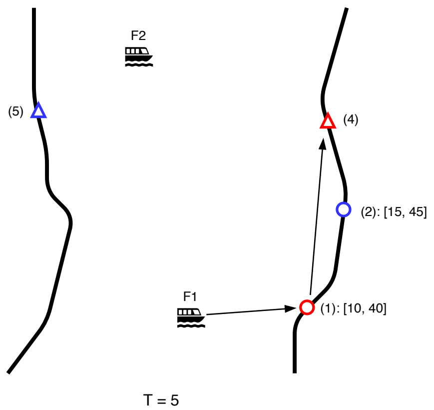


Figure 3.2: Ferry F1 is en-route to pickup node 1, while F2 is currently idle in the fjord. Request 2 is called in at $T = 5$ with a desired pickup time window. Both ferries can feasibly serve the new request.

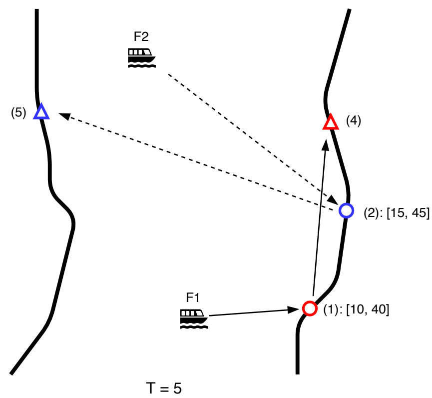
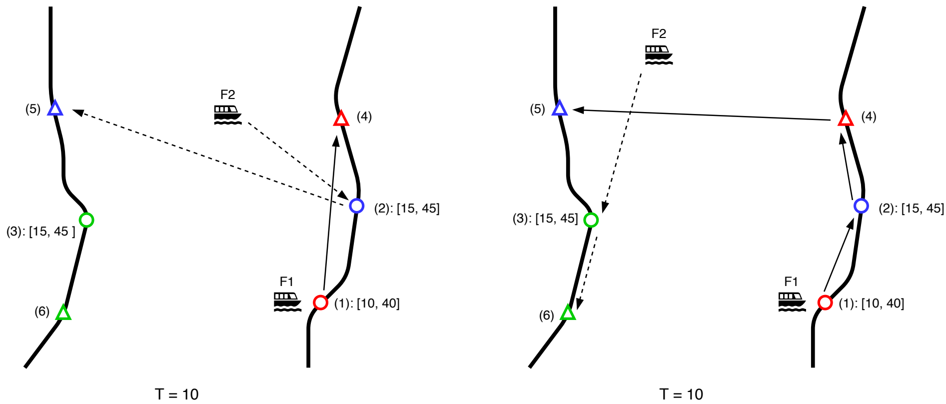


Figure 3.3: The idle ferry F2 is scheduled to serve request 2, and the route is updated with the corresponding pickup node (circle) and delivery node (triangle).

Furthermore, Figure 3.4a presents another incoming request at $T = 10$. However, due to the combination of the desired pickup time window and max ride time constraints for the passengers already on board, neither F1 nor F2 can feasibly serve the request. Consequently, the request is rejected. Figure 3.4b illustrates the case where all three requests can be served by changing only the preference for picking feasible solutions. Here, the same instance is considered, but the objective at each decision point is to minimize the total ferry distance traveled. The incremental increase in total ferry distance traveled is less for serving with ferry F1 than F2, and this would make F1 the preferred choice to accommodate request 2, known at $T = 5$. At time $T = 10$, the idle ferry F2 can serve request 3 within the desired pickup time window, and the route is updated accordingly.



(a) Minimize excess ride time **(b)** Minimize ferry distance traveled

Figure 3.4: Request 3 is called in at $T = 10$. Figure (a) shows the routing policy with minimizing excess ride rime as the objective. The system is forced to reject request 3, as it cannot be feasibly served by F1 nor F2. Figure (b) shows the same instance where the feasible solutions are chosen according to minimizing overall ferry distance traveled. In this case, request 2 would have been assigned to F1 at $T = 5$, and request 3 could be feasibly served by F2.

It is important to note that even though the policy change illustrated in Figure 3.4b made it possible to serve all three requests, it does not necessarily yield better long-term performance. Another scenario in a later period could, for instance, make the change unfavorable and vice versa. Short-term performance does not necessarily imply an equally long-term performance of a system. In this case, the operator defines a deliberate strategy to make the best decision based on the presented information at the time of decision. The long-term performances of the service need to be studied under the same conditions, to determine whether a design choice is preferable over another. As previously mentioned, the performance is not only judged by the demand met, but the operator tries to balance a varied set of performance indicators reflecting the customers' perceived service level and operational costs. This consideration exemplifies why simulation as a tool is suitable and necessary to evaluate the effects of specific service characteristics under different demand scenarios for the DDARP-AF.

Chapter 4

Mathematical Formulation

In this chapter, a mathematical formulation of the DARP-AF is presented. The modeling approach and assumptions are described in 4.1. The notation used to formulate the model is introduced in Section 4.2. The objective function and constraints are presented in Section 4.3 and Section 4.4. Finally, the relevance of the model with regards to the DDARP-AF is discussed in Section 4.5. It is emphasized that the mathematical model presented in this chapter is equivalent to the one formulated by Bui and Nguyen (2019).

4.1 Modeling Approach and Assumptions

A three-index formulation of the DARP-AF is necessary to keep track of the initial load on board, time windows, origin, and destination specifically for each ferry in the fleet. The modeling approach builds on a structurally similar problem from the maritime industry, namely the tramp ship routing and scheduling problem described by Christiansen and Fagerholt (2014). However, the main difference that needs to be considered is the transportation of passengers rather than cargoes. The distinction is expressed in the presented model by introducing passenger inconvenience through both hard and soft constraints. Excessive ride time duration is the main disutility considered in this model.

Some underlying assumptions are made to reduce the complexity of the model. First, it is assumed that all types of ferries and multiple ferries can berth at all ports for any given time, even though this could depend on the port characteristics, e.g., port size. As the system has a limited fleet in disposal, the requests with a higher number of passengers are prioritized. This assumption can be understood as more cost-efficient to the operator, as sailing time per passenger would be higher for feasibly serving the same number of passengers spread out on several requests. If a request has been accepted, the operator cannot cancel it, as it would be highly inconvenient for the passengers. Also, new passengers cannot exceed the maximum capacity of the ferry, and the passengers associated with a specific request cannot be split between several ferries. The assumption is considered reasonable from a practical point of view, as a group of passengers disallowed to travel together represents an apparent inconvenience for the passengers. Lastly, deterministic conditions are assumed, i.e., sailing times are, for instance, not affected by ferry breakdown or weather conditions. Consequently, passenger no-shows and passenger cancellations are not considered.

4.2 Notation

Let each request be represented by an index i. Each request i has an associated pickup node i and delivery node $n + i$, where n denotes the number of requests that might be handled during the planning horizon. Each node represents a port, but it is imperative to note that different nodes may correspond to the same physical port. For instance, given two requests, the associated pickup nodes $i = 1$ and $i = 2$ can both correspond to Bahnhof. This would imply that the distance and travel time between the nodes would equal zero, and could represent a servicing of two requests at the same physical port. Moreover, let $\mathcal{N}^P = 1, 2..., n$ be the set of pickup nodes, and $\mathcal{N}^D = n + 1, n + 2, ..., 2n$ be the set of delivery nodes. The set of pickup nodes is further partitioned into two subsets; a set of already accepted requests $\mathcal{N}^A$, that is mandatory to fulfill, and a set of optional requests $\mathcal{N}^O$.

Further, let $\mathcal{V}$ be the set of ferries. Each ferry v has an associated network $(\mathcal{N}_v, \mathcal{A}_v)$. N_v denotes the set of nodes that can be visited by ferry v, including the origin $o(v)$ and artificial destination $d(v)$ for ferry v. Practically, the origin $o(v)$ can geographically be a port or any location in the fjord, while the artificial destination $d(v)$ represents the last planned delivery port for ferry v. Accordingly, $d(v)$ reflects the same location as $o(v)$ if ferry v is not used. The set N_v is used to improve solving time, as the nodes that ferry v cannot service in time due to its current location are excluded from the set. Here, the sets of pickup and delivery nodes that ferry v may visit can be derived as $\mathcal{N}_v^P = \mathcal{N}^P \cap \mathcal{N}_v$ and $\mathcal{N}_v^D = \mathcal{N}^D \cap \mathcal{N}_v$, respectively. The set $\mathcal{A}_v$ contains all the feasible arcs for ferry v, which is a subset of $\mathcal{N}_v \times \mathcal{N}_v$.

For each ferry $v \in \mathcal{V}$ and each arc $(i, j) \in \mathcal{A}_v$, let T_{ijv}^S denote the sailing time from node i to node j, while T_{ijv}^B represents the berthing time (including embarking and disembarking) at node i. If node i and j corresponds to the same port, the berthing time is equal to zero. Each request i has an associated number of passengers P_i that needs to be transported, and a time window $\left[\underline{T}_{iv}, \overline{T}_{iv}\right]$ for ferry v associated with pickup node i. $\underline{T}_{iv}$ and $\overline{T}_{iv}$ defines the earliest and latest possible time for starting service at node i for ferry v, respectively. In practice, the time window is specific to the request, but the index v accounts for a heterogeneous fleet. Thus, a ferry with a higher sailing speed would have a smaller time window than a slower ferry and vice versa. $T_{i,n+i}^R$ denotes a reference direct ride time from the pickup node i to the delivery node $n + i$. Further, let T_i^{MAX} define a maximum ride time coefficient associated with request i, such that the maximum allowed ride time is proportional to the direct sailing time. The capacity of ferry v is denoted K_v.

The binary variable x_{ijv} is assigned the value 1 if ferry v sails directly from node i to node j, and 0 otherwise. Likewise, the binary variable y_i is assigned the value 1 if the optional request i is accepted, and 0 otherwise. Lastly, the variable t_{iv} represents the time for starting service at node i for ferry v, whereas the variable l_{iv} signifies the load, i.e., the number of passengers on board ferry v when leaving node i.

Indices

i, j	Nodes for pickup and delivery associated with a request
v	Ferry

Sets

$\mathcal{V}$	Set of ferries
$\mathcal{N}^A$	Set of accepted requests
$\mathcal{N}^O$	Set of optional requests
$\mathcal{N}_v$	Set of nodes that ferry v can visit
$\mathcal{N}_v^P$	Set of pickup nodes that ferry v can visit
$\mathcal{A}_v$	Set of feasible arcs for ferry v

Parameters

T_{ijv}^S	Sailing time from node i to node j for ferry v
T_{ijv}^B	Berthing time at node j for ferry v. If node i and j corresponds to the same port, the berthing time is equal to zero.
P_i	Number of passengers for request with pickup node i
$\underline{T}_{iv}$	Earliest possible start of service at node i for ferry v
$\overline{T}_{iv}$	Latest possible start of service at node i for ferry v
$T_{i,n+i}^R$	Reference direct ride time from pickup node i to delivery node $n+i$
T_i^{MAX}	Coefficient for maximum ride time for request with pickup node i
K_v	Capacity of ferry v

Variables

$$x_{ijv} = \begin{cases} 1, & \text{if ferry } v \text{ sails directly from node } i \text{ to node } j \\ 0, & \text{otherwise} \end{cases}$$

$$y_i = \begin{cases} 1, & \text{if request with pickup at node } i \text{ is accepted} \\ 0, & \text{otherwise} \end{cases}$$

t_{iv}	Time for starting service at node i for ferry v
l_{iv}	Number of passengers on board when ferry v is leaving node i

4.3 Objective Function

The objective function (4.1) maximizes the total number of passengers transported, weighted by the direct sailing time associated with each optional request to prioritize longer trips. Thus, a request with a direct ride time of, e.g., 30 minutes, is prioritized over another with five minutes, as it is assumed less inconvenient for the passengers to find alternative transportation for the latter. The second term represents a weighted penalty for excessive ferry trip durations. The binary flow variable $x_{i,n+1,v}$ assures that the second term is only active when the request is served, and ferry v does not sail directly from node i to node $n + i$. For each request i, the weighted penalty coefficient W_i should be determined such that $P_i W_i(t_{n+i,v} - t_{iv} - T^R_{i,n+1} x_{i,n+i,v}) < P_i T^R_{i,n+i} y_i$, to ensure a higher emphasis on the first term. The objective function incentivizes acceptance of requests, as a better objective value cannot be achieved by rejecting a request.

$$\max \sum_{i \in \mathcal{N}^O} P_i T^R_{i,n+i} y_i - \sum_{v \in \mathcal{V}} \sum_{i \in \mathcal{N}^O \cup \mathcal{N}^A} P_i W_i (t_{n+i,v} - t_{iv} - T^R_{i,n+i} x_{i,n+i,v}) \tag{4.1}$$

4.4 Constraints

The following section presents the different types of constraints related to the DARP-AF. Firstly, the constraints related to flow of the network are presented in Section 4.4.1. The load and time constraints are presented in Section 4.4.2 and Section 4.4.3, respectively. Linearization of relevant constraints is also discussed in the respective subsections. Lastly, the non-negativity and binary constraints are given in Section 4.4.4.

4.4.1 Flow Constraints

The constraints defined in this section are concerned with the flow into and out of the nodes. Constraints (4.2) state that all accepted requests must be fulfilled by a ferry v, likewise the constraints (4.3) ensures the same for optional requests if accepted. Constraints (4.4) – (4.6) describe the flow on the sailing route used by ferry v, from the origin node to the artificial destination node. Lastly, constraints (4.7) ensure that the same ferry v visits both the pickup node i and the delivery node $n + i$.

$$\sum_{v\in\mathcal{V}}\sum_{j\in\mathcal{N}_v} x_{ijv} = 1, \qquad i \in \mathcal{N}^A \quad (4.2)$$

$$\sum_{v\in\mathcal{V}}\sum_{j\in\mathcal{N}_v} x_{ijv} - y_i = 0, \qquad i \in \mathcal{N}^O \quad (4.3)$$

$$\sum_{j\in\mathcal{N}_v} x_{o(v)jv} = 1, \qquad v \in \mathcal{V} \quad (4.4)$$

$$\sum_{j\in\mathcal{N}_v} x_{ijv} - \sum_{j\in\mathcal{N}_v} x_{jiv} = 0, \qquad v \in \mathcal{V}, i \in \mathcal{N}_v\backslash\{o(v), d(v)\} \quad (4.5)$$

$$\sum_{i\in\mathcal{N}_v} x_{id(v)v} = 1, \qquad v \in \mathcal{V} \quad (4.6)$$

$$\sum_{j\in\mathcal{N}_v} x_{ijv} - \sum_{j\in\mathcal{N}_v} x_{n+i,jv} = 0, \qquad v \in \mathcal{V}, i \in \mathcal{N}_v^P \quad (4.7)$$

4.4.2 Load Constraints

Constraints (4.8) and (4.9) keep track of the number of passengers on board a ferry v after visiting the pickup and delivery nodes, respectively. Constraints (4.8) imply that if ferry v sails arc (i, j) from a node i to a pickup node j, the load after visiting node j is equal to the number of passengers on board after leaving node i including the passengers picked up at node j. The corresponding relation for delivery nodes are stated by constraints (4.9). Similarly, constraints (4.10) and (4.11) ensure that the capacity of ferry v is not violated after visiting the pickup and delivery nodes, respectively. It should be noted that constraints (4.11) state that the load after visiting a delivery node must lie between zero and the ferry capacity less the number of passengers delivered at node $n + i$ (rather than the ferry capacity), which provides a tighter formulation.

$$(l_{iv} + P_j - l_{jv})x_{ijv} \le 0, \qquad v \in \mathcal{V}, (i, j) \in \mathcal{A}_v | j \in \mathcal{N}_v^P \quad (4.8)$$

$$(l_{iv} - P_j - l_{n+j,v})x_{i,n+j,v} \le 0, \qquad v \in \mathcal{V}, (i, n + j) \in \mathcal{A}_v | j \in \mathcal{N}_v^P \quad (4.9)$$

$$\sum_{j\in\mathcal{N}_v} P_i x_{ijv} \le l_{iv} \le \sum_{j\in\mathcal{N}_v} K_v x_{ijv}, \qquad v \in \mathcal{V}, i \in \mathcal{N}_v^P \quad (4.10)$$

$$0 \le l_{n+i,v} \le \sum_{j\in\mathcal{N}_v} (K_v - P_i)x_{n+i,jv}, \qquad v \in \mathcal{V}, i \in \mathcal{N}_v^P \quad (4.11)$$

Linearization of Load Constraints

Constraints (4.8) and (4.9) are nonlinear and need to be linearized to efficiently solve the model in a commercial optimization solver. Big M formulations are used to linearize these constraints. The Big M values should be set as small as possible (but sufficiently large) to provide a tight formulation. The load after visiting a pickup node j can at most equal the capacity of ferry v. Hence, the ferry capacity K_v is chosen as the Big M coefficient. Note that the load constraints could be formulated as a combined set of constraints, but is kept separate with regards to pickup and delivery nodes. This provides a somewhat stronger formulation with increased solvability at the expense of some readability. The reformulated linearized load constraints are given by constraints (4.12) and (4.13).

$$l_{iv} + P_j - l_{jv} - K_v(1 - x_{ijv}) \leq 0, \qquad v \in \mathcal{V}, (i,j) \in \mathcal{A}_v | j \in \mathcal{N}_v^P \quad (4.12)$$

$$l_{iv} - P_j - l_{n+j,v} - K_v(1 - x_{i,n+j,v}) \leq 0, \qquad v \in \mathcal{V}, (i,n+j) \in \mathcal{A}_v | j \in \mathcal{N}_v^P \quad (4.13)$$

4.4.3 Time Constraints

Constraints (4.14) ensures that the time for starting service at node j must be greater than the departure time from the previous node i plus the sailing time between the nodes. Constraints (4.15) force ferry v to visit pickup node i before the corresponding delivery node $n+i$. Constraints (4.16) ensure that the time it takes from starting service at node i to starting service at node $n + i$ does not exceed a maximum allowed ride time, expressed as a value proportional to the berthing time and direct sailing time between the nodes. The time window within which service at pickup node i must start is defined by constraints (4.17). The sum of binary variables x_{ijv} is included to prevent the time variables taking an arbitrary value within the time windows for that (i,v)-combination, effectively forcing the starting time to zero if node i is not visited by ferry v. This careful consideration is necessary to prevent numerical errors in calculating the objective value given by the objective function (4.1).

$$(t_{iv} + T_{ijv}^B + T_{ijv}^S - t_{jv})x_{ijv} \leq 0, \qquad v \in \mathcal{V}, (i,j) \in \mathcal{A}_v \quad (4.14)$$

$$t_{iv} + \sum_{j \in \mathcal{N}_v} (T^B_{ijv} + T^S_{i,n+i,v}) x_{ijv} - t_{n+i,v} \leq 0, \qquad v \in \mathcal{V}, i \in \mathcal{N}^P_v \quad (4.15)$$

$$t_{n+i,v} - t_{iv} \leq (T^B_{ijv} + T^S_{i,n+i,v}) T^{MAX}_i, \qquad v \in V, i \in \mathcal{N}^P_v \quad (4.16)$$

$$\sum_{j \in N_v} \underline{T}_{iv} x_{ijv} \leq t_{iv} \leq \sum_{j \in N_v} \overline{T}_{iv} x_{ijv}, \qquad v \in \mathcal{V}, i \in \mathcal{N}^P_v \quad (4.17)$$

Linearization of Time Constraints

Constraints (4.14) are nonlinear, and are linearized through a Big M formulation. Similar to the linearized load constraints, the value of the Big M coefficient should be set with regards to a tight formulation. The reformulated linearized form are then given by constraints (4.18), where the Big M coefficient can be calculated as $M_{ijv} = max(0, \overline{T}_{iv} + T^B_{ijv} + T^S_{ijv} - \underline{T}_{jv})$.

$$t_{iv} + T^B_{iv} + T^S_{ijv} - t_{jv} - M_{ijv}(1 - x_{ijv}) \leq 0, \qquad v \in \mathcal{V}, (i, j) \in \mathcal{A}_v \quad (4.18)$$

4.4.4 Non-negativity and Binary Constraints

The non-negativity requirements for the time and load on board ferry v are given by constraints (4.19) and (4.20). Constraints (4.21) and (4.22) impose the binary requirements on the flow and optional request variables, respectively.

$$t_{iv} \geq 0, \qquad v \in \mathcal{V}, i \in \mathcal{N}_v \quad (4.19)$$

$$l_{iv} \geq 0, \qquad v \in \mathcal{V}, i \in \mathcal{N}_v \quad (4.20)$$

$$x_{ijv} \in \{0, 1\}, \qquad v \in \mathcal{V}, (i, j) \in \mathcal{A}_v \quad (4.21)$$

$$y_i \in \{0, 1\}, \qquad i \in \mathcal{N}^O \quad (4.22)$$

4.5 Relevance to the DDARP-AF

The modeling of the DDARP-AF is similar to the modeling of the DARP-AF. In the dynamic case, the mathematical model can be solved for the set of optional requests $\mathcal{N}^O$, containing the single incoming request. The set $\mathcal{N}^A$ contains the existing accepted requests in the system, and the corresponding decision variables are fixed accordingly. These are treated as mandatory to fulfill by constraints (4.2). However, as supported by the literature, optimal solutions are challenging to find within a relatively short amount of time. In other words, exact methods for solving the DDARP-AF does not scale well due to the number of constraints growing exponentially as more requests or ferries are introduced to the system.

Even though a dynamic dial-a-ride service is considered, some requests can be known depending on the determined policy of the ferry service. For instance, consider a case where the ferry service does not operate during the night, but bookings can be made for the following morning. Bui and Nguyen (2019) found that the DARP-AF can be solved for instances up to 100 requests and 30 ferries with reasonably small optimality gaps within half an hour. Given the requests at hand, an initial solution to the DDARP-AF can then be obtained by solving the initial static case.

Chapter 5

Simulation Model

This chapter presents the a process-oriented discrete-event simulation model to evaluate the performance of the DDARP-AF. The notation to further describe the simulation model and processes are presented in Section 5.1.1. Section 5.2 describes the simulation flow, and Section 5.3 describes the performance metrics that are derived from the simulation output.

5.1 Model Introduction

There are two main processes in this simulation model: *Incoming request process* and *Operate ferry process*. Furthermore, as can be seen in Figure 5.1, the structure of the simulation model consists of different components. The *Input manager* handles the initialization of the simulation based on the input provided. The *Request handler* controls the scheduling of each incoming request by triggering the replanning procedure *Incoming request process*. The process ensures that that the request is feasibly inserted based on the system state provided by the *Simulation controller*. If a request is scheduled to be served by a ferry v, the process *Operate ferry process* v is triggered. This process serves as an event controller for each ferry v in the fleet, updating the system state at every scheduled pickup and delivery event. When the simulation has ended, the *Output generator* provides statistics based on the final observed system state. The modular structure of the simulation model is intended to ease the implementation of potential model extensions, as the components divide the processes by separate functions.

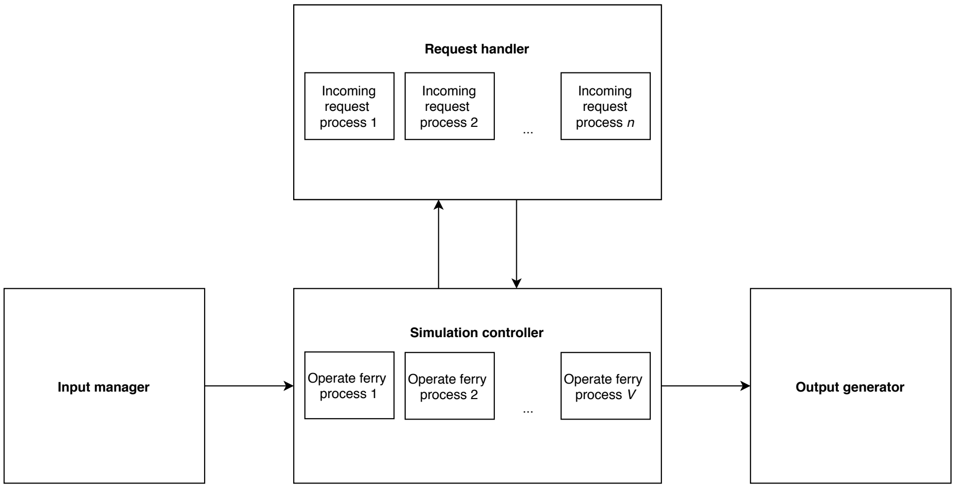


Figure 5.1: Modular structure of the simulation model.

5.1.1 Notation

Additional notation is introduced to further describe the processes of the simulation model. The notation presented in Section 4.2 is further kept, unless otherwise stated.

$\underline{CAT}$	Earliest call ahead time
$\overline{CAT}$	Latest call ahead time
PPT_i	Planned pickup time for request i
TW_i	Pickup time window width for request i
RT_i	Ride time for request i
MRT_i	Max ride time associated with request i
L_v	Current location of ferry v
R_v	Route for ferry v
CT_v	Current time for ferry v
Q_v	Number of passengers on board ferry v

D_v	Total distance associated with travelling route R_v for ferry v
$\underline{T}_i^P$	Earliest start of service at the pickup node for request i for ferry v
$\overline{T}_i^P$	Latest start of service at the pickup node for request i for ferry v
$\underline{T}_i^D$	Earliest start of service at the delivery node for request i for ferry v
$\overline{T}_i^D$	Latest start of service at the delivery node for request i for ferry v
P^{MAX}	Maximum number of passengers associated with a request
λ	Mean arrival rate of requests

5.1.2 Simulation Rules

In addition to the assumptions made in Chapter 3, some simulation rules are defined to efficiently model the ferry service. In reality, these simulation rules can be interpreted as some general routing and booking policy of the ferry service.

Rule 1 A request is either accepted or rejected.

Rule 2 If a request i is accepted, the scheduled ferry cannot deviate from the given planned pickup time PPT_i.

Rule 3 A customer can book a trip at latest $\underline{CAT}$ prior to the earliest desired pickup time.

Rule 4 A customer can book a trip at most $\overline{CAT}$ time in advance.

Rule 5 An idle ferry is always preferred if it can feasibly serve a new request.

Rule 6 If the next node in the ferry route R_v is a delivery node, ferry v sails directly to the next node after visiting the current node.

The first rule entails that the response provided by the operator is either acceptance or rejection. Consequently, the customer has to book a new request if the initial trip was rejected. The second rule provides the customer with a planned pickup time PPT_i if request i has been accepted. The scheduled ferry cannot deviate from the PPT_i to limit passenger inconvenience. Rules three and four define the bounds for how early and late the customer is allowed to book a trip, e.g., the passenger cannot expect to be served two minutes after booking a trip or book a trip two years ahead. As requests are known in an online matter throughout the planning period, the fifth rule intends to limit excess ride times through higher fleet utilization. Lastly, for the same reason, if the next node is a delivery node, the ferry is not allowed to wait at the current node as this would cause an increase in excess ride times. Note that this rule does not mean that waiting is not allowed if passengers are on board a ferry.

5.2 Simulation Flow

The simulation flow can be observed through the flowchart presented in Figure 5.2. First, the network is given as a distance matrix. The travel times between the nodes are calculated based on the distance and average speeds of the ferries on the links. Demand is represented as a set of requests generated beforehand, with the interarrival times generated by a Poisson process with mean arrival rate λ. Parameters describing the fleet, potential initial solutions, and the objective function for the replanning procedure are also given as inputs.

Given the inputs, the simulation model is initialized accordingly. Given that the set of requests is not empty, the first event is always the first incoming request i with the earliest call-in time. The replanning procedure is then triggered, and the DARP is solved according to the insertion heuristic, described in Chapter 6. If a feasible solution is found, the trip is inserted in the route R_v for the scheduled ferry v, and the total sailing distance of ferry v, D_v, is updated. PPT_i is updated for the newly inserted request i. If no solution is found, the request is added to a list of rejected requests. The simulation then checks for the next event in the event queue. Next events are either a new incoming request as previously, or a pickup/delivery event. If the next event is a pickup/delivery event, RT_i is updated for all requests i if the corresponding pickup node i has been visited. Correspondingly, the route R_v, location L_v, current time CT_v, number of passengers on board Q_v, and total route distance D_v are updated to reflect the current state of ferry v. The simulation progresses in the same loop until the event queue is empty, and the simulation is ended. The output of the simulation is reflected in the final system state, providing information such as the ferry routes, the set of rejected requests, ride times, ferry distances traveled, and ferry idle times.

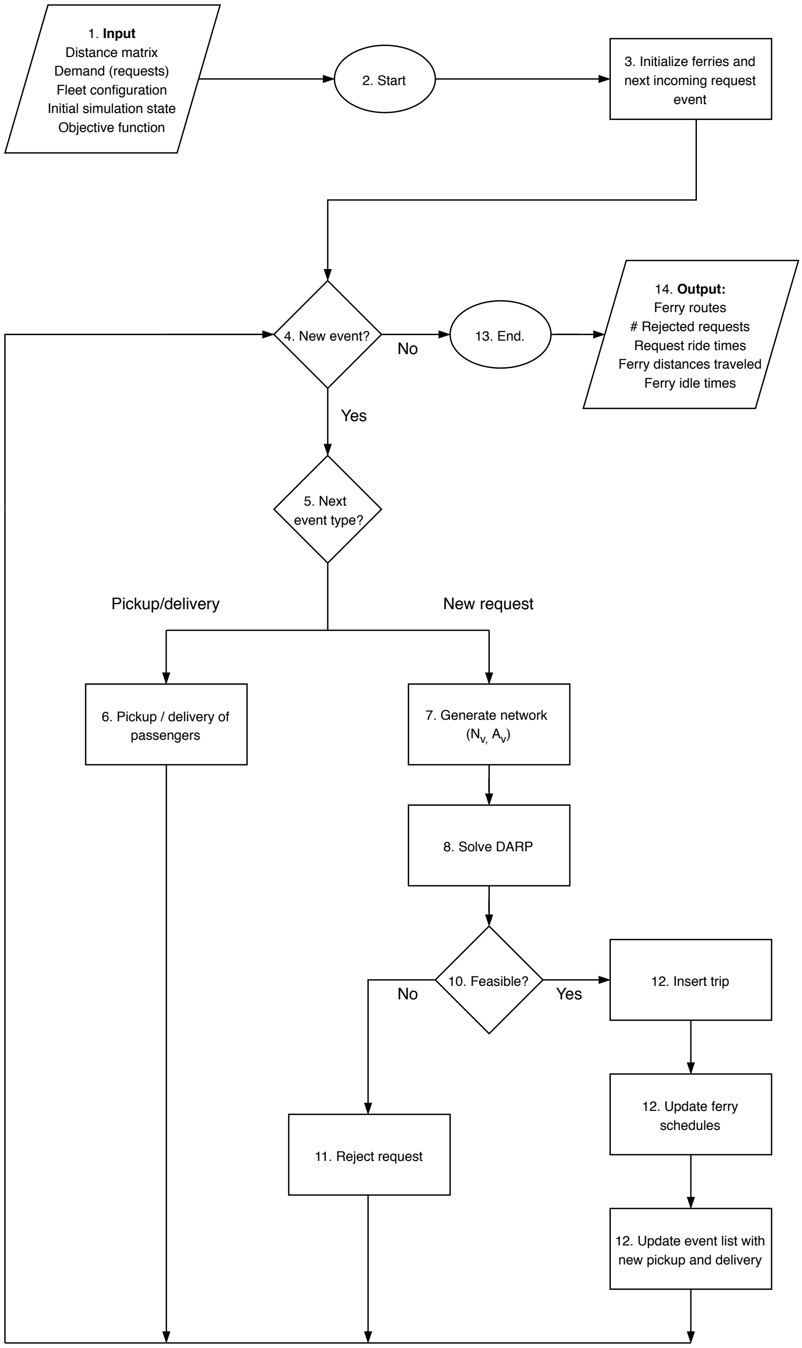


Figure 5.2: Overview of the simulation flow.

5.3 Simulation Output

The output of a simulation run provides general statistics about the system that can further be used to formulate various key performance indicators (KPIs) as evaluation metrics. The computational study conducted in Chapter 7 measures the efficiency of the ferry service with regards to the following KPIs.

- **Percentage rejected requests.** The number of rejected requests, divided by the total number of generated requests.
- **Percentage rejected passengers.** The number of passengers associated with every rejected requests, divided by the the total number of passengers generated with each request.
- **Average excess ride time per request.** The ride time less the associated direct sailing time for each accepted request, divided by the total number of accepted requests.
- **Average distance per ferry.** The total distance traveled by all ferries, divided by the number of ferries.
- **Average distance per request.** The total distance traveled by all ferries, divided by the number of accepted requests.
- **Average idle time per ferry.** The total operating time for all ferries, less the total final simulation time.

Chapter 6

Insertion of Requests

This chapter describes the insertion heuristic used to solve the planning problem at each incoming request. Section 6.1 describes the details of the overall procedure of handling an incoming request. The constraints needed to be fulfilled for feasible insertions are presented in Section 6.2, and lastly Section 6.3 discusses how the objective function is used to determine which of the feasible insertions is carried out.

6.1 Insertion Heuristic

As discussed in Chapter 3, the highly dynamic nature of the DDARP-AF makes insertion heuristics a viable solution method due to the ability to promptly find feasible solutions. The approach proposed in this chapter follows the same general greedy procedure first developed by Jaw et al. (1986). Each time a new request is called in, the procedure tries to insert the request into the existing ferry schedules in the most cost-efficient manner. Note that an insertion consists of adding two nodes, i.e., the pickup and delivery node of the request, into a ferry route. The insertion heuristic is performed according to these steps:

1. For each ferry, a time window check is first performed and then the max ride time and ferry capacity constraints are checked for feasible insertions into the route. For all feasible insertions, the objective value is computed according to an objective

function described in Section 6.3.

2. Insert the two nodes of the given request in the insertions with the lowest incremental cost added to the objective value.

3. If no feasible insertion exists, the request is added to a list of rejected requests.

When checking for schedule feasibility, the time for the start of service at the delivery nodes must be within $[\underline{T}_i^D, \overline{T}_i^D]$ for feasible insertions. Figure 6.1 illustrates how the delivery time window is constructed before a request i has been accepted. The earliest possible start of service at the delivery node is given as the earliest start of service at the pickup node plus the direct sailing time between the nodes after leaving the pickup node. Correspondingly, the latest start of service at the delivery node is calculated as the latest start of service at the pickup node plus the maximum ride time after leaving the pickup node. Figure 6.2 illustrates the construction of the delivery time window after a request i has been accepted. If request i has been accepted, the customer is provided with the planned pickup time PPT_i, and the delivery time window is calculated from the planned pickup time plus the direct sailing time or maximum ride time after leaving the pickup node.

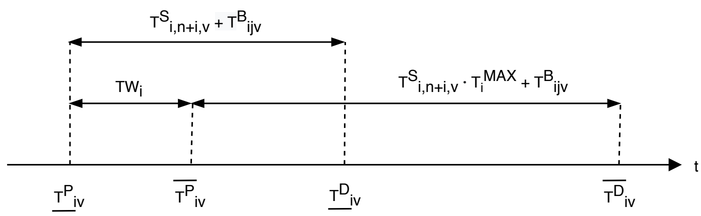


Figure 6.1: Relations for calculating the delivery time window for request i.

6.2 Feasibility Testing

This section presents the procedures for feasibility testing with respect to pickup time window constraints, max ride time constraints, as well as ferry capacity constraints. The insertion of a request requires the constraints to be checked for all requests already accepted, such that schedule feasibility is maintained for the affected requests as well. The

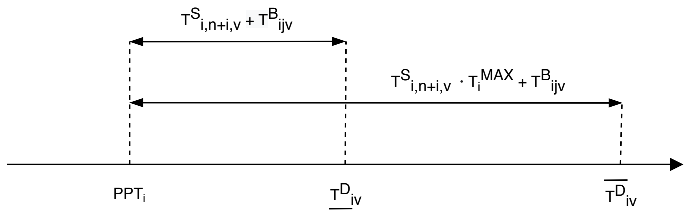


Figure 6.2: Relations for calculating the updated delivery time window after acceptance of request i.

overall combined procedure for feasibility testing performs at O(n^2) worst-case complexity, matching the performance of the revised method of Tang et al. (2010). Here, n denotes the number of requests.

Hunsaker and Savelsbergh (2002) point out that it is not clear how to quickly verify the feasibility of an insertion at the presence of complicating constraints, such as maximum ride time restrictions. However, the procedure presented in this section leverages the sixth rule presented in Section 5.1. Since ferry v does not wait if the next node in R_v is a delivery node, the earliest possible departure and arrival time at the delivery node can be calculated from the nearest prior pickup node. Figure 6.3 illustrate how this is calculated from PPT_i of a pickup node i, which the ferry cannot deviate from. This relation is used in both the procedure to verify time window feasibility and maximum ride time feasibility. The following subsections present the procedures of the three feasibility checks, respectively.

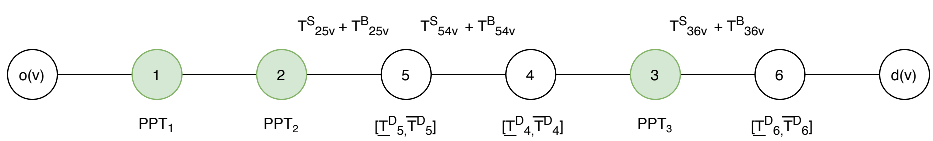


Figure 6.3: Route for a ferry v. The illustration depicts how feasibility is ensured through calculating the earliest possible arrival time at a delivery node (white) by using the nearest prior pickup node (green).

Time window feasibility

The pickup time window constraints ensure that the time for the start of a service at a pickup node is within the specified period provided by the customer. The procedure for the time window feasibility check is performed in quadratic worst-case time and is presented in Algorithm 1. First, a copy of the ferry route R_v is initialized, and the pickup node i and delivery node $n+i$ for a request i is inserted in the copied route. The planned pickup time PPT_i for the request i is updated with a pickup time within the pickup time window $[\underline{T}_i^P, \overline{T}_i^P]$. The procedure then iterates through all the pickup nodes in the copied route and identifies the position for each pickup node i. If the node before the pickup node i is a delivery node, it finds the nearest prior pickup node for use as a reference. This incident corresponds to the evaluation of pickup node 3, illustrated in Figure 6.3. Here, the pickup node 2 is used as a reference, and the earliest arrival time is calculated by adding the sailing and service times at each node in-between the evaluated node and the reference node. Similarly, if the node before the pickup node i is another pickup node, this corresponds to the evaluation of pickup node 2 with pickup node 1 as a reference. As long as the calculated earliest arrival time at the evaluated pickup node i is less or equal to the planned pickup time PPT_i, the insertion is feasible.

Maximum ride time feasibility

The maximum ride time constraints ensure that the passengers associated with a request at most experience a tolerable excess ride time. The procedure for the maximum ride time feasibility check is performed in quadratic worst-case time and is presented in Algorithm 2. The essence of this procedure resembles the procedure of the pickup time window feasibility. First, a copy of R_v is initialized, and the corresponding nodes for a request i are inserted accordingly. PPT_i is updated with a pickup time within the pickup time window $[\underline{T}_i^P, \overline{T}_i^P]$. The procedure then iterates through all the pickup nodes i in the copied route and finds the position of the corresponding delivery node $n+i$. If the node before the delivery node $n+i$ is a delivery node, it finds the nearest prior pickup node for use as a reference. This incident corresponds to the evaluation of the delivery node 4 in Figure 6.3 with the pickup node 2 as a reference. The added ride time is calculated by

Algorithm 1: Time window feasibility

Initialize a copy of the ferry route R_v and insert the pickup node and delivery node for request i. Update PPT_i for the corresponding pickup node i to a pickup time within $[\underline{T}_i^P, \overline{T}_i^P]$. Calculate the updated delivery time window $[\underline{T}_i^D, \overline{T}_i^D]$.

for *all pickup nodes* i *in the copied route* **do**
    Find position of i in the route
    $ferryTime \leftarrow CT_v$
    $travelTime \leftarrow 0$
    if i *is the next node* **then**
        $travelTime \leftarrow travelTime$+sailing time from L_v to i
    else
        $nodeBeforePnode \leftarrow$ node before i
        $travelTime \leftarrow$ sailing time from $nodeBeforePnode$ to i
        if $nodeBeforePnode$ *is a pickup node* **then**
            $ferryTime \leftarrow$ PPT for $nodeBeforePnode$ + T_{ijv}^B
        else
            Find position of $lastPnode$ before i
            $ferryTime \leftarrow$ PPT for $lastPnode$ + T_{ijv}^B
            for *all nodes between* $lastPnode$ *and* i **do**
                $ferryTime \leftarrow ferryTime$+sailing time between the nodes + T_{ijv}^B
            end
        end
    end
    if $PPT_i - (travelTime + ferryTime) < 0$ **then**
        return False
    end
end
return True

adding the sailing and service times at each node in-between the evaluated node and the reference node. If the node before the delivery node $n+i$ is a pickup node, this corresponds to the evaluation of delivery node 5 with pickup node 2 as a reference. As long as the calculated ride time RT_i is not higher than the max ride time MRT_i for each request i, the insertion is feasible.

Capacity feasibility

The ferry capacity feasibility check is noticeably easier to compute than for the previous constraints described. As can be seen in Algorithm 3, the procedure is performed in linear worst-case time due to being independent of time. The current number of passengers Q_v is used as a reference for the initial load of ferry v at the time of the check. Feasibility is determined by progressing chronologically through the copied route and correspondingly adding or subtracting the number of passengers P_i from the calculated capacity, depending on if the next node is a pickup or a delivery node. As the number of passengers on board cannot exceed the ferry capacity K_v, the procedure deems insertion infeasibility only if this is the case. Otherwise, the insertion is feasible in terms of ferry capacity.

Algorithm 3: Capacity feasibility

Initialize a copy of the ferry route R_v and insert the pickup node and delivery node for request i.

```
capacity ← Q_v
for all nodes i in the copied route do
    if i is a pickup node then
        capacity ← capacity + P_i
        if capacity > K_v then
            return False
        end
    else
        capacity ← capacity − P_{i−n}
    end
end
return True
```

Algorithm 2: Max ride time feasibility

Initialize a copy of the ferry route R_v and insert the pickup node and delivery node for request i. Update PPT_i for the corresponding pickup node i to a pickup time within $[\underline{T}_i^P, \overline{T}_i^P]$. Calculate the updated delivery time window $[\underline{T}_i^D, \overline{T}_i^D]$.

for *all pickup nodes i in the copied route* **do**
  Find position of $n+i$ in the route
  $rideTime \leftarrow CT_i$
  if *$n+i$ is the next node* **then**
    $rideTime \leftarrow rideTime$+sailing time from L_v to $n+i$
  else
    $nodeBeforeDnode \leftarrow$ node before $n+i$
    $travelTime \leftarrow$ sailing time from $nodeBeforeDnode$ to $n+i$
    if *$nodeBeforeDnode$ is a pickup node* **then**
      $rideTime \leftarrow$ (PPT for $nodeBeforeDnode + T^B_{ijv} + travelTime) - (PPT_i + T^B_{ijv})$
    else
      Find position of $lastPnode$ before $n+i$
      $rideTime \leftarrow$ PPT for $lastPnode + T^B_{ijv} + travelTime$
      for *all nodes between $lastPnode$ and $n+i$* **do**
        $rideTime \leftarrow rideTime$+sailing time between the nodes + T^B_{ijv}
      end
      $rideTime \leftarrow rideTime - (PPT_i + T^B_{ijv})$
    end
  end
  if $rideTime > MRT_i$ **then**
    return False
  end
end
return True

6.3 Objective Function

The insertion heuristic picks the feasible insertion with the lowest incremental cost, determined by the value of a given objective function. In practice, the objective function can comprise of indicators from the perspective of the operator's and the passengers' point of view. It can be interesting to evaluate how the preference in choosing feasible insertion affect the overall service level, when other characteristics inclined towards the same perspective are present in form of hard constraints. Therefore, two objective functions that reflects the perspective of the operator and passengers are used in the computational study conducted in Chapter 7, respectively. The objective function (6.1) minimizes the excess ride time for each request, while the objective function (6.2) minimizes the total ferry distance traveled. Here, $\mathcal{F}$ denotes the set of feasible insertions.

$$\min_{f \in \mathcal{F}} \left[\sum_{v \in V} \sum_{i \in N^O \cup N^A} P_i (t_{n+i,v} - t_{iv} - T^S_{i,n+i,v}) \right] \tag{6.1}$$

$$\min_{f \in \mathcal{F}} \left[\sum_{v \in V} D_v \right] \tag{6.2}$$

Chapter 7

Computational Study

In this chapter, configurations of the ferry service are studied through the simulation model described in Chapter 5. The proposed simulation model is implemented in Python with the process-based discrete-event simulation framework Simpy. By simulating various settings of the service with different demand scenarios, the performance and efficiency of the service can be observed through KPIs with regards to balancing operational cost with service levels. It is emphasized that the aim of this computational study is not to justify individual parameter values for the instances used, but instead, study how certain parameters are likely to affect the service with regards to balancing operational costs with a high service level. Each test instance was simulated over a planning horizon reflecting 500 hours of a given demand scenario to ensure that steady-state output could be reported.

Firstly, a description of the generation of test instances and data used in this study is provided in Section 7.1. Section 7.2 discusses and evaluates how the main service-related parameters associated with a high level of service, affect the service. Section 7.3 compares the results from using an objective function from the perspective of the operator and the customer, respectively. Lastly, Section 7.4 conducts a sensitivity analysis of the fleet size to further provide insight in the recommendation of the fleet configuration.

7.1 Generation of Test Instances

This section describes how the test instances for the computational study are generated. Since the environment in which the service of autonomous ferries operates within is theoretical, several considerations are made to the modeling and generation of test instances. In particular, these considerations are mainly given by the uncertainty in how the new ferry service affects demand. For this reason, three demand scenarios have been defined to evaluate the impacts in the performance of different designs of the service reflected in low, normal, and high demand. The demand scenarios are given as the mean arrival rate of requests per hour, as listed in Table 7.1. As the study of the DDARP-AF aims to provide a reliable service for day-to-day commuting, the computational study is conducted as if no other public transportation offerings exist, and demand needs to be fulfilled by the proposed service. As such, the study emphasizes on being able to meet demand in the peak period. However, the provided discussions of the test instances are seen in lights of the performances in all demand scenarios, as it is assumed that the demand during peak periods mainly occurs before and after work hours. In contrast, the other demand scenarios reflect the off-peak periods, with low demand practically indicating evening hours.

Table 7.1: Demand scenarios.

Scenario	Notation	Mean arrival rate
Low	λ_L	15 requests per hour
Normal	λ_N	20 requests per hour
Peak	λ_P	30 requests per hour

7.1.1 Generation of Customer Demand

As described in Chapter 5, the list of requests is generated a priori to the simulation execution. The procedure of generating the set of incoming requests is described in Algorithm 4. The call-in times for the requests are generated according to a Poisson process, where the interarrival times follow an exponential distribution. Note that a specific desired pickup time is generated for the request, even though the customer provides a pickup time window in practice. This is simply an implementation measure to prevent another corresponding set of requests being generated when evaluating the same demand scenario with respect to different time window widths. Consequently, the time window attribute is defined as $[\underline{T}_i^P, \underline{T}_i^P + TW_i]$ in the Request data structure.

The number of passengers associated with each request is uniformly sampled from a distribution skewed towards a smaller number of passengers. Figure 7.1 depicts the probability mass function of this distribution, and it can be observed that there is a 50 % probability that the incoming request regards a trip for a single person. The expected value is 2.45, which implies that for the *peak* demand scenario with a mean arrival rate $\lambda_P = 30$, it can be expected that 73.5 passengers on average want to travel with the ferry service every hour.

Algorithm 4: Generation of requests

Input : n, λ, $portDistribution$, $passengerDistribution$

Output : $listRequests$

$listRequests \leftarrow empty$

$arrivalTime \leftarrow 0$

for i : 1 ***to*** n **do**

> $randPortPair \leftarrow U(0, |portDistribution|)$
>
> $randNumPassengers \leftarrow U(0, |passengerDistribution|)$
>
> $portPair \leftarrow portDistribution[randPortPair]$
>
> $numPassengers \leftarrow passengerDistribution[randNumPassengers]$
>
> $interarrivalTime \leftarrow \exp(\lambda)$
>
> $arrivalTime \leftarrow arrivalTime + interarrivalTime$
>
> $pickupTime \leftarrow arrivalTime + U(\underline{CAT}, \overline{CAT})$
>
> $request \leftarrow Request(portPair, numPassengers, pickupTime, arrivalTime)$
>
> $listRequest \leftarrow listRequest \cup request$

end

return $listRequests$

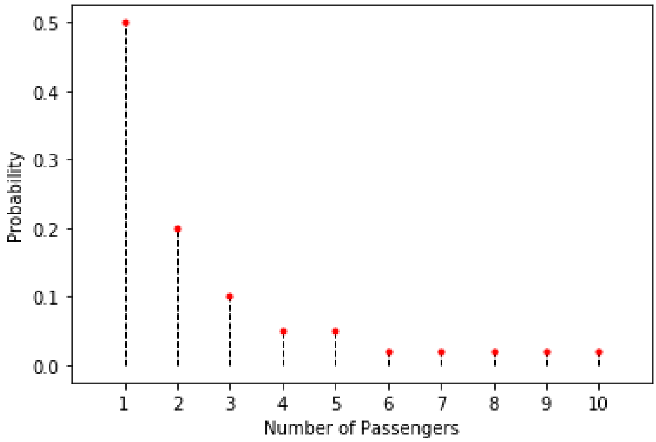


Figure 7.1: Distribution of the number of passengers associated with a request. The expected value is 2.45.

Port pair demand

The demand between ports is determined to reflect that some ports are more likely to experience customer traffic. The method for defining demand for each port pair is inspired by the approach presented in Aslaksen and Svanberg (2019). Here, the port attributes form a basis to determine the demand between a port pair. As presented in Table 7.2, ports are characterized in terms of size, and two nearby ports are defined as neighbors. Seasonal ports are excluded in the generation of customer demand, as it is assumed that the demand is highly seasonal.

Furthermore, the port size specifies factor values for each origin port and destination port, specifically. For the origin ports, small is set to factor 0.75, medium is set to 1.0, and large is set to 1.25. Correspondingly, 0.5, 1.0, and 1.5 are set for the small, medium, and large destination ports. An additional factor of 0.1 is associated with neighbor pair ports. Moreover, the corresponding factors for a given port pair are then multiplied to provide a weight. Lastly, each weight value is divided by the total sum of weights to compute the probability of the port pair. In Algorithm 4, the port pair is uniformly sampled from this new distribution. The numerical demand probabilities computed for each port pair are provided in Appendix A.

Table 7.2: Description of the port attributes used to determine demand between a port pair.

Port id.	Port name	Size	Neighbor
0	Laboe	Large	-
1	Möltenort	Medium	-
2	Mönkeberg	Medium	-
3	Dietrichsdorf	Large	4
4	Wellingdorf	Large	3
5	Bahnhof	Large	6, 7
6	Seegarten	Small	5, 7, 8
7	Reventlou	Large	5, 6, 8
8	Bellevue	Small	6, 7
9	Friedrichsort	Medium	-

7.1.2 Test Instances

The test instances are defined by the combination of the key characteristics listed in Table 7.3. Each key characteristics is given an instance name for ease of reference. As described in Section 6.3 two objective functions are considered in this study. The objective function OF1 refers to that of the passengers' point of view, minimizing excess ride times according to the objective function (6.1). Correspondingly, OF2 refers to the objective function (6.2). The key setting of time window width refers to the parameter TW_i, and is set for all requests i to $TW_i = 20$ and $TW_i = 30$ for TW1 and TW2, respectively. Similarly, the max ride time coefficient is set for all requests i to $T_i^{MAX} = 1.5$ for MRT1 and $T_i^{MAX} = 2$ for MRT2.

The fleet configuration is characterized by the fleet size, i.e., the number of ferries V, and the capacity K_v of each ferry. This study considers two specific fleets, representing a small (F1) and a large (F2) fleet, as given by Table 7.4. Generally, it is expected that the larger fleet will provide a higher service level as the operator has more flexibility in deployment, but at a higher cost. As such, the smaller fleet is provided with a higher

Table 7.3: Key settings and naming convention for test instances.

Key setting	Mode	Instance name
Objective function	Passenger	OF1
Objective function	Operator	OF2
Fleet	Small	F1
Fleet	Large	F2
Time window width	Small	TW1
Time window width	Normal	TW2
Max ride time coefficient	Small	MRT1
Max ride time coefficient	Normal	MRT2

total and individual ferry capacity to offset this assumption. Furthermore, other ferry attributes are assumed fixed, such as the sailing speed of each ferry. The sailing speed is based on the average sailing speed calculated from the current timetable of the ferry service provided by SFK, and the corresponding transit times are given as a matrix in Appendix B.

Table 7.4: Attributes of the fleets considered in the computational study.

Fleet	Fleet size	Ferry capacity	Total fleet capacity
Small	10	30	300
Large	15	15	225

The parameters which are independent of the different test instances are fixed for all instances according to the numerical values listed in Table 7.5.

Table 7.5: Fixed parameter values for all test instances.

Parameter	Notation	Value	Unit
Call ahead interval	$[\underline{CAT}, \overline{CAT}]$	[30, 180]	min
Berthing time	T^B_{ijv}	3	min
Max passengers per request	P^{MAX}	10	

The instance OF1-F1-TW2-MRT2 will serve as the base setting for comparing the test instances for the corresponding demand scenarios. As can be seen in the computed performance of the base setting presented in Table 7.6, this combination performs poorly at peak demand but can accommodate most requests at low demand. The percentage of accepted requests is referred to as the demand met. When evaluating the design with regards to a high level of service, the demand met is defined to be at least 70 % as a bare minimum threshold.

Table 7.6: Summary of KPIs for the base setting at different demand scenarios.

KPI	Low	Normal	Peak
Rejected requests (%)	8.97	25.08	49.31
Rejected passengers (%)	8.81	24.43	49.06
Avg. excess ride time (min.)	1.45	2.56	3.69
Avg. distance per ferry (km.)	5798	5888	5478
Avg. distance per request (km.)	8.49	7.86	7.20
Avg. idle time per ferry (min.)	3386	1670	727

7.2 Effects of Service-Related Parameters

In this section, OF1 is used as the setting for the objective function to study the effects of the service-related parameters. Generally, for the low demand scenario, all combinations of the time window widths and max ride time coefficients are found to provide an acceptable level of demand met given either fleet. However, at low demand, the large fleet is able to fulfill all requests with few exceptions. As such, the idle times are almost twice as much compared to the small fleet, with up to 90 % demand met. For the normal demand scenario, the same case can be observed with regards to the larger fleet. In contrast, the performance of the small fleet is able to provide a sufficient level of demand met. The performance at low demand for each instance with the small fleet is summarized in Table 7.7. It can be observed that the base setting overall provides better performance with regards to cost-efficiency and service quality.

Table 7.7: The performance of different combinations of service-related parameters with the small fleet at normal demand. The relative performance compared to the base setting is given in the parentheses.

Fleet: F1

Demand: Normal

KPI	TW1-MRT1	TW1-MRT2	TW2-MRT1	TW2-MRT2
Rejected requests (%)	29.20 (+16%)	25.55 (+2%)	28.15 (+12%)	25.08
Rejected passengers (%)	28.69 (+17%)	25.20 (+3%)	27.35 (+12%)	24.43
Avg. excess ride time (min.)	1.01 (-60%)	2.98 (+16%)	0.84 (-67%)	2.57
Avg. distance per ferry (km.)	5562 (-5%)	5627 (-4%)	5878 (-0%)	5888
Avg. distance per request (km.)	7.86 (+0%)	7.56 (-4%)	8.18 (+4%)	7.86
Avg. idle time per ferry (min.)	2952 (+77%)	2545 (52%)	1903 (14%)	1670

At peak demand, the instances with the large fleet give similar results as for the small fleet at the normal demand scenario. The performance of these instances is presented in Table 7.8. As expected, due to the increase in the number of ferries, the large fleet vastly outperforms the base setting. Some interesting features can be observed for both the considered demand scenarios with given fleets. For instance, the combinations TW1-MRT2 and TW2-MRT2 tend to provide the least rejected requests at the same level, but the latter combination produces significantly lower average idle time per ferry. Intuitively, a high level of service combined with low idle times would indicate an efficient use of the fleet. However, in theory, the feasible solutions found in the TW1-MRT2 instance should be feasible in TW2-MRT2, given that the same insertions are performed accordingly throughout the simulation. The reason for this is that the same demand scenario is regarded, and the difference lies in that TW1 yields a tighter time window constraint for the same feasible region. Therefore, the increase in idle time with the same level of service provided indicates a more efficient use of the fleet in this case. The simulation results show that the implemented insertion heuristic gravitates towards more efficient deployment with this setting, although a wider time window in practice should provide more flexibility in this matter.

Table 7.8: The performance of different combinations of service-related parameters with the large fleet at peak demand. The relative performance compared to the base setting is given in the parentheses.

Fleet: F2

Demand: Peak

KPI	TW1-MRT1	TW1-MRT2	TW2-MRT1	TW2-MRT2
Rejected requests (%)	31.63 (-36%)	27.63 (-44%)	30.51 (-38%)	27.69 (-44%)
Rejected passengers (%)	31.36 (-36%)	27.68 (-44%)	30.07 (-39%)	27.97 (-43%)
Avg. excess ride time (min.)	1.04 (-72%)	3.06 (-17%)	0.96 (-74%)	2.68 (27%)
Avg. distance per ferry (km.)	5305 (-3%)	5269 (-4%)	5499 (+0%)	5489 (+0%)
Avg. distance per request (km.)	7.76 (+8%)	7.28 (+1%)	7.91 (+10%)	7.59 (+5%)
Avg. idle time per ferry (min.)	1801 (+147%)	1617 (+122%)	1099 (+51%)	923 (+27%)

Another observation includes that the MRT2 setting is the main contributor to the increase in excess ride time, regardless of the combined parameter. It should be noted that the doubled increase in ride times due to this setting still provides low values, as one could argue that three minutes in excess ride times are still acceptable. Also, in providing the least percentage rejected requests, allowing for higher values of maximum excess ride times seem to be more efficient than providing the operator with wider time windows. Of the four combinations of service-related parameters, the TW1-MRT1 setting provides the least flexibility for the operator, and the expected relative performance of the service with this setting is observed accordingly. Furthermore, the average distance traveled per ferry is generally on the same level for all combinations varying between instances and demand scenarios.

7.3 Effects of the Objective Function

This section considers the test instances with the setting OF2 for the objective function. The impacts of the changed objective function seem to affect the performances of the test instances similarly for different fleet and demand scenarios. Generally, the same combinations of service-related parameters are still favorable. However, the change in objective function in favor of minimizing total ferry distance traveled provides much greater efficiency in deployment in all test instances. The relative change in performance for the corresponding test instances considered, can be observed in Table 7.7 and Table 7.8. The change for corresponding test instances are more prevalent for the test instances with the large fleet and peak demand, but seems to affect test instances with different fleet and demand scenarios in an equal manner.

Table 7.9: The performance of different combinations of service-related parameters with the small fleet at normal demand with OF2. The relative performance compared to the corresponding OF1 instance is given in the parentheses.

Fleet: F1
Demand: Normal

KPI	TW1-MRT1	TW1-MRT2	TW2-MRT1	TW2-MRT2
Rejected requests (%)	24.20 (-17%)	18.75 (-27%)	21.86 (-22%)	16.46 (-34%)
Rejected passengers (%)	23.40 (-18%)	18.04 (-28%)	21.01 (-23%)	15.47 (-37%)
Avg. excess ride time (min.)	3.02 (+199%)	8.23 (+176%)	3.11 (+270%)	8.19 (+219%)
Avg. distance per ferry (km.)	5335 (-4%)	5186 (-8%)	5627 (-4%)	5378 (-9%)
Avg. distance per request (km.)	7.04 (-10%)	6.38 (-16%)	7.20 (-12%)	6.44 (-18%)
Avg. idle time per ferry (min.)	3337 (+13%)	3441 (+35%)	2307 (+21%)	2720 (+63%)

The TW2-MRT2 combination receives the biggest impact in the reduction of percentage rejected requests, and is the setting that yields the highest demand met for all scenarios. With the changed objective, it can be observed that this setting receives a drastic increase in average idle time per ferry, but still provides the least average idle time compared to the other combinations. However, this comes at the cost of much higher passenger inconvenience in terms of increased excess ride times. An interesting note is that the test instances with the TW2-MRT1 have the highest increase in excess ride times, even though the max ride time coefficient is set with the lowest value. Overall, the average total distance is reduced as expected, but the effect is limited. The relative increase in demand met compared to the reduced average distance traveled suggests that a balanced design provides a more overall beneficial service in terms of both the perspective of the operator and customer. For instance, the operator could prefer the setting TW2-MRT2 to provide a service that can meet more demand. Alternatively, the change in objective function makes the setting TW2-MRT1 outperform all the combinations with OF1, reducing rejected requests while reducing average travel distance traveled per ferry with the same acceptable excess ride times of about three minutes.

Table 7.10: The performance of different combinations of service-related parameters with the large fleet at peak demand with OF2. The relative performance compared to the corresponding OF1 instance is given in the parentheses.

Fleet: F2

Demand: Peak

KPI	TW1-MRT1	TW1-MRT2	TW2-MRT1	TW2-MRT2
Rejected requests (%)	24.32 (-23%)	19.33 (-30%)	22.39 (-27%)	16.87 (-39%)
Rejected passengers (%)	24.14 (-23%)	19.43 (-30%)	22.06 (-27%)	16.94 (-39%)
Avg. excess ride time (min.)	3.42 (+229%)	8.87 (+191%)	3.57 (+272%)	8.93 (+233%)
Avg. distance per ferry (km.)	5034 (-5%)	4860 (-8%)	5245 (-5%)	5025 (-8%)
Avg. distance per request (km.)	6.65 (-14%)	6.03 (-17%)	6.76 (-15%)	6.04 (-20%)
Avg. idle time per ferry (min.)	2174 (+21%)	2373 (+29%)	1407 (+28%)	1683 (+83%)

7.4 Sensitivity of the Fleet Size

The operator faces a trade-off between ferry capacity and fleet size, as these main attributes determine the overall service capacity. It is assumed that the cost of increasing the fleet size is significantly higher than for increasing the capacity of a ferry. Generally, a smaller fleet is a more desirable option for the operator, provided that a high level of service can still be maintained. Therefore, given a specific setting for the design of the ferry service, it is in the interest of the service provider to assess the impact in the performance of varying the fleet size. As discussed in the previous sections, the large fleet was identified as necessary to provide a sufficient level of service in the peak demand scenario. The test instance OF2-F2-TW2-MRT2 is chosen as the basis for the conducted sensitivity analysis, due to being able to provide a performance with a low percentage of rejected requests at high demand. The total fleet capacity is kept constant to investigate the sensitivity of the fleet size by reducing or increasing the fleet size or the ferry capacity accordingly.

As can be seen in Figure 7.2, the demand met seem to be almost linearly proportional to the fleet size. This relation makes the increase in fleet size a decisive influence on the ability to serve a more substantial proportion of requests. This makes sense as the characteristics of time windows and ride time requirements generally make larger fleets better suited for flexible on-demand services. The minimum acceptable level of service can be achieved with 13 ferries, and some diminishing returns can be observed by introducing more ferries. This is further expressed in the increased rate of average idle times. As the simulations represent a planning horizon of 500 hours, it can be added that the observed average time each ferry spent being idle accounts for only 4.5-8.5 % of the total time. The modest increase in idle times can be understood as the service's inability to efficiently exploit the increased flexibility provided by a larger fleet, resulting in the diminishing returns.

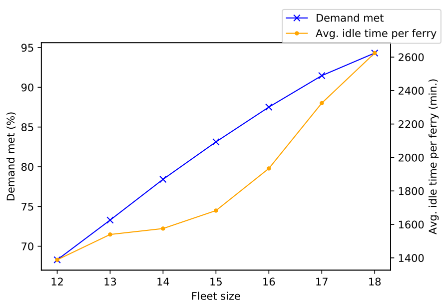


Figure 7.2: Sensitivity of the fleet size considering demand met and average idle time.

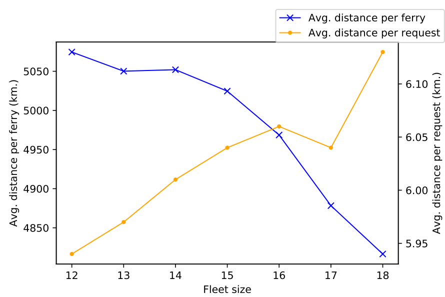


Figure 7.3: Sensitivity of the fleet size considering average distance per ferry and average distance per request.

In Figure 7.3, it can be observed that the average distance traveled per ferry generally decreases with the number of ferries. However, this should be seen in relation to the increased average idle times, as a positive rate in average idle time when increasing the fleet size indicates a less efficient deployment. Thus, it is expected that the average distance traveled per ferry decreases with the increase in the number of ferries. In fact, the distance traveled per request seems to increase with the fleet size. The increase is likely due to the planning procedure not being able to exploit the increased flexibility efficiently. This observation points out that more efficient routing could potentially provide economies of scale in terms of increased demand met and decreased average distance traveled per request. Also, from a practical view, the observed changes in the average distance traveled per request is arguably negligible in deciding the fleet size. Since varying the fleet size yields minor changes to this performance indicator, altering other service characteristics are more efficient in reducing the distance traveled, e.g., the presented change in the objective function.

Figure 7.4 shows no significant deviation between the percentage rejected requests and percentage rejected passengers. This correlation is expected due to the passenger distribution in which the set of requests are generated. However, for the depicted fleet configurations, this implies that the ride-sharing capabilities are not the limiting cause for accepting requests. In other words, the ferry capacity is generally not binding when finding feasible insertions. That implies that the requirements in terms of fulfilling the time windows and maximum ride time puts an upper bound on how many requests can practically be served by the same ferry simultaneously. However, since the customers are allowed to book trips for groups of ten people, the fleet cannot consist of ferries with lower capacities than this number. The maximum number of passengers associated with a request should serve as a lower bound to the ferry capacities to prevent the need for split passenger loads. Furthermore, as for the average distance traveled per request, the average excess ride times are practically not affected by the adjustments in fleet size. Consequently, an acceptable level of excess ride times should be determined through the study of other service-related parameters, before determining the fleet size with regards to the final intended demand met.

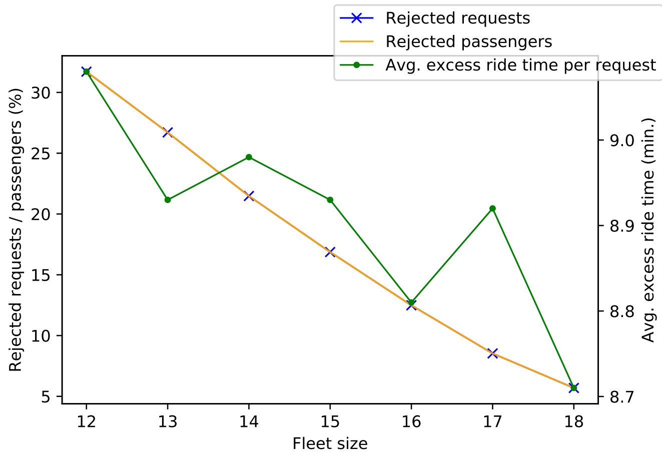


Figure 7.4: Sensitivity of the fleet size considering percentage rejected requests/passengers and the average excess ride time.

Overall, other than the percentage of rejected requests/passengers, the sensitivity analysis of the fleet size shows that the performance indicators are not significantly affected by varying the fleet size. Depending on the service provider's preference in determining a reasonable level of service, the fleet size of at least 13 ferries with the presented settings can ensure that the appropriate passenger demand can be catered in the peak demand scenario. One of the main benefits of autonomous ferries is the removal of costs related to personnel. For instance, the primary cost associated with increased average idle time for autonomous ferries is a higher cost per passenger trip, reflected by the initial fixed investment cost. As such, the operator should balance the cost of acquiring the fleet with the intended provided service level. Furthermore, a high utilization rate could potentially lead to increased depreciation rates and the need for frequent maintenance due to strained use. At the presence of these events, the short-term performance and availability of the service are likely to be affected. Therefore, the value of redundancy should be considered in determining the fleet configuration.

Chapter 8

Concluding Remarks

This chapter concludes the thesis. Section 8.1 provides closure of the study related to investigating the DDARP-AF. Section 8.2 outlines possible approaches for future research based on the amassed knowledge of the topic and overall hindsight.

8.1 Conclusion

This thesis has presented and examined the Dynamic Dial-a-Ride Problem with Autonomous Ferries (DDARP-AF). The problem concerns the design of an on-demand ferry service, where incoming requests with potentially very short call ahead times are booked throughout the planning period. As such, the operational planning procedure needs to efficiently determine if the requests can feasibly be served in an online manner, and plan the routing and scheduling of the ferries accordingly. The thesis contributes to the project CAPTin Kiel, which seeks to explore how autonomous ferries can be utilized to provide a cost-efficient, sustainable and energy-efficient service with great flexibility. Therefore, the design of the on-demand ferry service must be evaluated in terms of providing a high level of service perceived by the passengers, as well as balancing the concern with the operating costs. Typically, from the passenger's point of view, minimizing excess ride times is considered. In contrast, from the perspective of the operator, minimizing average distance traveled per ferry is of interest due to induced transport externalities, i.e., congestion and emissions.

A simulation model is proposed to measure the performance of different service characteristics. The output from the simulation model is used to formulate KPIs, which defines metrics to compare the performance of the services. The simulation model solves the operational planning problem through the procedure of an insertion heuristic due to the ability to efficiently provide feasible solutions. Feasible insertions are determined by pickup time window constraints, maximum ride time constraints, and ferry capacity constraints.

The proposed simulation model was implemented to conduct a computational study of different key parameter settings defining the ferry service. As the effects of changing demand due to introducing the new ferry service are unknown, several considerations were given in the modeling of demand for the test instances. Specifically, the effects of service-related parameters given by the pickup time window width and the maximum ride time coefficient were investigated under three demand scenarios. The results indicate that some settings were more favorable in all scenarios. It was found that higher flexibility for the operator in terms of wider time window width could not efficiently be exploited. Generally, the TW2-MRT2 setting performed better in terms of demand met at the expense of higher average excess ride time, but still at an arguably acceptable level. The change in objective function overall increased the performance of the service up to 39 % in being able to serve requests, implying that excessive consideration towards minimizing excessive ride times limits the overall performance. The sensitivity analysis shows that the increase in fleet size does not significantly affect the KPIs, other than the demand met. Given the preference of the service provider, a fleet of at least 13 ferries is found to provide a sufficiently acceptable level of service in the peak demand scenario.

In conclusion, more efficient deployment through other routing strategies can likely be achieved. Regardless, these effects can be further studied similarly through the proposed simulation model. The study of the operational performance for various settings can provide decision support in determining the fleet configuration and overall design of the dial-a-ride service with autonomous ferries in Kiel.

8.2 Future Research Opportunities

The following section highlights the research opportunities that can provide further insight and decision support concerning the design of a demand-responsive ferry service with autonomous ferries in the Kiel Fjord. Firstly, improvements to the heuristic method for solving the operational planning problem are discussed in Section 8.2.1. Furthermore, Section 8.2.2 presents potential extensions to the simulation framework to better reflect real-world scenarios. Finally, Section 8.2.3 considers how the ferry service could benefit from integrating the on-demand system with a fixed-route system.

8.2.1 Improving the Heuristic Solution

Improvements to the heuristic solution method can be made to provide more efficient deployment of the given fleet potentially. Generally, the solutions produced by an insertion heuristic tend to be of poor quality as no effort is provided for improving the current schedules. As such, a reoptimization procedure can be included in the request handler of the simulation model, as illustrated in Figure 8.1. Several improvement procedures can be suggested, such as simple iterative reinsertions or comprise of more advanced metaheuristics. Regardless, it is important to note that the available computational time for reoptimization is limited in a practical aspect. The reason is that the procedure must be able to provide new feasible solutions between the time after applying the insertion heuristic and the arrival time of a new request. This can make it challenging to develop efficient and effective procedures. Alternatively, the insertion heuristic can be combined with demand-anticipatory capabilities such as presented in van Engelen et al. (2018). The look-ahead could potentially counteract the myopic view of insertion heuristics by influencing the preferred choice of feasible insertions in long-term planning. At the same time, simpler reoptimization approaches can be used for short-term replanning. As the solutions provided by heuristic methods are likely in a local optimum, the study of these should be similarly conducted as presented in this thesis, to assure the overall performance of the ferry service.

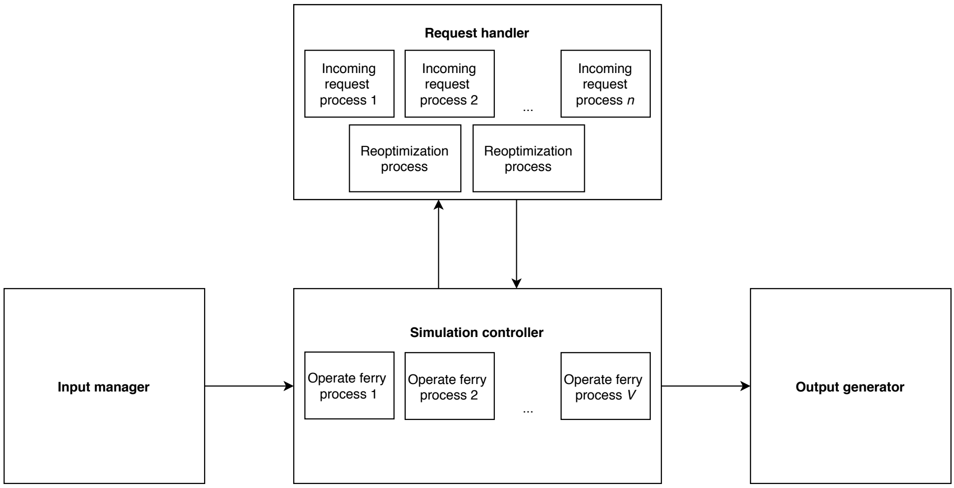


Figure 8.1: A reoptimization procedure can be included as a as part of the request handler.

8.2.2 Extensions to the Simulation Framework

Several extensions can be made to the simulation model that allows for more advanced policies to be studied. Such features could, for instance, include revised planned pickup times and postponed response to allow for bundling of requests. Moreover, as the simulation model intends to evaluate different service characteristics, real-world uncertainty would likely affect the performance of the service. Stochastic events such as arrival times uncertainty, passenger no-shows, and cancellation may lead to schedules of poor quality or infeasible solutions. Due to the dynamic and stochastic environment in real-life planning, an extended simulation-optimization framework could be developed to study the effects of real-world uncertainties further and provide robust schedules with regards to these.

8.2.3 The Integrated DARP-AF

Due to varying demand throughout the day, it would be interesting to combine the on-demand service with a fixed-route service. As previously discussed in this thesis, a fleet with substantial size would be needed to provide a sufficient level of service in the peak periods. Since it is assumed that the high demand mainly occurs around work-hours, the redundant fleet size would lead to ferries spending a high proportion of time being idle during the rest of the day. Therefore, an integrated dial-a-ride service with autonomous ferries could provide fixed schedule departures that could overtake certain percentages of the demand during peak hours. The fleet size for the on-demand service can then be catered accordingly to reflect the experienced demand level during most of the day. As illustrated in Figure 8.2, the integrated service combines the cost-efficiency of the fixed-route service and limits the necessary fleet size needed to provide flexibility in on-demand services. The benefits of this integrated solution could lower the cost per passenger trip while maintaining a high level of service. However, the complex planning at the strategic, tactical, and operational level for each service would need to be studied, but also considered in relation to each other.

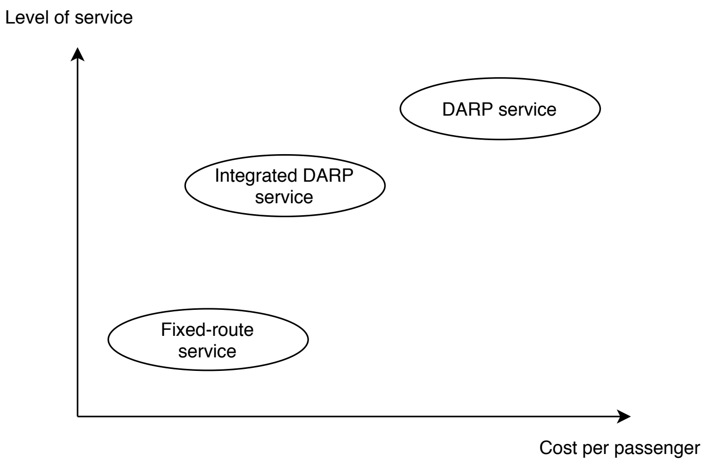


Figure 8.2: Potential benefit of the integrated DARP-AF.

Bibliography

Anderson, J. M., Kalra, N., Stanley, K. D., Sorensen, P., Samaras, C. & Oluwatola, T. A. (2014). *Autonomous vehicle technology: A guide for policymakers*. RAND Corporation.

Aslaksen, I. E. & Svanberg, E. B. (2019). *Network design of autonomous ferries in the kiel fjord*. Project report from NTNU.

Bailey, W. A. & Clark, T. D. (1987). A simulation analysis of demand and fleet size effects on taxicab service rates. In *Wsc '87*.

Beaudry, A., Laporte, G., Melo, T. & Nickel, S. (2010). Dynamic transportation of patients in hospitals. *OR Spectrum*, *32*, 77–107.

Braekers, K., Caris, A. & Janssens, G. (2014). Exact and meta-heuristic approach for a general heterogeneous dial-a-ride problem with multiple depots. *Transportation Research Part B: Methodological*, *67*, 166–186.

Bui, A. T. A. & Nguyen, J.-T. T. T. (2019). *On-demand transit service with autonomous ferries in the kiel fjord*. Project report from NTNU.

Christiansen, M. & Fagerholt, K. (2014). Chapter 13: Ship routing and scheduling in industrial and tramp shipping. (pp. 381–408).

Cordeau, J.-F. (2006). A branch-and-cut algorithm for the dial-a-ride problem. *Operations Research*, *54*, 573–586.

Cordeau, J.-F. & Laporte, G. (2003). The dial-a-ride problem (darp): Variants, modeling issues and algorithms. *Quarterly Journal of the Belgian, French and Italian Operations Research Societies*, *1*(2), 89–101.

Cordeau, J.-F. & Laporte, G. (2007). The dial-a-ride problem: Models and algorithms. *Annals of operations research*, *153.1*, 29–46.

Cross, J. & Meadow, G. (2017). Autonomous ships 101. *Journal of Ocean Technology, Vol 12*, 23–27.

Deflorio, F. P. (2011). Simulation of requests in demand responsive transport systems. *IET Intelligent Transport Systems*, *5*(3), 159–167.

Diana, M., Dessouky, M. & Xia, N. (2006). A model for the fleet sizing of demand responsive transportation service with time windows. *Transportation Research Part B: Methodological*, *40*, 651–666.

European Commission. (2017). *Energy, climate change, environment - climate action.*

Fu, L. (2002). A simulation model for evaluating advanced dial-a-ride paratransit systems. *Transportation Research Part A: Policy and Practice*, *36*, 291–307.

Gendreau, M., Laporte, G. & Semet, F. (2001). A dynamic model and parallel tabu search heuristic for real-time ambulance relocation. *Parallel Computing*, *27*(12), 1641–1653.

Gerrard, M. J. (1974). Comparison of taxi and dial-a-bus services. *Transportation Science*, *8*(2), 85–101.

Häll, C. H., Högberg, M. & Lundgren, J. T. (2012). A modeling system for simulation of dial-a-ride services. *Public transport*, *4*(1), 17–37.

Häll, C. H., Lundgren, J. & Voß, S. (2015). Evaluating the performance of a dial-a-ride service using simulation. *Public Transport*, *7*(2), 139–157.

Hanne, T., Melo, T. & Nickel, S. (2009). Bringing robustness to patient flow management through optimized patient transports in hospitals. *Interfaces*, *39*(3), 241–255.

Haugland, D. & Ho, S. (2010). Feasibility testing for dial-a-ride problems. (pp. 170–179).

Heathington, K. W., Miller, J., Know, R. R., Hoff, G. C. & Bruggeman, J. (1968). Computer simulation of a demand scheduled bus system offering door-to-door service. *Transportation Research Part B*, *251*, 26–40.

Ho, S., Szeto, W., Kuo, Y.-H., Leung, J. M., Petering, M. & Tou, T. W. (2018). A survey of dial-a-ride problems: Literature review and recent developments. *Elsevier: Transportation Research Part B: Methodological*, *111*, 395–421.

Hunsaker, B. & Savelsbergh, M. (2002). Efficient feasibility testing for dial-a-ride problems. *Transportation Research B*, *30*(3), 169–173.

Hyland, M. & Mahmassani, H. (2018). Dynamic autonomous vehicle fleet operations: Optimization-based strategies to assign avs to immediate traveler demand requests. *Transportation Research Part C Emerging Technologies*, *92*, 278–297.

Jaw, J.-J., Odoni, A. R., Psaraftis, H. N. & Wilson, N. H. M. (1986). A heuristic algorithm for the multi-vehicle advance request dial-a-ride problem with time windows. *Transportation Research B*, *20*(3), 351–357.

Kongsberg. (2019). Autonomous ship project, key facts about yara birkeland. Retrieved June 20, 2020, from https://www.kongsberg.com/maritime/support/themes/autonomous-ship-project-key-facts-about-yara-birkeland/

Kretschmann, L., Burmeister, H.-C. & Jahn, C. (2017). Analyzing the economic benefit of unmanned autonomous ships: An exploratory cost-comparison between an autonomous and a conventional bulk carrier. *Research in Transportation Business and Management*, *25*, 76–86.

Madsen, O. B. G., Ravn, H. F. & Rygaard, J. M. (1995). A heuristic algorithm for a dial-a-ride problem with time windows, multiple capacities, and multiple objectives. *Annals Of Operations Research*, *60*, 193–208.

Noda, I., Ohta, M., Shinoda, K., Kumada, Y. & Nakashima, H. (2003). Evaluation of usability of dial-a-ride systems by social simulation. (Vol. 2927, pp. 167–181).

Pankratz, K. & Müller-Lupp, W. (2020). Captin, a transdisciplinary innovation platform of the kiel university. Retrieved June 20, 2020, from https://www.captin.uni-kiel.de/en

Parragh, S., Doerner, K., Hartl, R. & Gandibleux, X. (2009). A heuristic two-phase solution method for the multi-objective dial-a-ride problem. *Networks*, *54*, 227–242.

Pillac, V., Gendreau, M., Guéret, C. & Medaglia, A. L. (2013). A review of dynamic vehicle routing problems. *European Journal of Operational Research*, *225*(1), 1–11.

Pitera, K. & Marinelli, G. (2017). *Autonomous e-mobility as a service - final report.* NTNU.

Quadrifoglio, L., Dessouky, M. M. & Ordóñez, F. (2008). A simulation study of demand responsive transit system design. *Transportation Research Part A*, *42*(4), 718–737.

Rolls-Royce. (2018). Rolls-royce and finferries demonstrate world's first fully autonomous ferry. Retrieved June 20, 2020, from https://www.rolls-royce.com/media/press-releases/2018/03-12-2018-rr-and-finferries-demonstrate-worlds-first-fully-autonomous-ferry.aspx

Ropke, S., Cordeau, J.-F. & Laporte, G. (2007). Models and branch-and-cut algorithms for pickup and delivery problems with time windows. *Networks*, *49*, 258–272.

SFK. (2020). Schlepp- und fährgesellschaft kiel (sfk). Retrieved April 27, 2020, from https://www.sfk-kiel.de/en/

Sørensen, U. (2017). *Shipping and digitalization*.

Tang, J., Kong, Y., Lau, H. & Ip, A. W. H. (2010). A note on "efficient feasibility testing for dial-a-ride problems". *Operations Research Letters*, *38*(5), 405–407.

Toth, P. & Vigo, D. (1996). Heuristic algorithms for the handicapped persons transportation problem. *Transportation Science*, *31*(1), 60–71.

Uchimura, K., Takahashi, H. & Saitoh, T. (2002). Demand responsive services in hierarchical public transportation system. *Vehicular Technology, IEEE Transactions on*, *51*, 760–766.

van Engelen, M., Cats, O., Post, H. & Aardald, K. (2018). Enhancing flexible transport services with demand-anticipatory insertion heuristics. *Elsevier: Transportation Research Part E: Logistics and Transportation Review*, *110*, 110–121.

Wilson, N. H. M., Sussman, J. M., Hiconnet, B. T. & Goodman, L. A. (1969). The use of simulation in the design of a dial-a-ride... of a computer aided routing system (cars). *Transportation Research Part B*, *318*, 66–76.

Appendix A

Demand Data

Table A.1: Probability that a request is generated with a specific (regular) port pair (%).

Port no.	0	1	2	3	4	5	6	7	8	9
0	0	1.27	1.27	1.91	1.91	1.91	0.64	1.91	0.64	1.27
1	1.53	0	1.02	1.53	1.53	1.53	0.51	1.53	0.51	1.02
2	1.53	1.02	0	1.53	1.53	1.53	0.51	1.53	0.51	1.02
3	1.91	1.27	1.27	0	0.19	1.91	0.64	1.91	0.64	1.27
4	1.91	1.27	1.27	0.19	0	1.91	0.64	1.91	0.64	1.27
5	1.91	1.27	1.27	1.91	1.91	0	0.06	0.19	0.64	1.27
6	1.14	0.76	0.76	1.14	1.14	0.11	0	0.11	0.04	0.76
7	1.91	1.27	1.27	1.91	1.91	0.19	0.06	0	0.06	1.27
8	1.14	0.76	0.76	1.14	1.14	1.14	0.04	0.11	0	0.76
9	1.53	1.02	1.02	1.53	1.53	1.53	0.51	1.53	0.51	0

Appendix B

Transit Times Data

Table B.1: Sailing times between regular ports for a ferry. Longest possible link is 32 minutes.

Port no.	0	1	2	3	4	5	6	7	8	9
0	0	10	18	30	32	30	27	25	20	10
1	10	0	10	20	22	25	22	20	13	5
2	18	10	0	10	12	15	13	9	5	13
3	30	20	10	0	2	15	10	7	10	25
4	32	22	12	2	0	17	12	9	12	27
5	30	25	15	15	17	0	5	8	14	30
6	27	22	13	10	12	5	0	5	10	26
7	25	20	9	7	9	8	5	0	6	20
8	20	13	5	10	12	14	10	6	0	15
9	10	5	13	25	27	30	26	20	15	0

Appendix C

Simulation Results for Low Demand

Table C.1: The performance of different combinations of service-related parameters with the small fleet at normal demand with OF1. TW2-MRT2 provides the most balanced performance. Note that the higher average distance per ferry is due to higher utilization. All instances provide relative low average excess ride times.

Fleet: F1
Demand: Low

KPI	TW1-MRT1	TW1-MRT2	TW2-MRT1	TW2-MRT2
Rejected requests (%)	14.01	10.57	11.65	8.97
Rejected passengers (%)	13.67	10.05	11.16	8.81
Avg. excess ride time (min.)	0.57	1.83	0.46	1.45
Avg. distance per ferry (km.)	5491	5527	5756	5798
Avg. distance per request (km.)	8.52	8.24	8.69	8.49
Avg. idle time per ferry (min.)	4614	4325	3660	3386

NTNU

Norwegian University of
Science and Technology