Norwegian School of Economics

Bergen, Spring 2021

Cargo Scheduling and Ship Routing

Simultaneous Evaluation of Fuel Costs, Opportunity Costs and Port Charges

Eirik Veland Gjeldnes and Sigve Furholt

Supervisor: Julio César Góez

Master thesis, Economics and Business Administration, Business Analytics.

NORWEGIAN SCHOOL OF ECONOMICS

Abstract

In this thesis we propose an approach to find cost minimizing schedules on how contracts of affreightment can be carried out.

For cost control our approach covers fuel costs and a set of vessel specific costs which are not related to fuel. Consumption of fuel is expressed as a function of distance sailed, sailing speed and the consumption properties of each specific vessel. For non-fuel related costs, we include opportunity costs and some port charges. Each vessel has an opportunity cost associated with its use and the fees charged at different ports may depend on the size and type of the vessel.

Our model is based on a network flow formulation of the *capacitated heterogeneous vehicle routing problem with time windows and with pickup and delivery*. Speed as a variable makes the model considerably more demanding in terms of solving time. For that reason, the problem is divided into two formulations. First: the model solves a main problem that finds the cost minimizing schedule using a fixed speed. Second: the model solves a speed optimization problem separately for each route used in the schedule. Compared to a model without this splitting, our model approach offers a reduced ability to find minimum cost schedules. However, it allows small instances of the problem to be solved within a reasonable amount of time while also including some way of adjusting speed.

Our aim is to model an approach that contributes to the development of a tool which can provide useful insights and improve schedulers' ability to find cost effective schedules. The model's solutions are not to be interpreted as complete schedules ready to be put into action unsupervised; The model is deterministic and no considerations of uncertainties are integrated. The purpose of the model's output is to serve as a cost-efficient starting point for schedulers to further adjust and tailor for their risk preferences.

Acknowledgements

We wish to thank Julio César Góez for good guidance during our work with the thesis. We also want to thank Dataloy for providing data to test our model and especially Julie Schasler and for sharing useful and much needed knowledge about the operations of their customers.

Table of contents

1. Introduction

1.1 Background

Transportation of goods by sea is an important part of the global economy. In 2018 the global volume of goods transported on sea was 10.7 billion tons (Ge, et al., 2019). This gives an increase of 7 billion tons of transported goods since 1980, where it was transported 3.7 billion tons of goods (United Nations, 2017). Simultaneously the capacity of the world fleet has increased. In 1980 the world fleet had a deadweight ton[1] capacity of 683 millions (Christiansen, Fagerholt, Nygreen, & Ronen, 2013), while in 2017 this number had increased to 1.86 billions dwt (United Nations, 2017).

As the market increases so does the potential gain of improving and streamlining any aspect of operations. Dataloy is a company which have developed a voyage management system to support shipping companies. This can aid their customers in contract management, chartering, operations, post voyage analysis as well as voyage planning and scheduling. Although Dataloy's system is facilitating an easier planning process for their customers, there are room for further development in specific parts of their system.

Typically, a customer at Dataloy has committed itself to multiple *contracts of affreightment*, referred to as COA or cargo. Such a contract commits the shipping company to carry a given quantity of a commodity from one location to another. Generally, the shipping company's chartering department deals with the assessment of which cargo contracts to enter. Often the shipping companies and the cargo owners make agreements of two-way cooperation for given periods to make the near future of business more predictable. Many of these contracts include time window for pickup and sometimes with a time window for delivery. The shipping company may lose a contract if they fail to pick up the cargo within time and fees may accrue should they fail to deliver within the delivery window. It is typically the scheduling department's responsibility to allocate cargo contracts to vessels and to find schedules such that operations are executed within the terms of the contracts.

[1] Deadweight tonnage is the sum of the weights of cargo, fuel, fresh water, ballast water, provisions, passengers, and crew.

Especially in the process of cargo allocation and vessel scheduling within a given planning period we believe Dataloy's customers would benefit from better tools. Specifically, a tool that accounts for multiple cost factors and can find and suggest low-cost schedulers. This includes that all cargos are assigned to a vessel and that these vessels have a specific route which ensures all cargoes are loaded and discharged on time. Such planning or scheduling problems can, as later elaborated in chapter 4, be a very difficult task, with many different variables that must be considered simultaneously. Dataloy provides their customers with tools to ease the scheduling process, but there are no model or algorithm that can evaluate a set of inputs to then propose an actual schedule. Currently these schedules are created manually, relying on the knowledge and experience of the schedulers. Although the shipping companies can use Dataloy's tools to compare and analyze their schedules, there are few or no metrics to evaluate if the schedule is as good as its potential.

1.2 Delimitations

The focus of this thesis is companies engaging in tramp shipping. For ship schedulers at such companies, it means routinely receiving new contracts which require schedules to be updated or replaced. The thesis only explores the part of tramp shipping that regards scheduling and not the part of assessing opportunities and entering contracts. This means we consider all given contracts of affreightment as binding and mandatory to carry out.

To limit the scope of the thesis, uncertainty is not considered when developing our approach. A further limitation of the scope regards to performance: We focus our efforts on the aspects to include in a problem formulation and are not deploying specific techniques to provide good performance in terms of solving time.

Throughout the thesis we frequent use some common terms that may benefit from clarification.

- We refer to a *visit-sequence* as a vessel's order of locations that is visited for loading- and unloading different cargos. Hence, a *visit-sequence* implies which cargo is loaded or unloaded at each location as well as the total cargo load onboard the vessel after each visit.
- We refer to a vessel's *timetable* as the time for when the vessel begin service at each location in its *visit-sequence*.

- A *route* is the combination of a *visit-sequence* and its corresponding *timetable*. When referring to a vessel's *route* or *visit-sequence* it regards the whole planning period for this vessel.
- By *schedule* we refer to the comprehensive set of *routes* that combined will be able to carry out all contracts.

1.3 Problem statement

In this thesis we intend to formulate an approach for finding cost-minimizing schedules for executing and completing a given set of contracts.

When finding and selecting a good schedule multiple aspects require consideration. Crucial aspects are the terms of the contracts, physical constraints, costs, risk, and uncertainty. Our aim is to provide low-cost schedules within the constraints of contracts and physical limitations. The purpose is to provide useful insights or to serve as a good starting point for schedulers before they make adjustments to account for uncertainty and to match their risk preference.

In its essence this problem is two-fold and consist of finding feasible schedules and then evaluating their performance in terms of a given objective. Our pursuit in this regard is to find an appropriate formulation for a cost-minimizing objective and to propose an approach for how this objective can be reached.

1.4 The structure of the thesis

In chapter 2 we conduct a literature review on what has been written about the different aspects of ship scheduling and routing.

In chapter 3 we outline a general perspective on cost drivers in maritime transportation of goods, while we in chapter 4 studies the decision problem regarding how scheduling is currently done by Dataloy's customers and the possible problems with this method. Chapter 5 describes our proposed approach as well as describing why certain factors should be included in the scheduling tool while others should not. In Chapter 6 we discuss how to use the model and interpret its solutions, using an example with a solution. Chapter 7 will summarize the thesis and propose a suggestion for future development of our findings.

2. Literature review

The transport of goods at sea is a complex endeavor and one of the main transportation modes of cargo. Each year there are carried around 10 billion tons of goods (Ge, et al., 2019). Transportation of goods at sea is also the most cost-effective option for transportation of large volumes. An improvement and optimization of the scheduling and speed might give a significant impact on the cost effectiveness, given the large volume of goods transported by sea (Norstad, Fagerholt, & Laporte, 2011).

2.1 Ship routing and scheduling

In the article “Decomposition of a combined inventory and time constrained ship routing problem” (Christiansen, 1999) there was done a study on a real ship planning problem, which combined an inventory management problem and a routing problem with time windows.

The problem in the article by Christiansen (1999) is a scheduling problem for ammonia transportation from Norsk Hydro ASA by ship. Planners within Hydro sits on information about the production and consumption pattern of ammonia and are responsible for the stock level at all Hydro factories. The planners therefore need to schedule routes for a fleet of heterogeneous ships from production- to consumption harbours. The location of the harbours and the production and inventory at the harbours, along with the ship capacities, dictates the possible arrivals at each harbour and the possible time windows for start of service.

Christiansen (1999) formulates the problem to be a deterministic cost minimization problem. The model is using an arc flow formulation which is a binary flow network that gives the total set of harbours and the set of available ships to be routed and scheduled. It also shows where a ship sails, if a ship sails or not, and if the ships is set to visit another harbour or if it is at the end of its route. The objective function in the article consists of the cost of visiting the harbours and the cost of channel passing dependent on the size of the ship. It also considers the channel costs dependent on the load status of the ship and some specific costs based on the channels that are used. The objective function also included fuel costs.

Within shipping it is generally three modes of operation: industrial, tramp and liners. Industrial shipping is described as a form of shipping where the cargo owner also controls the ship, while in tramp shipping there are ship owners and cargo owners, and then the ships go wherever there are available cargo. The liners operate according to a regular schedule and are sailing the same route back and forward throughout the year (Brønmo, Christiansen, & Nygren, 2007).

In tramp shipping there are often long-term contracts between the shipowners and cargo owners. But in many cases, there might be available capacity on the ship that the long-term contracts do not fill, in this is the case they can acquire more cargo in the spot market.

In the article “Tramp ship routing and scheduling with speed optimization” (Norstad, Fagerholt, & Laporte, 2011) they study a tramp ship routing and scheduling problem where they consider both mandatory and optional cargo, and the speed the ship sails with. They also operate with time windows which dictates when the cargo needs to be picked up and, in some cases, also delivered.

In the article from Norstad et al. (2011) they make an objective function that maximize profit by taking the total freight income minus the operating cost. Norsted et al. (2011) is, as Christiansen (1999), using a mathematical arc flow formulation for the ship routing and scheduling problem.

Norstad et al. (2011) also considers the speed optimization problem which gives their model another dimension. They formulate the speed optimization problem as a sequence of port calls with time windows that a given ship must sail, and where the objective will be to determine the speed for each leg in the route to minimize the fuel consumption. They also state that the cost function in this optimization problem is a non-linear function which describes the relationship between the consumption of fuel and sailing speed.

Another noteworthy point from this article regards to the properties of most fuel consumption functions. One of the methods they use, a recursive smoothing algorithm, make use of functions that are convex and non-negative. If a vessel's fuel consumption function has such properties, then total fuel consumed will be minimized by holding the speed constant and as low as possible. This means that exact knowledge about the fuel consumption function is not necessary as long as it is convex and non-negative within the interval of a vessel's minimum and maximum speed.

2.2 Split loads

In most of the literature about ship routing, a single cargo cannot be transported by more than one ship. However, a few articles have investigated what effect split loads could have on the revenue of a shipping company.

In the article "A large neighbourhood search heuristic for ship routing and scheduling with split loads" (Korsvik, Fagerholt, & Gilbert, 2011) the authors look at the possibility of splitting loads onto several ships. By doing a case study with 50 cargoes and a homogenous fleet of eight ships, they analyse the effect of split loads. The computational study done by Korsvik et al. (2011) concludes that when the cargo quantities are small the benefit of split loads are small. It also concludes that split loads could be of great values to the shipping companies since it gives them the possibility to better utilize their fleet capacity and transport more additional spot cargoes. The study also shows that the introduction of split loads will make the ship routing and scheduling problem harder to solve and it will also increase the computational time.

The article "Ship routing and scheduling with flexible cargo sizes" (Brønmo, Christiansen, & Nygren, 2007) they found that the introduction of flexible cargo sizes, as they call it, may lead to higher profit for the shipping company. This because they can utilize the capacity of the fleet and carry more spot cargo. In the computational study done by Brønmo et al. (2007) they conclude that through their tests they found a substantial short-term effect by introducing the split loads. However, they were not able to conclude on the potential long-term consequences an increase of total load could have. The uncertainty is mostly because many of the cargoes are part of a long-term contract that could be spanning over several planning horizons. They also concluded that the computational complexity of the problem would increase with the size of the cargo quantity intervals.

2.3 Computational performance

On the topic of simultaneous cargo-scheduling and vessel routing many articles formulate their approaches mathematically as mixed integer network flow problems. The formulations used are often intuitive, making it easier to comprehend how the constraints and objectives relate to the underlying real-world problems. This allows the authors to efficiently communicate their further developments on how to approach their version of the problem. However, these formulations are very computationally demanding and can only solve smaller instances of the problems. For this reason, much of the effort in the literature goes to how problems, as stated in the initial formulations, can be solved in a more reasonable amount of time.

Christiansen (1999) formulate and thoroughly explain a network flow model for solving a ship routing problem. She then goes on to explain further reformulation after stating that the initial formulation only can solve small instances of the problem. The initial formulation is reformulated to apply a column generation approach. This method splits the problem into a master- and a sub problem for each ship, where the solution process alternates between solving the master- and the sub problem. The sub problem only considers one ship a time and finds feasible routes while the master problem evaluates the combinations of all routes that are found. A more thorough discussion of a similar application of the approach is had in (Solomon, Kallehauge, Larsen, & Madsen, 2005).

In the article by Brønmo et al. (2007), they propose a solution to a ship routing problem with flexible cargo sizes. They do also use an initial formulation to present their approach before describing how the problem can be solved in a manner that requires less time. They suggest a set partitioning approach where a set of candidate schedules for each vessel is first generated. All schedules are then passed to a model that will find the combination of schedules that delivers all cargos and give the largest profit.

In Norstad et al. (2011) sailing speed is introduced as a variable, and they present how this can be implemented using an arc flow formulation. They then comment that any real-life instance of the problem will be too large to solve. Two different methods are then proposed to solve instances of a speed optimization problem for separate routes. A heuristic approach is then presented to provide a solution for the schedule as a whole.

3. Cost of shipping

3.1 Cost of fuel consumption

Fuel expenses is a major cost driver in shipping and the price of fuel certainly plays an important role, but multiple other factors are also important. The price of fuel can be subject to large fluctuations. This is beyond the control of the ship owners and adds an extra layer of uncertainty to the industry. However, many factors affecting total fuel consumption is within the control of the shipping company. Their ability to execute their operations by minimizing sail distance can have a large impact. Ship type, size, load level and engine will also play a role. Also sailing speed plays a major role in total fuel consumption. The idea of slower sailing speed or *slow steaming* has gained traction in the last decades to reduce cost (Meyer, Stahlbock, & Voss, 2012). The relationship between fuel consumption and speed is exponential within a given speed interval for most vessels (Keh-Kim, Boung-Yew, & King-Hwa, 2018). This means that the total fuel consumption of two similar voyages is higher for the one where sailing speed is higher.

3.1.1 Possible costs of emission

The direct price of fuel may not be the only cost that affect the total fuel expenses. Emission of greenhouse gases is, depending on the vessel’s properties, a function of fuel consumption. For the most part the shipping industry does not pay directly for its emissions today, but as the focus on the climate is increasing this may change. As shown in *Figure 1* (World Bank, 2020, p. 11) the share of global greenhouse gas emissions that are subject to a tax has increased rapidly in the last years. Lately it has been several pushes to also make the shipping industry subject to such taxation.

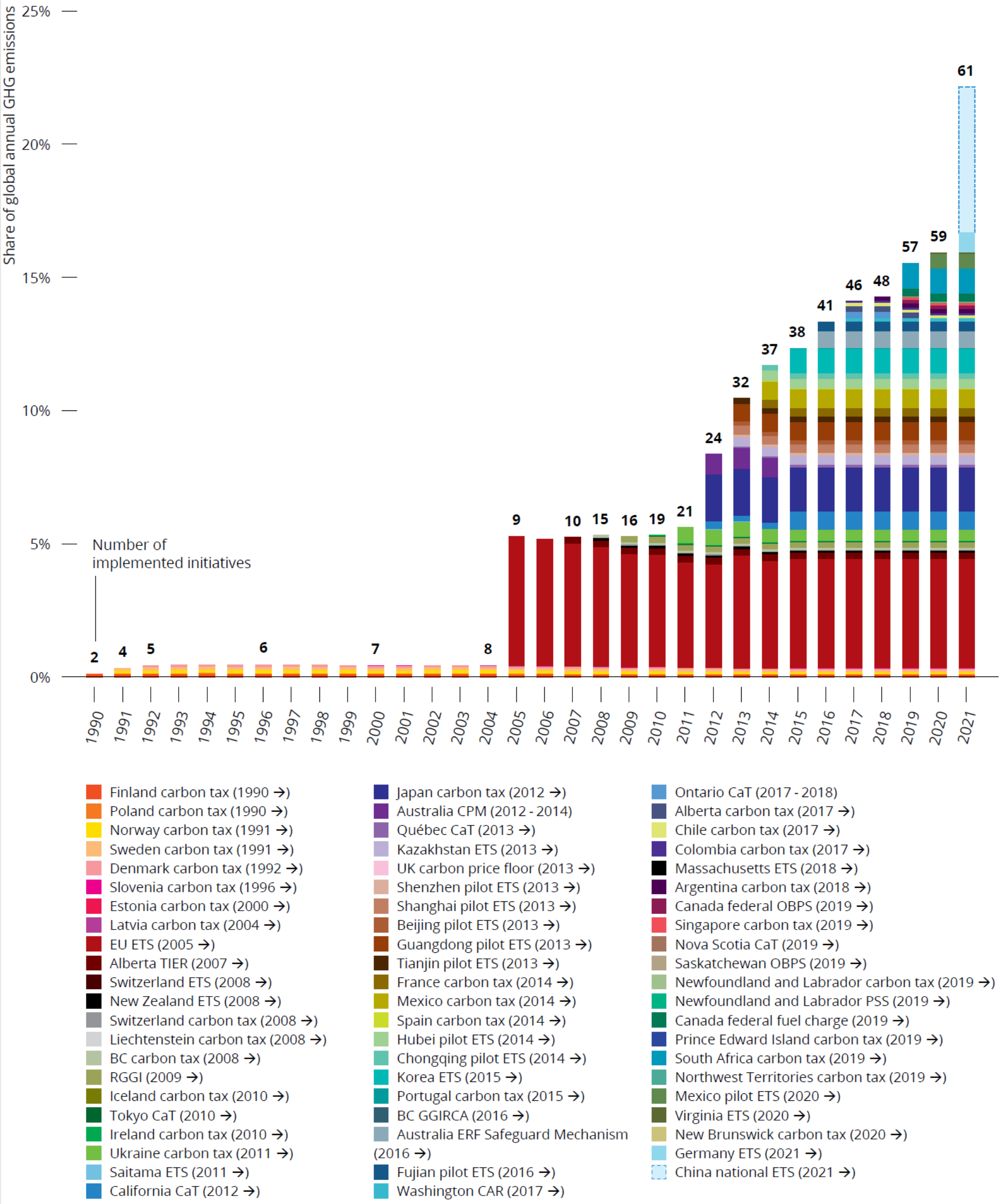


Figure 1 - Note. From "State and Trends of Carbon Pricing" by (World Bank, 2020, p. 11) https://openknowledge.worldbank.org/bitstream/handle/10986/33809/9781464815867.pdf?sequence=4&isAllowed=y)

Following are some examples on pushes to introduce cost on emissions for the shipping industry. One submission was put forward to the International Maritime Organization (IMO) - an agency under the UN, by an alliance of shipping industry associations; They proposed mandatory contributions from shipping companies of 2 USD per ton of fuel oil purchased for consumption (World Bank, 2020). Furthermore, the implementation of "market-based measures" (European Commission, 2013, p. 4) in the EU are mentioned as one approach to reduce greenhouse gas emissions. It is not explicit how such measures would be implemented, but one solution could be to include the shipping sector into the EU's Emission Trading System (ETS). In the ETS the companies must acquire emission allowances through grants or by purchase, to be allowed to cause emissions. A working paper from the international monetary fund (IMF) concludes that a carbon tax can be an effective tool to reduce emissions within the maritime sector (IMF Working Paper No. 18/203, 2018). An international agreement on emissions taxation for the shipping industry is still just a possibility of the future, but there are already some locations where the shipping industry is subject to cost for emissions. In Norway there has been a carbon tax since 1991 that also applies to the shipping industry (Norwegian Ministry of Climate and Environment, 2019). The rate of this tax is planned to be increased by 5% annually to 2025.

To summarize, it is not for us to predict the future direction of bunker fuel prices, but the uncertainty of fuel prices combined with the possibility of a future tax on emissions lead us to the conclusion that controlling fuel consumption will remain a crucial part of cost control in shipping.

3.2 Other costs

Managing fuel consumption is a crucial part of cost control, but proper management requires considerations for a multitude of other cost drivers as well. Important factors are wages of the crew, insurance cost, maintenance and repair, marine insurance and P& I, management costs and port charges. While fuel consumption is a largely is a function of distance and speed, other costs are functions of other metrics. Many costs are a function of time. Such time dependent costs can be the financial costs, wages, maintenance and overhead, or the opportunity costs that accrue when one operation excludes the opportunity to execute another.

An idle vessel will still inflict costs upon the ship owner. The payments to the crew do not necessarily stop just because the vessel is unused, especially if the idle time is for a short period of time. Furthermore, wear and tear will always be present, causing future maintenance costs. And regardless of its use, a vessel binds a lot of capital that the ship owner could otherwise have invested elsewhere. This means that disregarding all payable costs associated with the vessels there are continuously running financial opportunity costs associated with owning it. In this regard the ship-owner really wants to avoid situations with unused vessels.

Opportunity costs

If the shipping companies in general does a good job of finding use for its fleet, then the schedulers are faced with an opportunity costs associated with their choices. Such an assumptions relies on the company’s ability to find alternatives uses for its vessels. This can be other transporting tasks or renting the vessel out to another company, or in the case of prolonged inactivity or selling it.

The size of a vessel’s opportunity cost obviously depends on the vessel and the market in general. It may be periods or situations where the value of these opportunities is very low or zero and thus have a small impact on the schedulers decisions.

Port charges

Every time a vessel goes to port to load or discharge cargo there are port charges like terminal handling charges and cargo dues. Many ports will charge differently depending on the size and type of the vessel. The size dependent part of the charge can typically be calculated as a positive function of the vessel’s gross tonnage, meaning that a large vessel will be charged more than a smaller one.

4. Scheduling

The process of finding a good schedule is two-fold. First feasible schedules must be found, then a decision must be made on how they should be evaluated in order to select one of them. This chapter mention some of the different aspects of finding feasible schedules and of how to decide which schedule to use.

4.1 Considerations when finding a good schedule

Multiple aspects must be accounted for simultaneously when planning how to carry out all contracts of affreightment within the terms of each contract. A complete schedule contains information about which vessels that will be used and the route that each of them will sail. It also tells which cargo each vessel will pick up along its route, as well as the time it is picked up and the time it is delivered.

A minimum requirement imposed on the schedulers is to produce schedules that are practically possible to execute while delivering all cargo in accordance with the contracts. However, the task quickly becomes challenging when additional goals like cost reduction is introduced.

4.1.1 Vessel selection and cargo allocation

The attributes of the different vessels are important to consider when selecting vessels to transport the different cargos. This can a ships load capacity, speed limitations and its whereabout at the start of the planning period. The physical size of a ship can also be relevant regarding its compatibility with the different ports or other physical restrictions like canal transit.

Costs associated with the different vessels is also a crucial aspect of vessel selection. There are differing costs associated with crew, fuel consumption and port charges that may depend on the size or type of the vessel. In a comprehensive assessment of the company’s overall operations the opportunity cost of each vessel should also be accounted for.

4.1.2 Speed

The sailing speed plays an important role for controlling fuel consumption as well as making sure the vessel enters the scheduled ports within the agreed upon time windows. Variable speed gives the schedulers more flexibility when creating schedules, but in terms of cost control it also complicates the problem by a lot.

Higher speed will result in higher total fuel consumption for a given distance, all else equal. Fuel consumption of a vessel often increases with the square of its speed within a speed interval depending on each vessel (Keh-Kim, Boung-Yew, & King-Hwa, 2018). A higher speed will result in less time spent covering a given distance, but all else equal, the amount of saved time is not enough to justify the increased fuel consumption. Increasing the speed by 10% at a given distance will increase the total fuel consumption at that distance by more than 10%.

Zaman, Pazouki, Norman, Younessi, & Coleman (2016) performed experiments that reproduced the finding of a non-linear relationship. They measured total fuel consumption on a given distance for different speed settings and found an optimal sailing speed for that vessel in terms of minimizing fuel. They display the findings of the experiment in the plot shown in *Figure* 2 (Zaman, Pazouki, Norman, Younessi, & Coleman, 2016). This also shows that total fuel consumption increases below the speed that is optimal for the vessel in the experiment. This supports similar findings regarding speed and fuel consumption in shipping. However, such speeds are rarely used and there is probably no reason to go slower than the fuel minimizing speed setting while sailing.

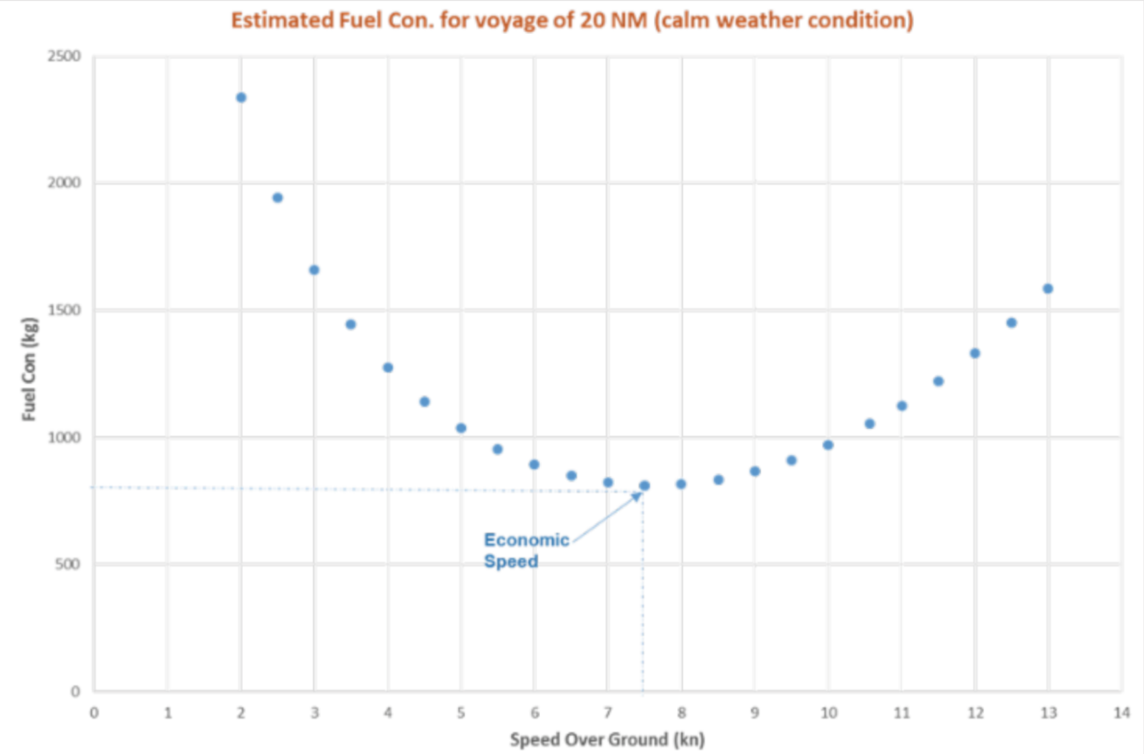


Figure 2 - Note. From "Utilising Real-Time Ship Data To Reduce Fuel Consumption And Carbon Emission" by *(Zaman, Pazouki, Norman, Younessi, & Coleman, 2016, p. 3)*

4.1.3 Vehicle routing

A core part of this problem is *the vehicle routing problem*, a very well-known problem within the field of mathematical programming. Attempting to solve this manually will quickly reveal the potential size of the problem. Any small routing choice may have large implications for the entire schedule. It will affect the vessels available load capacity, thus potentially changing the set of cargos it has capacity to pick up next. It may lead the vessel into a route where it must travel longer than necessary or into a schedule with situations of empty backhaul. It is also too difficult to solve for the general mathematical formulation of the problem when applied to many real situations. Instead, it is often solved by introducing heuristics and other approaches to the formulation (Laporte, 2009).

4.2 Evaluating schedules manually

To find a feasible schedule and to evaluate if it is good, is not a trivial task. The decision maker may excel at finding schedules, but it will be hard or even impossible to know if the schedule is good without comparing it to all others. The number of feasible schedules can be huge and grows very rapidly as the problem increases.

To appreciate the potential number of feasible schedules one can consider a simple case of a *vehicle routing problem with pickup and delivery*; Here 5 cargoes are to be picked up at different locations and delivered at 5 other locations by one vessel (The single vessel makes the proper term for the problem a *travelling salesman problem with pickup and delivery*). Each cargo will be picked up and delivered in its entirety, meaning that each pickup location is visited exactly once, and each delivery location is visited exactly once. In this example, we disregard constraint like vessel cargo capacity or time windows for pickup and delivery. There is at least one specific set of visit-sequences that are truly counterproductive and can be discarded; Any visit-sequences where the delivery location of a cargo is visited before its corresponding pickup location is useless. After discarding those visit-sequences the total number of potentially useful ones can be expressed as $5! * \prod_{n=1}^{5}\big((2 * n) - 1\big) = 113{,}400$. In theory this means that the scheduling department will have to find and compare 113,400 visit-sequences to conclude with certainty which one that has the shortest distance, lowest cost or largest/smallest of any other metric they care about.

Accounting for constraints like cargo capacity and time windows the number of feasible schedules will in most cases be drastically reduced. Also, schedulers use heuristics and their experience to exclude many feasible schedules from consideration. Still the point remains, a comprehensive assessment of a schedule’s quality requires many other feasible schedules to be found and compared. Furthermore, heuristics that intuitively makes sense can in some cases exclude good schedules. Finding schedules using mathematical programming or other approaches proposed in the literature is much less prone to miss solutions that for the human eye may be “hidden” behind intuition and heuristics.

From the vast number of feasible schedules that often exist, it is not necessarily difficult to find some of them. The decision problem facing the schedulers is to select which one to use. If the goal is to select the very best, then all schedules must be found and then compared. Doing this task manually will require huge amounts of effort for any problem of some size.

In comparison, any algorithm or mathematical programming approach, if properly formulated, will find (or implicitly find) and compare all feasible schedules at a superior pace. To summarize this chapter, specific approaches exist that is specialized in certain parts of the scheduling problem that is at an advantage compared to the entirely manual approach.

5. The model

The purpose of our model is to support schedulers in the pursuit of low-cost schedules that delivers on a set of given contracts. Our scope is not to decide if contracts are profitable. The set of contracts is given exogenously and are in terms of the model mandatory to complete. Because total revenue is given exogenously from the contracts, we disregard revenue and have taken the approach of minimizing costs rather than maximizing profit.

5.1 Discussion

In this section we explain the factors or dimensions we consider to be most important in a good schedule and discuss some other factors not included in the model. When we selected which factors to include, exclude or just include implicitly we mainly used three criteria: *How much value does the factor potentially add to the solution?* Meaning how much of an impact the factor can have on the cost of executing a solution produced by the model. *How much solve time does the factor add to the model?* Some factors could help the model find better solutions, but if they add much time required to solve the problem it is perhaps not worthwhile. *Is the data to evaluate the dimension properly available?* Some dimensions we would like to account for would require inputs to the model that either were not easily available or would require extra work on the part of the scheduler.

Because we actively have excluded some factors from the tool that is still present in the real world it is important to emphasize that the model is unable to produce comprehensive solutions ready for execution. This means that a proposed solution can save the scheduler time and labor while providing useful insights based on what a finished schedule might look like. However, review and adjustments by the schedulers is required to account for aspects not considered in the model. We will elaborate on this in *chapter 6 - Using the model and interpreting the results.*

5.1.1 Factors in the model

The core of the model is the general *capacitated heterogeneous vehicle routing problem with time windows and with pickup and delivery*. For this we need inputs on the capacity of the vessels, the amount of capacity each cargo lay claim to, travelling time between all ports as well as time windows and locations in all contracts.

We simplify all capacity and cargo measurements to only account for weight. The volume and shape of a cargo, as well as its stowing properties, is obviously crucial to regard before any schedule can be executed. A windmill component may have a low weight, but still not fit into a cargo compartment. Yet, we only account for weight in our model formulation. This is done to simplify the explanation and presentation of our formulation but does not break the generality of our approach.

Fuel consumption

As already established, fuel consumption plays a major role in the total operating costs of a vessel. Both fuel prices and emission fees are for the most part beyond the control of the shipping company, and the costs they inflict are directly correlated to fuel consumption. Unlike price and fees, many factors that affect total fuel consumption is within the schedulers realm of control. For this reason, total fuel consumption will be an undisputable part of our model. Fuel consumption is affected by a multitude of factors like distance sailed, engine type, the size of the vessel, weather, total load onboard, shape of the hull, coating condition of the hull etc. In the following we discuss the most important factors for reducing fuel consumption.

Distance

Regarding total fuel consumption and the travelling time between ports, required inputs are all distances between all ports and travelling speed. The main driver of fuel consumption is travelled distance and consequently selection of visit-sequences. As already discussed under *4.1.3 - Vehicle routing* all choices regarding visit-sequences made by the scheduler can have a large impact on travel distance. Because of its direct effect on fuel consumption, minimizing total travel distance will be a crucial part of the objective function.

In the model formulation each vessel will have a distance matrix that are specific for that vessel. This is because the vessels may be at different locations at the start of the planning period, thus having different distances from there to the first port they visit. Furthermore, there are constraints regarding passages; A vessel too large for the Panama Canal will have a significantly longer distance from the east- to the west coast of the USA compared to a smaller vessel.

Load and fuel consumption

For fuel consumption we use the vessel's estimated consumption for a laden voyage. We do not include any calculations on the relationship between fuel consumption for different levels of load onboard. This is a crude approach and a decision that will have implications for the solution. All else equal this will overestimate the fuel consumed during ballast voyages like empty backhauls or when a vessel is travelling will less than a full cargo compartment. Avoiding unwanted outcomes like empty backhaul or travelling with low-capacity utilization are not explicitly implemented in the model. Yet, since a large part of the objective is to minimize distance, sailing with low-capacity utilization will implicitly be quite heavily penalize depending on the distance of that specific arc.

Too much weight will be given to the part of the objective that implicitly seeks to avoid sailing with low-capacity utilization. This is because we have chosen the estimates for laden fuel consumption in the objective. A result of this may be that some schedules that would be more cost effective may be overlooked. We still believe this simplification can be justified. Load/fuel-consumption functions for each vessel would add to the computational solve time of the problem as well as increasing the required detail of the inputs.

Vessel properties

Vessel properties like size, weight, engine age and condition or properties regarding the shape or condition of the hull could be included if we were to create our own estimates for fuel consumption. However, as we use explicit data on expected fuel consumption for each of the different vessels this allows us to not account for any of these properties in regard to fuel consumption as these presumably are incorporated in the estimates.

Speed

As already mentioned, the relationship between speed and fuel consumption per mile is not linear. This means that the speed settings at different arcs or legs of the route have an impact on total fuel consumption. Including explicit fuel consumptions for different speed settings for each ship could improve the fuel minimizing abilities of the model. This would require data on these exact functions for each ship. Although the exact fuel consumption functions may be available it is not necessarily interesting to know the exact fuel consumption at different speeds; As mentioned by Norstad et al. (2011), the total fuel consumption as a function of speed is convex for the feasible speed range of the vessel. This means that minimizing the speed while also holding it constant throughout the whole route will achieve the lowest possible fuel consumption. In other words; when the speed/fuel consumption function is convex and when regarding each route separately it is redundant to know the exact function in terms of fuel minimization. However, because the problem regards multiple vessels with different levels of fuel consumption at the same time, it is interesting to know how these differ in order to select fuel efficient vessels.

The introduction of speed variables for each vessel along every arc increase the amount of time required to solve the problem drastically. Parameters that are otherwise exogenous like sailing time and fuel consumption become endogenous when speed is introduced as variables. As presented in Norstad et al. (2011) the decisions regarding cargo allocation and vessel routing will simultaneously depend on the speed variables and vice versa. A general network flow formulation of our model with variable speed would only be able to solve tiny instances of the problem. Yet, regarding fuel consumption we considered speed to be too important to be excluded entirely. To reach some sort of middle ground we split the formulation into two problems: A main problem for cargo allocation and determining routes using fixed speed. Then a speed optimization formulation (SOP) is applied to the route of each vessel that is used in the optimal solution of the main problem. This is elaborated on under *5.2 - Model description.*

Other vessel costs

Minimizing sailing distance and reducing speed and is established as a target, but also other important aspects can affect vessel selection. Regarding fuel consumption it can be beneficial to assign multiple cargos to one vessel if the vessel has capacity for it. When two cargos are going from A to B at the same time there may be fuel economies of scale when those cargoes are assigned to one vessel instead of two. However, in absolute terms the vessels will differ in costs other than just fuel.

Consider this toy example: 2 containers are to be transported. There are 3 available vessels. 2 smaller vessels with capacity for one container each and 1 large vessel with capacity for five containers. The decision maker sees two viable options to minimize cost. *A*: both containers are transported by the large vessel. *B*: The smaller vessels transport one container each. If fuel is the only cost to consider, option *A* might be the least costly. Yet, if other size related costs like maintenance, crew, port charges are accounted for then the lowest cost option may be different.

Although fuel is a main cost driver in shipping, excluding other vessel specific costs will increase the chance of proposing sub-optimal schedules in terms of cost efficiency. To mitigate this, we need a way to price the different vessels in the model.

Operating costs and opportunity costs

From the perspective of the shipping company many of the costs inflicted by the vessels will partially accrue regardless of whether the vessels is sailing or idling. Financial costs, wages, insurance, and maintenance are such costs. Decisions regarding vessel utilization should be taken such that the operation of the whole company benefits and not just from the perspective of a given set of contracts. For this reason, we disregard these continuously running costs.

For our modelling approach we have assumed that the ship-owners are able to utilize their fleet and find alternatives for their vessels if they are not used. This has important implications for the scheduler; When a vessel is used for one task it is simultaneously excluded from being a part of other opportunities. This means that in the presence of alternative uses there are opportunity costs associated with using a vessel in a specific schedule. Because of the potential implication opportunity costs have in the comprehensive context of the shipping company's overall operations we concluded that it should be incorporated in the model.

Depending on its size, incorporating opportunity costs in the model will mainly have two effects. Firstly, it will introduce a time cost and consequently lead the model towards solutions that uses fewer vessel or vessel-days, unless it is cost effective or otherwise necessary to use more. Secondly it will give the model a better chance at evaluating the true cost of using one vessel over another. Consider this simple example: One vessel provides a slight advantage to the solution in terms of distance minimization. However, if the same vessel can be rented away at a great price, then the model will be able to account for this information. The outcome may be that the cargo is reallocated to other vessels to make that specific vessel available for other opportunities.

We incorporated this by introducing a vessel specific opportunity cost parameter to the objective function. In the objective this cost will accrue linearly with the amount of time the vessel is used. Depending on what the company deems appropriate for each vessel this parameter hold the time dependent opportunity costs associated with using the different vessel.

How much time a vessel is used is calculated from the time when its route start to when it ends. This gives no opportunity for “free” idle time in between tasks in the planning period. We justified this for two reasons. First is an assumption that the model will produce solutions where little idle time occurs. Second is that the periods of idle time may be too short for the vessel to take part in alternative opportunities and capitalize on those. An argument against this approach is that idle time before the start of the route or after its end is treated differently from the idle time within the route as shown in *Figure 3*. A counter argument that justifies this is that a vessel may take part in other operations that overlap the start- and end day of the planning period of the model.

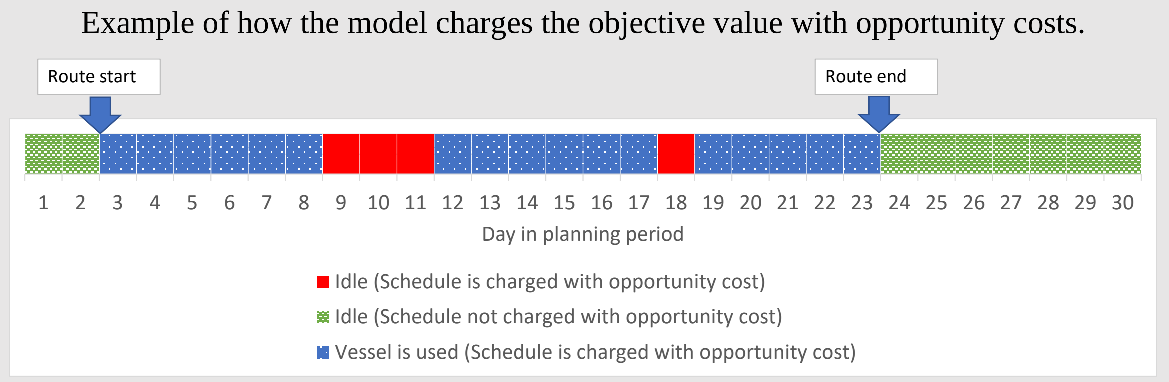


Figure 3 – A schedule is by the model charged the vessels opportunity cost for the start to the end of the voyage proposed by the model.

Port charges

Some part of the port charges only relates to the cargo and can be excluded from the model. Because we have excluded split loads from our model, each cargo will be loaded in its entirety exactly once and discharged once. This means that charges like *terminal handling charges* will accrue exactly twice for each cargo. Those charges are assumed independent of the type of vessel doing the pickup or delivery. Because they accrue independently of the schedule they can be excluded from the model.

On the other hand, the charges that does in fact depend on a vessel’s size and type will be included in the model. We incorporate this by a adding a vessel and port specific cost each time a vessel loads or discharges cargo. Because the size and rules regarding these charges can be different for all ports, the model requires a matrix containing size related port charges for all ports for all vessels.

5.1.2 Excluded factors

Split loads

Allowing a cargo to be split and delivered by two or more vessels can in some cases produce better solutions in terms of lower costs. When multiple vessels can be assigned to the same cargo then smaller vessels will be able contribute to the transport of larger cargos. In addition to this it will provide the model with a larger feasible area in its search for cost minimizing solutions. However, we decided to drop this factor from the model for two reasons. Instances of cost-efficient split loads are infrequent because it increases the number of required loading- and discharging procedures, thus increasing terminal handling charges. Including this option is also computationally demanding and makes the problem harder to solve. For these reasons we concluded that the potential gain of including *split loads* were not enough to justify the added time required to solve the problem and produce a solution.

Transshipment

Transshipment is not included in the model although it in some instances could produce more cost-efficient solutions. The justification for this is much the same as for split loads. It increases solve time and increases the required details in information on port charges. If the decision maker specifically wants a cargo to be delivered at a specific hub or location before it is shipped on to two locations from there, then this can be input to the model as three separate cargos.

Uncertainty

There is no concept of uncertainty in the model. High traffic or special incidents at port may result in queues and hold-ups the cause delays in a schedule. This can cause loss of contracts at that port but also jeopardize the chance of arriving within time windows during the rest of the route. Weather conditions may also alter how hard the engine must run and thus affect fuel consumption. Although there is a vast range of unexpected circumstances that can have a dramatic effect on a schedule, none of these are accounted for in the model. Uncertainty and how to handle it, is a crucial aspect of scheduling, but is outside the scope of this thesis.

5.2 Model description

Our proposed model formulation is to a large extent inspired by the network flow formulations as presented in Solomon et al. (2005), Brønmo et al. (2007) and Norstad et al. (2011).

Introducing variable speed and speed dependent fuel consumption into our model would make it near useless in terms of the time required to solve any problem. To maintain a minimum level of performance while also considering speed we decided to split the problem into two parts: One main problem for creating a schedule and a second part consisting of multiple speed optimizing problems.

Section 5.2.1 describes the first part of the model which regards the problem of creating a coherent schedule. The interaction between the main problem and the speed optimization problem is specified in *section 5.2.2*.

5.2.1 The main problem formulation

For our problem we reproduce most of the same mathematical formulations as presented by Brønmo et al. (2007) and Norstad et al. (2011). V is the set of vessels indexed by *v*. The number of contracts or cargos in the problem *n* defines the set $N^p = \{1, \ldots, n\}$ that represent the set of cargos and the set of pickup nodes indexed by i where cargo i is loaded at node i. $N^d = \{n + 1, \ldots ,2n\}$ is the set of delivery nodes meaning that cargo *i* is to be delivered at node *n+1*. This notation is useful for the cargo allocation in the formulation but does not in itself give any information about the physical location of the node. This means that multiple nodes can be located at the same port.

$N^c = N^p \cup N^d$ is the set of all nodes that are a pickup- or a delivery node. Furthermore, every vessel, v, has a starting node $node_v^{source}$ and a node representing the end of the vessels voyage $node_v^{sink}$.

Each vessel has a set of arcs, A_v, assosiated with it. This could be set up as a matrix $\{node_v^{source}, N^c\} \times \{N^c, node_v^{sink}\}$, but as every arc means an extra variable in the problem, some of them can be removed in preprocessing without risk of removing useful schedules. This assumes that each cargo is delivered in its entirety by one vessel. This means that arcs from a delivery node to its corresponding pickup node will never be useful. In other words, all arcs $(i + n, i)$ can be removed. Arcs where any of the nodes is associated with a cargo larger than that specific vessel's capacity can be removed from the set of arcs associated with that vessel. Each arc will have a parameter, $d_{v,i,j}$, containing the distance from node *i* to node *j* for vessel *v*. The reason for vessel specific distances is as already mentioned that the source nodes for each vessel may be at different locations and that some vessels may be unable to transit certain passages.

Sets

V Set of vessels

A_v Set of Arcs for vessel *v*

N^p Set of pickup nodes

N^d Set of delivery nodes

N^c = $N^p \cup N^d$. Set of cargo nodes. All nodes except the source- and sink-nodes.

Parameters

s_v Fixed speed (knots) for vessel *v*.

f_v Fuel consumption per mile at speed s_v for vessel *v*

$d_{v,i,j}$ Arc distance in miles for vessel *v*

$p_{v,i,j}$ Binary parameter. 1 if arc between node *i* and node *j* for vessel *v* is between different ports. 0 if both nodes are at the same port.

$c_{v,i}^{port}$ Cost representing additional port charges that dependent on the size and type of a vessel *v* at node *i*.

c_v^{time} Parameter representing the daily opportunity costs of using vessel *v*.

c_{v,s_v}^{fuel} Fuel cost per ton of fuel for vessel *v* sailing at speed *s*.

a_i^{serve} Service time at node *i* associated with loading, discharging and other activities while at port.

a_i^{open} Time at which the service time window opens at node *i*. This can be the layday start of a cargo at the pickup node, or the arrival window start at the delivery node.

a_i^{close} Time at which the service time window closes at node *i*. This can be the layday end of a cargo at the pickup node, or the arrival window end at the delivery node.

q_i Quantity of cargo to be loaded at node *i*. The parameter holds a negative value for delivery nodes.

Q_v Maximum total load at vessel *v*

Variables and variable declarations

$x_{v,i,j}$ Binary variable. 1 if vessel v sails from node i to node j, 0 otherwise.

$$x_{v,i,j} \in \{0,1\}, \quad v \in V, (i,j) \in A_v \tag{5.1}$$

$l_{i,v}$ Total load onboard vessel v just after the service is completed at node i.

$$0 \leq l_{i,v} \leq Q_v, \quad i \in \{node_{source}, N^c\}, v \in V \tag{5.2}$$

$t_{i,v}$ Time at which service begins at node i for vessel v.

$$a_i^{open} \leq t_{i,v} < a_i^{close}, \quad i \in \{node_{source}, \dots, node_{sink}\}, v \in V \tag{5.3}$$

Minimize:

$$\sum_{v \in V} \sum_{(i,j) \in A_v} x_{v,i,j} * \left(\left(d_{v,i,j} * f_v * c_{v,s_v}^{fuel} \right) + c_{v,i}^{port} \right) + \sum_{v \in V} \left(t_{node_v^{sink},v} - t_{node_v^{source},v} \right) * c_v^{time} \tag{5.4}$$

The objective function minimizes three elements: The fuel consumption, the size dependent port charges for each arc, and the opportunity cost associated with the time a vessel is used.

Subject to:

$$\sum_{v \in V} \sum_{j \in N_c} x_{v,i,j} = 1\,, i \in N^p \tag{5.5}$$

$$\sum_{(i,j) \in A_v | i = i^p} x_{v,i,j} - \sum_{(i,j) \in A_v | i = i^p + n} x_{v,i,j} = 0, \quad v \in V, i^p \in N^p \tag{5.6}$$

$$\sum_{(i,j) \in A_v | i = i^c} x_{v,i,j} - \sum_{(i,j) \in A_v | j = i^c} x_{v,i,j} = 0, v \in V, i^c \in N^c \tag{5.7}$$

$$\sum_{(i,j) \in A_v | j = node_v^{sink}} x_{v,i,j} = 1, v \in V \tag{5.8}$$

$$l_{i,v} = 0, \quad v \in V, (i,j) \in A_v |\, i = node_v^{source} \tag{5.9}$$

$$x_{v,i,j} * \left(l_{i,v} + q_i - l_{j,v} \right) = 0, \quad v \in V, (i,j) \in A_v |\, j \in N^c \tag{5.10}$$

$$x_{v,i,j} * \left(t_{i,v} + a_i^{serv} + \frac{d_{v,i,j}}{s_v} - t_{j,v} \right) \leq 0, \quad v \in V, (i,j) \in A_v \tag{5.11}$$

$$t_{i,v} + a_i^{serv} + \frac{d_{v,i,i+n}}{s_v} - t_{i+n,v} \leq 0, \quad v \in V, i \in N^p \tag{5.12}$$

5.5 Ensures that all pickup nodes are visited exactly once. The location of the vessels is covered in the follow three constraints: *5.6* ensures that a vessel that visit a pickup node also visit the corresponding delivery node. 5.7 ensures that no vessel can leave a cargo node, N^c, without first arriving at that node. Each vessel arrives a sink node exactly once, which is covered by 5.8.

The total load capacity of a vessel and the non-negativity of the load variable is ensured in the variable declaration in 5.2. Constraint 5.9 sets the initial load for all vessels when leaving its source node to 0 while constraint 5.10 ensures that the total load when leaving a node is equal to the sum of the quantity loaded at the node and the total load onboard the vessel when it left the previous node.

5.3 declares the variable for service start at node and constrains it at the same time to be within the open service window at that node. The last two constraints also relate to the time variables where 5.11 ensures the continuity for time service starts at the different nodes in the route for each vessel. No delivery node is visited before its corresponding pickup node already is visited which is covered by 5.12. The inequalities in *5.11* and *5.12* allows for waiting time by a node before service starts.

Linearization

We experienced better performance in terms of solve time when solving a linearized version of the formulation. The above formulation is non-linear but can be linearized if required using the same method as in (Christiansen, 1999). The load continuity constraint 5.10 can be linearized by replacing it with two constraints, 5.13 and 5.14. Constraint 5.11 is linearized by replacing it with 5.15.

$$l_{i,v} + q_i - l_{j,v} + (x_{v,i,j} * Q_v) \leq Q_v, \qquad v \in V, (i,j) \in A_v | \, j \in N^c \qquad 5.13$$

$$l_{i,v} + q_i - l_{j,v} - \left(x_{v,i,j} * Q_v\right) \geq -Q_v, \qquad v \in V, (i,j) \in A_v | \, j \in N^c \qquad 5.14$$

$$t_{i,v} + a_i^{serv} + \frac{d_{v,i,j}}{s_v} - t_{j,v} + \left(x_{v,i,j} * a_i^{close}\right) \leq a_i^{close}, \qquad v \in V, (i,j) \in A_v \qquad 5.15$$

5.2.2 Separating schedule creation and speed optimization

Creating a coherent schedule means that all cargo is allocated to a vessel, visit-sequences are found, and all constraints are complied with. This is all done in the main problem. When the main problem is solved then the visit-sequence of each vessel is passed one by one to the speed optimization problem (SOP). The SOP minimizes the vessel’s sailing speed. In order to reduce speed, it may change the timetable for when service start at each node as long as it is within the time constraints. The structure of the process is displayed in the flow chart in *Figure 4.*

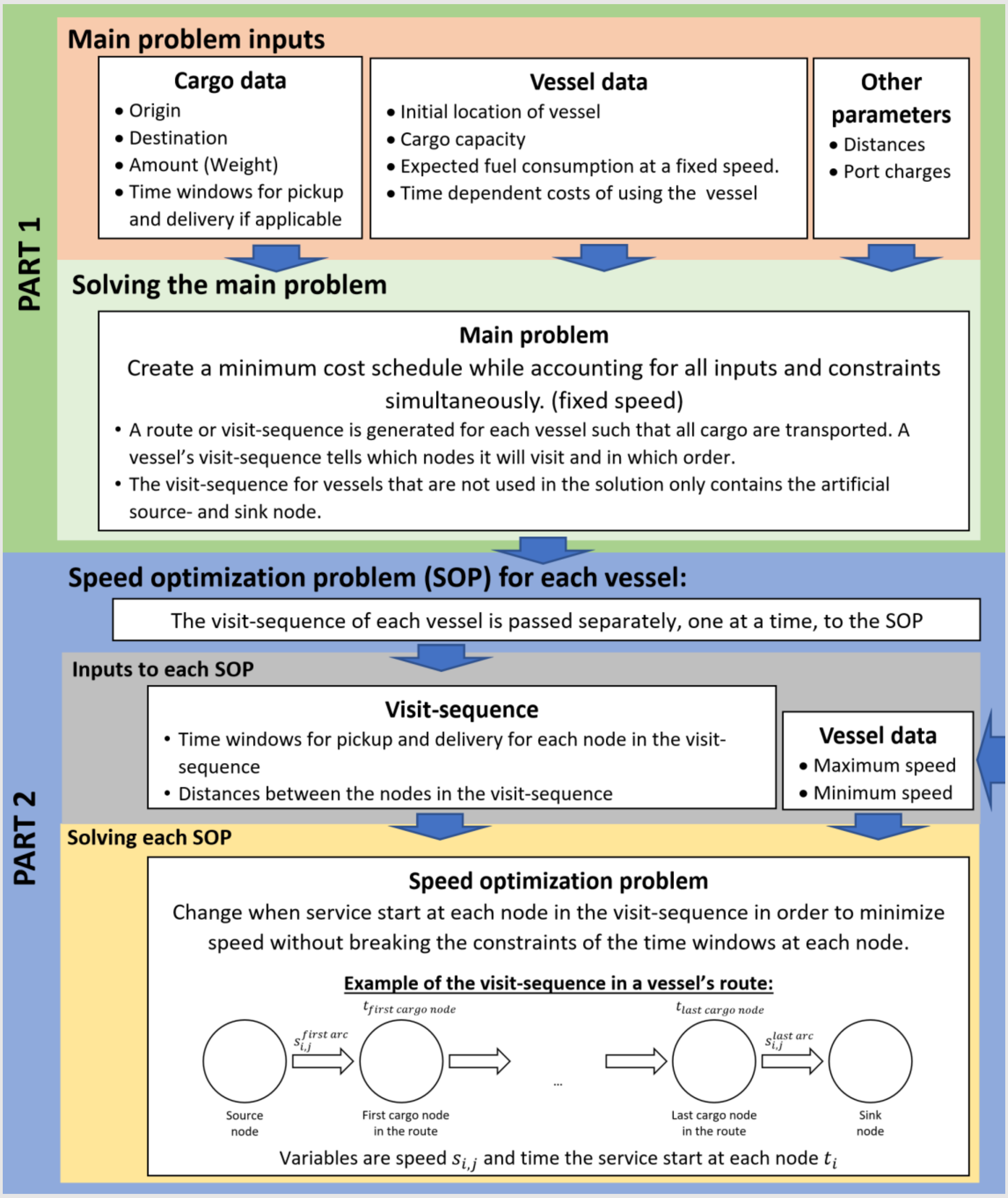

PART 1
Main problem inputs
Cargo data
• Origin
• Destination
• Amount (Weight)
• Time windows for pickup and delivery if applicable
Vessel data
• Initial location of vessel
• Cargo capacity
• Expected fuel consumption at a fixed speed.
• Time dependent costs of using the vessel
Other parameters
• Distances
• Port charges
Solving the main problem
Main problem
Create a minimum cost schedule while accounting for all inputs and constraints simultaneously. (fixed speed)
• A route or visit-sequence is generated for each vessel such that all cargo are transported. A vessel's visit-sequence tells which nodes it will visit and in which order.
• The visit-sequence for vessels that are not used in the solution only contains the artificial source- and sink node.
PART 2
Speed optimization problem (SOP) for each vessel:
The visit-sequence of each vessel is passed separately, one at a time, to the SOP
Inputs to each SOP
Visit-sequence
• Time windows for pickup and delivery for each node in the visit-sequence
• Distances between the nodes in the visit-sequence
Vessel data
• Maximum speed
• Minimum speed
Solving each SOP
Speed optimization problem
Change when service start at each node in the visit-sequence in order to minimize speed without breaking the constraints of the time windows at each node.
Example of the visit-sequence in a vessel's route:
$t_{first\ cargo\ node}$
$t_{last\ cargo\ node}$
$s_{i,j}^{first\ arc}$
$s_{i,j}^{last\ arc}$
...
Source node
First cargo node in the route
Last cargo node in the route
Sink node
Variables are speed $s_{i,j}$ and time the service start at each node t_i

Figure 4 - Flow chart of the model

This approach gives the model some way of adjusting speed to reduce cost. Unfortunately, the approach also comes with a downside. If speed was included in the main problem, then the model would find optimal solutions to feasible problems. By optimality we mean optimal in terms of the inputs available to the model. Splitting the model like we have done cannot prove optimality with the same confidence as if all variables were decided in one main problem. Using fixed speed in the main problem opens the possibility of excluding cost-efficient schedules that would have been found if speed were a variable.

Splitting the problem also introduced a new issue for us. What should be the fixed speed and the fuel consumption for each vessel in the main problem? It could be set to the speed at which the scheduler expects the route to be sailed at. If the scheduler's intuition is somewhat precise then the model will produce a solution based on a more accurate value of fuel consumption per distance sailed. On the other hand, if the schedulers set the speed slower than maximum speed it could exclude cost effective schedules that would require some parts to be sailed at a higher speed. In this regard maximum speed in the main problem will have more feasible schedules. Yet, a visit-sequence which requires some parts to be sailed at maximum speed leaves no room for unexpected delays and will probably be deemed too risky. Because of this we set the fixed speed for each vessel in the main problem to 90% of its maximum speed and uses the vessel's expected fuel consumption at this speed in the objective.

The fixed speed in the main problem is set with the assumption that the SOP often will find opportunities to reduce it. This means that we recognize that distance sailed often will be penalized too hard in the main problem.

The variable in the SOP is speed at each arc and the time variables for when service starts at each destination in the route. The objective here is to minimize speed and attempting to hold it constant during the entire route if the time constraints allow it.

5.2.3 The speed optimization formulation

Each visit-sequence from the schedule found by the main problem is passed to the speed optimization problem to minimize fuel consumption in each visit-sequence. For this problem we use a formulation like the one presented in Norstad et al. (2011).

The implementation of the SOP in AMPL is displayed in the Appendix under section 0. Note that we made some adjustments to account for a limitation regarding the solvers ability to handle the initial formulation of equation 5.18.

Sets and parameters

A^{route} Ordered set of arcs for each vessel used in the solution from the main problem.

s^{max}, s^{min} The maximum and minimum speed of the vessel used on the route the problem is applied on.

Otherwise, the sets and parameters are the same as in the main problem.

Variables

The speed optimization problem is only aware of the time and speed constraints as other constraints are enforced by the main problem.

$t_{i,v}$ The time variables keep the same declaration as in 5.3.

$s_{i,j}$ The vessel's speed along each arc in the route provided by the solution of the main problem.

$$s^{min} \leq s_{i,j} \leq s^{max}, (i,j) \in A^{route} \qquad 5.16$$

Minimize

$$\sum_{(i,j)\in A^{route}} d_{vessel,i,j} * s_{i,j}^{2} \qquad 5.17$$

The objective function will minimize sum of squared speed multiplied by distance for each arc. The squared speed is to achieve the convex properties as most real fuel consumptions. This has the effect of finding the route with lowest constant speed within the time constraints of the contracts.

Subject to

$$t_{i,v} + a_i^{serv} + \frac{d_{v,i,j}}{s_{i,j}} - t_{i+n,v} \leq 0, \qquad (i,j) \in A^{route} \qquad 5.18$$

The inequality in 5.18 allows the vessel to at node *j* and wait before the service begins.

6. Using the model and interpreting the results

A schedule proposed by the model can provide useful information and suggestions for the schedulers. Yet, the model is still far from a true representation of reality. To gain value from the model it is important to know its limitations. To better understand what a solution may look like we present a specific example in the following section. After that, a general discussion on how to interpret a solution is held in *section 6.2*.

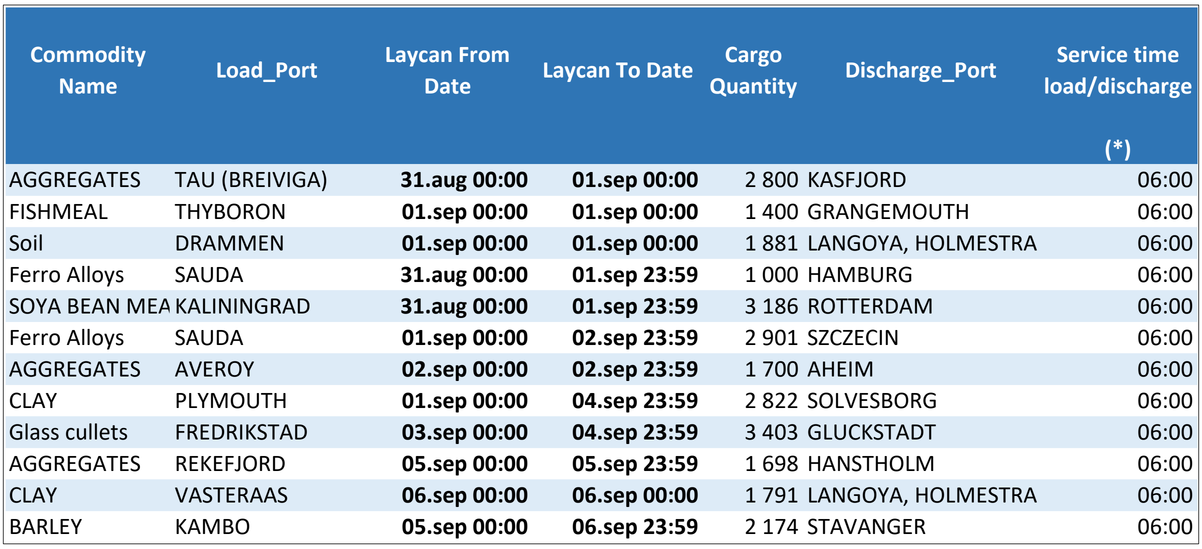
6.1 Example scenario with solution

6.1.1 Setting up the example problem

For the example we ran the model on a problem with 12 available vessels and 12 cargos to be transported. For the contracts we used some real data provided by Dataloy, but we did not have all inputs required to run the model. For this example, we replaced the missing inputs with some crude approximations and simplifications. Most of them are arbitrarily set and may cause the inputs to be somewhat odd. However, if the customers of Dataloy where to apply the model on a real case they would have access to those parameters. The missing parameters in question are basic measures like fuel consumption at different speed settings, port charges and how much time it takes to load each specific cargo. The purpose of this example is to elaborate on how to interpret a solution, not to produce the most realistic example.

Cargo

The cargo used in the example is listed in *Table 1*. On the first row is a cargo consisting of 2,800 tons of *AGGREGATES*. The cargo must start loading at *TAU (BREIVIGA)* port within the time window 31.aug 00:00 to 1.sep 00:00. The time required to load and to discharge is given a value arbitrarily by us. We sat this to 6 hours for all cargos. The *AGGREGATES* is then to be delivered at *KASFJORD* port.



Commodity Name	Load_Port	Laycan From Date	Laycan To Date	Cargo Quantity	Discharge_Port	Service time load/discharge (*)
AGGREGATES	TAU (BREIVIGA)	**31.aug 00:00**	**01.sep 00:00**	2 800	KASFJORD	06:00
FISHMEAL	THYBORON	**01.sep 00:00**	**01.sep 00:00**	1 400	GRANGEMOUTH	06:00
Soil	DRAMMEN	**01.sep 00:00**	**01.sep 00:00**	1 881	LANGOYA, HOLMESTRA	06:00
Ferro Alloys	SAUDA	**31.aug 00:00**	**01.sep 23:59**	1 000	HAMBURG	06:00
SOYA BEAN MEA	KALININGRAD	**31.aug 00:00**	**01.sep 23:59**	3 186	ROTTERDAM	06:00
Ferro Alloys	SAUDA	**01.sep 00:00**	**02.sep 23:59**	2 901	SZCZECIN	06:00
AGGREGATES	AVEROY	**02.sep 00:00**	**02.sep 23:59**	1 700	AHEIM	06:00
CLAY	PLYMOUTH	**01.sep 00:00**	**04.sep 23:59**	2 822	SOLVESBORG	06:00
Glass cullets	FREDRIKSTAD	**03.sep 00:00**	**04.sep 23:59**	3 403	GLUCKSTADT	06:00
AGGREGATES	REKEFJORD	**05.sep 00:00**	**05.sep 23:59**	1 698	HANSTHOLM	06:00
CLAY	VASTERAAS	**06.sep 00:00**	**06.sep 00:00**	1 791	LANGOYA, HOLMESTRA	06:00
BARLEY	KAMBO	**05.sep 00:00**	**06.sep 23:59**	2 174	STAVANGER	06:00

Table 1 - Cargo in example problem.
Made-up inputs are marked with "(*)" in the header.

Another assumption we have made for this example regards to cargos where *Laycan From Date* exactly equals *Laycan To Date*. One instance of is the *FISHMEAL*-cargo as seen on the second row in *Table 1*. We added 0.99 days to the *Laycan To Date* in such instances. This was done under the assumption that the cargo owner has not intended that this cargo must be loaded at exactly the minute the clock strike 00:00 on September 1st.

Vessels

The 12 vessels made available for the model is displayed in *Table 2*. As concluded in *section 5.2.2*, the fixed speed used in the first part of the model is set to 90% of a vessels maximum speed. This means that *Fuel per mile at laden 90% speed* in *Table 2* is represented as f_v in the main problem while s_v is equal to $0.9 * \textit{Max speed}$.

The port charges that depend on a vessels size and type, $c_{v,i}^{port}$, were simply set to a one-to-one relation with the cargo capacity of the vessel. Meaning that the vessel *CHANCE* is charged with a cost of 5,476 each time it visit's a node.

Vessel Name	Deadweight Summer (Capacity)	Speed Laden	Min speed	Max speed	Fuel per day at laden speed	Fuel per mile at laden speed	Fuel per mile at laden 90% speed	Initial port	Daily opportunity cost of use
(*)			(*)	(*)		(*)	(*)	(*)	(*)
CHANCE	4 750	10	9	20	6.5	0.027	0.088	GISMARVIK	5 476
DISCOVERY	4 045	12.3	11.1	24.6	5.5	0.019	0.060	SZCZECIN	4 738
JEWEL	3 800	10.5	9.5	21	5	0.020	0.064	BERGEN	4 479
LUMINOUS	3 800	10.5	9.5	21	5	0.020	0.064	SANTANDER	4 479
MAGNIFICENT	6 419	13	11.7	26	5.5	0.018	0.057	ROTTERDAM	7 180
SERENDIPITY	2 300	9	8.1	18	3.5	0.016	0.053	PORSGRUNN	2 851
SOMALI	6 041	10.5	9.5	21	8.5	0.034	0.109	BERGNESET	6 798
SPIRIT	5 200	11	9.9	22	5.5	0.021	0.068	IMMINGHAM	5 940
THE ETERNITY	1 572	8.6	7.7	17.2	5.5	0.027	0.086	DRAMMEN	2 024
THE FORTUNE	2 574	9.5	8.6	19	5.5	0.024	0.078	INVERGORDON	3 155
THE PROMISE	3 500	9.5	8.6	19	4.7	0.021	0.067	DUNKIRK	4 160
THE TEVIOT BANK	1 380	9	8.1	18	5.5	0.026	0.083	KARMOY	1 800

Table 2 - Available vessels in the problem.
Inputs that are approximations, made up or based on assumptions are marked with "(*)" in the header

Other inputs and solver settings

Fuel price, c_{v,s_v}^{fuel}, was set to 500. The start of the planning period was set to the earliest of the *Laycan From Date* from the cargo list in *Table 1*. In order to reduce the number of time variables in the problem each cargo was set to be delivered no later than 30 days after its corresponding *Laycan To Date*.

The distances used are real and were provided by Dataloy.

To solve the example, we used Gurobi as the solver and a relative optimality tolerance of 5% for the main problem as well as for the SOPs.

6.1.2 Example solution

Each arc used in the solution is listed with information about that arc. Unused vessels that in terms of the model "sails" from its artificial source node directly to its sink node are not displayed. For better readability, the presentation of the solution is split into two tables. The cargo and vessel load for each arc are presented in *Table 3* while time and speed along the arcs are presented in *Table 4*. In the following we have made a short description of some of the routes in the solution.

From the first row in *Table 3* we can read that the vessel *CHANCE's* initial location at the start of the planning period is *GISMARVIK*. From there it sails 23.09 miles to *TAU (BREIVIGA)* with an empty cargo compartment. The second row displays that 2,800 tons of *AGGREGATES* is loaded to the vessel.

Compared to the other vessels *MAGNIFICENT* is given the highest daily opportunity cost of use, but it also has the largest cargo capacity and a low fuel consumption per mile. Especially the low fuel consumption makes it seem intuitively reasonable that many cargos are assigned to this vessel.

Throughout the schedule some arcs are sailed without any cargo. This is not a desired outcome, but not necessarily possible to avoid. Especially smaller problems like this, with few cargos and fewer combinations of visit-sequences can be prone to multiple empty trips.

Vessel	Arc orig node	Arc orig port	Cargo weight loaded at arc orig	Commodity	Total cargo load when arc is sailed	Arc dest node	Arc dest port	Arc distance	Vessel capacity utilization
CHANCE	0	GISMARVIK	0	-	0	1	TAU (BREIVIGA)	23.09	0 %
CHANCE	1	TAU (BREIVIGA)	2800	AGGREGATES	2800	8	AVEROY	327.33	59 %
CHANCE	8	AVEROY	1700	AGGREGATES	4500	20	AHEIM	114.77	95 %
CHANCE	20	AHEIM	-1700	AGGREGATES	2800	13	KASFJORD	514.64	59 %
CHANCE	13	KASFJORD	-2800	AGGREGATES	0	25	0	0	0 %
DISCOVERY	0	SZCZECIN	0	-	0	3	KALININGRAD	283.56	0 %
DISCOVERY	3	KALININGRAD	3186	SOYA BEAN MEAL	3186	15	ROTTERDAM	712.29	79 %
DISCOVERY	15	ROTTERDAM	-3186	SOYA BEAN MEAL	0	6	PLYMOUTH	377.13	0 %
DISCOVERY	6	PLYMOUTH	2822	CLAY	2822	18	SOLVESBORG	874.82	70 %
DISCOVERY	18	SOLVESBORG	-2822	CLAY	0	12	VASTERAAS	324.65	0 %
DISCOVERY	12	VASTERAAS	1791	CLAY	1791	24	LANGOYA, HOLMI	806.59	44 %
DISCOVERY	24	LANGOYA, HOLMI	-1791	CLAY	0	25	0	0	0 %
JEWEL	0	BERGEN	0	-	0	5	DRAMMEN	379.68	0 %
JEWEL	5	DRAMMEN	1881	Soil	1881	17	LANGOYA, HOLMI	20.55	50 %
JEWEL	17	LANGOYA, HOLMI	-1881	Soil	0	9	FREDRIKSTAD	53.63	0 %
JEWEL	9	FREDRIKSTAD	3403	Glass cullets	3403	21	GLUCKSTADT	395.98	90 %
JEWEL	21	GLUCKSTADT	-3403	Glass cullets	0	25	0	0	0 %
MAGNIFICENT	0	ROTTERDAM	0	-	0	4	THYBORON	343.41	0 %
MAGNIFICENT	4	THYBORON	1400	FISHMEAL	1400	2	SAUDA	228.9	22 %
MAGNIFICENT	2	SAUDA	1000	Ferro Alloys	2400	7	SAUDA	0	37 %
MAGNIFICENT	7	SAUDA	2901	Ferro Alloys	5301	16	GRANGEMOUTH	399.04	83 %
MAGNIFICENT	16	GRANGEMOUTH	-1400	FISHMEAL	3901	14	HAMBURG	510	61 %
MAGNIFICENT	14	HAMBURG	-1000	Ferro Alloys	2901	19	SZCZECIN	342.82	45 %
MAGNIFICENT	19	SZCZECIN	-2901	Ferro Alloys	0	25	0	0	0 %
SERENDIPITY	0	PORSGRUNN	0	-	0	10	REKEFJORD	163.51	0 %
SERENDIPITY	10	REKEFJORD	1698	AGGREGATES	1698	22	HANSTHOLM	119.12	74 %
SERENDIPITY	22	HANSTHOLM	-1698	AGGREGATES	0	11	KAMBO	170.35	0 %
SERENDIPITY	11	KAMBO	2174	BARLEY	2174	23	STAVANGER	269.64	95 %
SERENDIPITY	23	STAVANGER	-2174	BARLEY	0	25	0	0	0 %

Table 3 - Solution to example: Cargo and load.
Green columns are values from the solution produced by the model or inputs. Orange headers were calculated for this display after the outputs were produced.

For the presentation of the time and speed in *Table 4* we have added three calculated columns; *Sail time* is how much time it takes to sail the arc. *Waiting time before service start at dest* is the time waiting by the destination node before service start. Finally, *Buffer time at dest before Laycan To Date* shows how much time is left before that laycan window closes when loading begins at that pickup node

Concerning route of *DISCOVERY* some worry may arise when regarding its timetable in *Table 4*. It has to keep a quite high speed from its initial location while making multiple visits before reaching the final minute of the *Laycan To Date* for picking up *CLAY* at *VASTERAAS*. This timetable leaves no room for delays. On the other hand, the vessel can sail even faster if unexpected delays should occur. It will be for the scheduler to assess the risk and decide if the route is acceptable or require adjustments.

Vessel	Service start at arc orig	Arc orig port	Service time at orig	Arc sail speed	Arc distance	Service start at arc dest	Arc dest port	Sail time	Waiting time before service start at dest	Buffer time at dest before "Laycan To Date"
CHANCE	**31.aug 00:00**	GISMARVIK	00:00	9	23.09	**31.aug 02:33**	TAU (BREIVIGA	2:33	0:00	21:26
CHANCE	**31.aug 02:33**	TAU (BREIVIGA	06:00	9	327.33	**02.sep 00:00**	AVEROY	36:22	3:07	23:59
CHANCE	**02.sep 00:00**	AVEROY	06:00	9	114.77	**28.sep 08:49**	AHEIM	12:45	614:09	
CHANCE	**28.sep 08:49**	AHEIM	06:00	9	514.64	**01.okt 00:00**	KASFJORD	57:10	0:00	
CHANCE	**01.okt 00:00**	KASFJORD	06:00	9	0	**06.okt 23:59**	0	0:00	138:00	
DISCOVERY	**31.aug 00:00**	SZCZECIN	00:00	17.85	283.56	**31.aug 15:53**	KALININGRAD	15:53	0:00	32:05
DISCOVERY	**31.aug 15:53**	KALININGRAD	06:00	17.85	712.29	**02.sep 13:47**	ROTTERDAM	39:54	0:00	
DISCOVERY	**02.sep 13:47**	ROTTERDAM	06:00	17.85	377.13	**03.sep 16:55**	PLYMOUTH	21:07	0:00	31:03
DISCOVERY	**03.sep 16:55**	PLYMOUTH	06:00	17.98	874.82	**05.sep 23:34**	SOLVESBORG	48:39	0:00	
DISCOVERY	**05.sep 23:34**	SOLVESBORG	06:00	17.85	324.65	**06.sep 23:45**	VASTERAAS	18:11	0:00	0:00
DISCOVERY	**06.sep 23:45**	VASTERAAS	06:00	11.1	806.59	**06.okt 00:00**	LANGOYA, HOI	72:39	617:31	
DISCOVERY	**06.okt 00:00**	LANGOYA, HOI	06:00	11.1	0	**06.okt 23:59**	0	0:00	18:00	
JEWEL	**31.aug 00:00**	BERGEN	00:00	9.5	379.68	**01.sep 15:58**	DRAMMEN	39:57	0:00	7:47
JEWEL	**01.sep 15:58**	DRAMMEN	06:00	9.5	20.55	**04.sep 12:20**	LANGOYA, HOI	2:09	60:14	
JEWEL	**04.sep 12:20**	LANGOYA, HOI	06:00	9.5	53.63	**04.sep 23:59**	FREDRIKSTAD	5:38	0:00	0:00
JEWEL	**04.sep 23:59**	FREDRIKSTAD	06:00	9.5	395.98	**04.okt 23:59**	GLUCKSTADT	41:40	672:14	
JEWEL	**04.okt 23:59**	GLUCKSTADT	06:00	9.5	0	**06.okt 23:59**	0	0:00	42:00	
MAGNIFICENT	**31.aug 00:00**	ROTTERDAM	00:00	14.31	343.41	**01.sep 00:00**	THYBORON	23:59	0:00	23:45
MAGNIFICENT	**01.sep 00:00**	THYBORON	06:00	12.73	228.9	**01.sep 23:59**	SAUDA	17:58	0:00	0:00
MAGNIFICENT	**01.sep 23:59**	SAUDA	06:00	11.7	0	**02.sep 05:59**	SAUDA	0:00	0:00	18:00
MAGNIFICENT	**02.sep 05:59**	SAUDA	06:00	11.7	399.04	**03.sep 22:05**	GRANGEMOUT	34:06	0:00	
MAGNIFICENT	**03.sep 22:05**	GRANGEMOUT	06:00	11.7	510	**05.sep 23:40**	HAMBURG	43:35	0:00	
MAGNIFICENT	**05.sep 23:40**	HAMBURG	06:00	11.7	342.82	**02.okt 23:59**	SZCZECIN	29:18	612:57	
MAGNIFICENT	**02.okt 23:59**	SZCZECIN	06:00	11.7	0	**06.okt 23:59**	0	0:00	90:00	
SERENDIPITY	**31.aug 00:00**	PORSGRUNN	00:00	8.1	163.51	**05.sep 00:00**	REKEFJORD	20:11	99:50	23:59
SERENDIPITY	**05.sep 00:00**	REKEFJORD	06:00	8.1	119.12	**05.sep 20:57**	HANSTHOLM	14:42	0:14	
SERENDIPITY	**05.sep 20:57**	HANSTHOLM	06:00	8.1	170.35	**06.sep 23:59**	KAMBO	21:01	0:00	0:00
SERENDIPITY	**06.sep 23:59**	KAMBO	06:00	8.1	269.64	**06.okt 17:59**	STAVANGER	33:17	674:38	
SERENDIPITY	**06.okt 17:59**	STAVANGER	06:00	8.1	0	**06.okt 23:59**	0	0:00	0:00	

Table 4 - Solution to example: Time and speed
Green columns are values from the solution produced by the model or inputs. Orange headers were calculated for this display after the outputs were produced.

SERENDIPITY sails with a tight time schedule before it reaches *KAMBO* with zero buffer time minute on 6th of October 23:59. However, it is still sailing its entire route at its minimum speed, allowing for faster sailing should delays occur. We can read the vessel is scheduled to reach the port on exactly September 6th at 23:45, with zero buffer time before the time window closes.

A remark regarding *Waiting before service start at dest* in *Table 4*; The start- and finish time of a vessel's route is irrelevant to the SOP as it is only regarding speed. This can produce odd start- and end times for the routes. An instance of this can be seen at the first arc sailed by *SERENDIPITY*. Here the vessel sail from *PORSGRUNN* on August 31st at minimum speed only to wait 99 hours and 50 minutes by *REKEFJORD* before starting to load. In terms of total time the vessel is in use, this is a waste of the vessel's time. Another instance of this is *MAGNIFICENT* sailing from *HAMBURG* on September 5th. It is sailing to *SZCZECIN* to unload its last cargo in the route but is set to wait 612 hours by the port before starting to unload.

The scheduler must be aware of this property in the current formulation of the SOP. However, the undesired outcomes only apply to the start or the end of a route. The new speed and timetable the SOP finds for the route in-between will have the desired outcome. This means that it is a trivial task to correct any odd outcome at the start or end of a route. It can probably be handled by a very simple postprocessing algorithm. An alternative approach could be to expand the objective of the SOP. This would require accurate fuel consumption functions of speed as well as the same opportunity cost formulation in the objective as in the main problem. Yet, we do not recommend this approach for future development; Rather we propose a more comprehensive development approach that will also handle this issue.

6.2 General interpretation

The scheduling department can feel confident that a proposed schedule is efficient in terms of the factors included in the objective function. However, multiple factors are not accounted for by the model and there are many aspects to be aware of when interpreting a solution.

The following section highlights some aspects the schedulers should be especially attentive of when reading a solution. In *section 6.2.2* we have made the point of why the model can produce useful insights despite some missing features.

6.2.1 Important aspects not covered by the model.

Simplifications

The potential effect some of our simplifications can have schedules that are feasible in the model, but impossible to execute in the real world. Only cargo weight and weight capacity are included. For some cargo it is volume or other measures that is the constraining factor. Schedulers must account for this when reading a proposed solution. Alternatively, additional cargo and capacity measures could be introduced directly to the model formulation without breaking any parts of it by adding variables and constraints. for

Also ignored by the model, are possible instances where one or more vessels are incompatible with some ports or with specific cargos.

Uncertainty

In addition to the physical constraints not included in the model there is no conception of uncertainty in its formulation. Unexpected delays occur regularly and if this is not accounted for in a schedule then late arrivals may cause the company to lose contracts. In port there may be hold-up or jams delaying the service of a vessel. Also, weather conditions, blocked passages or other factors that can cause delays are not part of the model.

If a port is known to frequently experience jams or delays the schedulers should be extra aware when a vessel is scheduled to arrive at this port. If a vessel is already scheduled with much idle waiting time outside this port, the scheduler might be satisfied. However, it is important to emphasize again that the model has no concept of uncertainty and will make no attempt to add buffer time to the solutions. This means that if a proposed solution has safe arrival times with buffers, then this has occurred by chance and not because the model looked for it. On the other hand, the speed minimizing element in the model can in some instances implicitly give the vessels leeway to mitigate unexpected delays by sailing faster. Consider a vessel with a very tight time schedule. This may be deemed too prone to late arrivals, but if the vessel is scheduled to sail at low speed, then the opportunity to sail faster can mitigate the fear of late arrivals. Either way, the schedulers must be prepared to adjust proposed solutions to achieve the level of time buffers they are comfortable with.

True costs and weightings in the objective

The cost parameters used in the main problem's objective function affects how the model weights the different aspects of the objective. Still, fuel consumption will in most cases carry the largest weight. In fact, too much weight will often be given to fuel consumption in the main problem. This is a result of the fixed fuel consumption parameters assigned to each vessel in the main problem. From the perspective of the model fixed speed means that vessel selection and distance sailed are the only factors affecting fuel consumption. We decided to set the fixed speed and its corresponding fuel consumption to a high level. This were done to not exclude schedules that would require higher speed along certain arcs. If the SOPs can reduce the speed on any arc, then expected fuel consumption per mile will be lower than what the main problem assumed when selecting a schedule. Hence too much weight may often be put on reducing fuel consumption.

When this is said, total fuel consumption for each vessel cannot be known for sure before each route is completed. Factors like actual sailed speed, weather conditions and load level will play a role in this. To summarize this point, some inaccurate weights of the different aspects of the objective function will not decimate the model's ability to select cost efficient schedules. Yet, the schedulers should be aware that the solutions are proposed by a model somewhat skewed towards giving fuel consumption more weight than it ideally should.

6.2.2 Making use of a solution proposed by the model.

A schedule proposed by the model will have its strengths and a set of missing aspects. It may be easy for the schedulers to detect the mentioned weaknesses in a proposed solution, but it may not be obvious how to handle them. Any adjustment the schedulers make to the solution will break its optimality in terms of the formulated objective function. Nonetheless, all of reality is not covered in the model. For this reason, the model's concept of optimality should not be interpreted as anything more than an approximation or direction for what a true optimal schedule could look like. Besides, the primary aim of the model is not to guide the schedulers towards true optimal or perfect solutions. Rather, it is to avoid the bad solutions and find helpful starting points for good schedules to be built on.

If the prices used in the objective function is somewhat accurate the schedulers can be confident that the proposed solution is truly among the better feasible schedules within the formulated objective. This means that a proper search for feasible schedules is executed and that all alternatives are compared or implicitly compared. The decision on which schedule the model is proposed will be the cheapest in terms of fuel costs, vessel costs and port charges. Either way, the proposed schedules will be a balancing of shortest path visit-sequencing and cheaper vessel selection.

7. Concluding remarks

7.1 Summary

In this thesis we have explored how exact approaches can be introduced to the toolbox used by shipping schedulers. Our motivation has been to improve the current approach used by Dataloy's customers. Much of the planning and scheduling is currently done manually without proper tools to evaluate the quality of a finished schedule. This is manageable if the sole purpose is to find feasible schedules, but if cost considerations are included this becomes more challenging. Our response has been a deterministic approach for finding schedules with respect to contracts and a selection of cost drivers, not accounting for aspects of uncertainty.

Fuel consumption is a major cost driver in the shipping industry, but in the view of cost-efficient scheduling other vessel-specific costs may also play an important role. We have decided to measure the cost of a schedule by five dimensions. First and foremost, fuel consumption as a function of distance sailed. This means that one of the shorter path schedules will be proposed by our model unless extraordinarily high costs should accrue from the other cost dimensions. Second, sailing at a lower speed will, all else equal, have a large impact on reducing total fuel consumption. The following three cost dimensions are related to vessel selection and are included to let the model differentiate and evaluate vessels in terms of cost. Every vessel has different properties regarding fuel consumption. They may also pay different charges at port depending on their size and type. Finally different opportunity costs are associated with each vessel and will be included as a time dependent parameter.

Our proposed approach is a model using a network flow formulation of the *capacitated heterogeneous vehicle routing problem with time windows and with pickup and delivery*. To solve the model on small problems within a reasonable amount of time, we have removed variable speed from the main problem. Still, to give the model some way to adjust speed we included a speed optimizing formulation to the model that is applied to each visit-sequence when the main problem is done.

We are not proposing an unsupervised model for providing complete schedules ready to be put right into action. The model has no concept of uncertainties integrated. Yet, we firmly believe that our model approach can contribute to the development of a tool that provide useful insights and improve the schedulers' ability to find cost effective schedules.

7.2 Further development - Performance and better incorporation of variable speed

In the pursuit of a tool equipped to provide schedulers with real support there are undoubtedly many aspects surrounding our model that must be further developed. However, in the context of the model formulation the primary focus for further development should be performance**.** The current formulation can only solve small instances of the problem within a reasonable amount of time. This is enough to test various adjustments to the objective, but not enough to provide useful support to most real-sized problems.

All the literature we have read that present their approach using network flow formulations also propose methods to be able to solve larger instances of the problem. For two reasons we consider a column generation approach to be especially appealing.

The first reason is that formulations as the one we use are well suited for splitting the problem into a subproblem and a master problem as stated in Solomon et al. (2005). This is because only one constraint connects the vessels to each other, in our case the constraint in equation 5.5. This allow the subproblem and master problem to only account for some constraints each. The solving process alternates back and forth between solving each problem and sending new information obtained in a solution to the other problem. The subproblem find routes and account for constraints regarding load and service times. The master problem evaluates the different combinations of routes by considering the total cost of each route combination subject to two constraints; That the combination of routes is such that all cargos are delivered exactly once, and that no vessel is used in multiple routes simultaneously.

The second reason we find a column generation approach appealing is because this allows speed to be considered before the final combination of routes is decided. This is done in Norstad et al. (2011). They find optimal speed and therefore a more accurate fuel consumption for each route before the cost contribution of the route is considered in the main problem. This contrast with our current approach where speed is fixed during the process of deciding on a combination of visit-sequences.

8. References

Brønmo, G., Christiansen, M., & Nygren, B. (2007). Ship routing and scheduling with flexible cargo sizes. *Journal of the Operational Research Society, 50*.

Christiansen, M. (1999). Decomposition of a Combined Inventory and Time Constrained Ship Routing Problem. *Transportation Science 33 (1),* 3-16.

Christiansen, M., Fagerholt, K., Nygreen, B., & Ronen, D. (2013). *Ship routing and scheduling in the new millennium.* European Journal of Operational Research 228.

European Commission. (2013). COM (2013) 479 - Integrating maritime transport emissions in the EU's greenhouse gas reduction policies. Brussels. Retrieved April 27, 2021, from https://ec.europa.eu/clima/sites/default/files/transport/shipping/docs/com_2013_479_en.pdf

Ge, J., Zhu, M., Sha, M., Notteboom, T., Shi, W., & Wang, X. (2019). Towards 25,000 TEU vessels? A comparative economic. *Springer Nature Limited 2019*. doi:https://doi.org/10.1057/s41278-019-00136-4

Keh-Kim, K., Boung-Yew, S. L., & King-Hwa, R. Y. (2018). *Prediction of Ship Fuel Consumption and Speed Curve by Using Statistical Method.* Journal of Computer Science & Computational Mathematics. doi:10.20967/jcscm.2018.02.002

Korsvik, J. E., Fagerholt, K., & Gilbert, L. (2011). A large neighbourhood search heuristic for ship routing and scheduling with split loads. *Computers & Operatins Research*.

Laporte, G. (2009). Fifty Years of Vehicle Routing. *Transportation Science*, 408-416. doi:10.1287/trsc.1090.0301

Meyer, J., Stahlbock, R., & Voss, S. (2012). Slow Steaming in Container Shipping. *2012 45th Hawaii International Conference on System Sciences*, (pp. 1306-1314). doi:https://doi.org/10.1109/HICSS.2012.529

Norstad, I., Fagerholt, K., & Laporte, G. (2011). Tramp ship routing and scheduling with speed optimization. *Transportation Research. Part C, Emerging Technologies, 19(5),* 853-865. doi:https://doi.org/10.1016/j.trc.2010.05.001

Norwegian Ministry of Climate and Environment. (2019). *The Norwegian Government's action plan for green shipping.* Retrieved from https://www.regjeringen.no/en/dokumenter/the-governments-action-plan-for-green-shipping/id2660877/

Parry, I., Heine, D., Kizzier, K., & Smith, T. (2018). *Carbon Taxation for International Maritime Fuels: Assessing the Options.* International Monetary fund. Retrieved from https://www.imf.org/-/media/Files/Publications/WP/2018/wp18203.ashx

Solomon, M. M., Kallehauge, B., Larsen, J., & Madsen, O. B. (2005). Vehicle Routing Problem with Time Windows. In G. Desaulniers, J. Desrosiers, & M. M. Solomon, *Column Generation* (pp. 67-98). Springer Science + Business Media, Inc.

United Nations. (2017). *Review of Maritime Transport 2017.* United Nations Conference on Trade and Development.

World Bank. (2020). *State and Trends of Carbon Pricing 2020.* Washington, DC:: World Bank. © World Bank. Retrieved from https://openknowledge.worldbank.org/bitstream/handle/10986/33809/9781464815867.pdf?sequence=4&isAllowed=y

Zaman, I., Pazouki, K., Norman, R., Younessi, S., & Coleman, S. (2016). Utilising Real-Time Ship Data To Reduce Fuel Consumption And Carbon Emission. *Shipping in Changing Climates Conference.* Retrieved from http://conferences.ncl.ac.uk/scc2016/

9. Appendix

9.1 Main problem formulation in AMPL

```
# main_problem.mod

param N_vessels default 1;
set Vessel_Name;
param Deadweight_Summer {Vessel_Name} >= 0;
param v_source_port {Vessel_Name} symbolic;
param v_min_speed {Vessel_Name} >= 0;
param v_max_speed {Vessel_Name} >= 0;
param v_index {Vessel_Name} >= 0;
param v_daily_cost_of_use {Vessel_Name} >= 0;
param v_fuel_per_mile_at_laden_speed {Vessel_Name} >= 0;
param v_fuel_at_90pst_speed {Vessel_Name} >= 0;
set VESSELS := {v in Vessel_Name: v_index[v] <= N_vessels};
param fastes_speed := max{v in VESSELS} v_max_speed[v];

set id_cargo;
param Commodity_Name {id_cargo} symbolic;
param Load_Port {id_cargo} symbolic;
param Laycan_From_Date {id_cargo} >= 0;
param Laycan_To_Date {id_cargo} >= 0;
param Cargo_Quantity {id_cargo} >= 0;
param Discharge_Port {id_cargo} symbolic;
param service_time_load_discharge {id_cargo};

####          _____          ####
####          NODES          ####
param N_cargos default 1;
param N_nodes_cargo := N_cargos * 2;
param node_source := 0;
param node_sink := N_nodes_cargo + 1;
set nodes_pickup_all := 1..N_cargos;
set nodes_delivery_all := (N_cargos+1)..N_nodes_cargo;
set nodes_cargo_all := 1..N_nodes_cargo;
set NODES_all := node_source..node_sink;
param node_quantity_to_load{i in NODES_all} :=
  if i in nodes_pickup_all then Cargo_Quantity[i]
  else if i in nodes_delivery_all then -Cargo_Quantity[i-N_cargos]
  else 0
  ;
param node_cargo_quantity{i in NODES_all} :=
  if i in nodes_pickup_all then Cargo_Quantity[i]
  else if i in nodes_delivery_all then Cargo_Quantity[i-N_cargos]
  else 0
  ;
param node_commodity_relation{i in NODES_all} symbolic:=
  if i in nodes_pickup_all then Commodity_Name[i]
  else if i in nodes_delivery_all then Commodity_Name[i-N_cargos]
  else "-"
  ;
param node_port_relation{v in VESSELS, i in NODES_all} symbolic:=
  if 0 == i then v_source_port[v]
  else if i <= N_cargos then Load_Port[i]
  else if N_cargos < i < node_sink then Discharge_Port[i-N_cargos]
  ;
set nodes_pickup{v in VESSELS} :=
  setof{i in nodes_pickup_all: node_cargo_quantity[i] <= Deadweight_Summer[v]} i;
```

```
set nodes_delivery{v in VESSELS} :=
  setof{i in nodes_delivery_all: node_cargo_quantity[i] <= Deadweight_Summer[v]} i;

set nodes_cargo{v in VESSELS} := nodes_pickup[v] union nodes_delivery[v];
set NODES{v in VESSELS} := {node_source} union nodes_cargo[v] union {node_sink};



####          ____________          ####
####          TIME WINDOWS          ####
param time_start_N_days_before_first_laycan_start default 0;
param time_max_N_days_after_pickup_before_delivery default 0;
param hours_per_time_unit default 24;
param time_end :=
  max{i in 1..N_cargos}Laycan_To_Date[i] + time_max_N_days_after_pickup_before_delivery;
param first_laycan_start := min{i in 1..N_cargos} Laycan_From_Date[i];
param time_start := first_laycan_start - time_start_N_days_before_first_laycan_start;

param node_service_window_open{i in NODES_all} :=
  if i in nodes_pickup_all then Laycan_From_Date[i]
  else
    if i in nodes_delivery_all then Laycan_From_Date[i-N_cargos]
    else time_start
  ;
param node_service_window_close{i in NODES_all} :=
  if i in nodes_pickup_all then
    if Laycan_From_Date[i] = Laycan_To_Date[i] then Laycan_To_Date[i]+0.99
    else Laycan_To_Date[i]
  else
    if i in nodes_delivery_all
      then Laycan_To_Date[i-N_cargos] + time_max_N_days_after_pickup_before_delivery
    else time_end
  ;


####          ____          ####
####          ARCS          ####
set arc_ship_is_unused := {(node_source,node_sink)};
set arcs_start := {{node_source}, nodes_pickup_all};
set arcs_end := {nodes_delivery_all,{node_sink}};

# no ship return to the pickup node after arriving at the corresponding delivery
# node (Assumes entire cargo is delivered at the same time)
set arcs_cargo := {i in nodes_cargo_all, j in nodes_cargo_all: i != j && i!=j+N_cargos};

set ARCS_all := arc_ship_is_unused union arcs_start union arcs_end union arcs_cargo;
set ARCS_v_all{v in VESSELS} :=
  setof{(i,j) in ARCS_all:
    node_cargo_quantity[i] <= Deadweight_Summer[v]
    && node_cargo_quantity[j] <= Deadweight_Summer[v]} (i,j);


param nodes_in_arc_has_different_time_windows{v in VESSELS, (i,j) in ARCS_v_all[v]} :=
  if (
    node_service_window_open[i] == node_service_window_open[j]
    && node_service_window_close[i] == node_service_window_close[j]
    ) then 0
  else 1
  ;
param arc_is_between_different_ports{v in VESSELS, (i,j) in ARCS_v_all[v]} :=
  if node_port_relation[v,i] == node_port_relation[v,j] then 0
  else 1
  ;
```

```
set ARCS{v in VESSELS} :=
  setof{(i,j) in ARCS_v_all[v]:
    i < j
    || arc_is_between_different_ports[v,i,j] = 1
    || nodes_in_arc_has_different_time_windows[v,i,j] = 1}
    (i,j);

set ARC_DIST dimen 2;
param Distance_in_nm {(i,j) in ARC_DIST};
set ports = setof {(i,j) in ARC_DIST} i;

param arc_dist{
  v in VESSELS, i in NODES_all, j in NODES_all: i!=node_sink && j!=node_source
  } :=
  if j == node_sink then 0
  else
     if node_port_relation[v,i] == node_port_relation[v,j] then 0
     else Distance_in_nm[node_port_relation[v,i],node_port_relation[v,j]]
  ;

param service_time {i in NODES_all} :=
  if i in nodes_pickup_all then service_time_load_discharge[i]
  else
    if i in nodes_delivery_all then service_time_load_discharge[i-N_cargos]
  else 0
  ;

param fuel_price default 500;
param vessel_dependent_port_charges{v in VESSELS, i in NODES[v]} default 0;
param fixed_sail_speed{v in VESSELS} default 10;
param fuel_consumption_at_fixed_speed{v in VESSELS} default 0.02;




var x{v in VESSELS, (i,j) in ARCS[v]} binary; # (5.1)
var l{node_source..N_nodes_cargo, v in VESSELS}  >= 0, <= Deadweight_Summer[v]; # (5.2)

# (5.3)
var t{i in NODES_all, VESSELS}
  >= node_service_window_open[i],
  <= node_service_window_close[i];

# (5.4)
minimize vessel_cost_adjusted_distance:
  sum{v in VESSELS, (i,j) in ARCS[v]}
    x[v,i,j] * arc_dist[v,i,j]*fuel_consumption_at_fixed_speed[v]*fuel_price
  +
  sum{v in VESSELS, (i,j) in ARCS[v]} x[v,i,j] * vessel_dependent_port_charges[v, i]
  +
  sum{v in VESSELS} v_daily_cost_of_use[v] * (t[node_sink,v]-t[node_source,v])
  ;


subject to
vessel_used_time_lower_bound{v in VESSELS}: t[node_sink,v] - t[node_source,v] >= 0;

# CONSTRAINS FOR ENSURING THAT ALL CARGO IS DELIVERED
# (5.5) All pickup nodes are visited at exactly once
c02{i in nodes_pickup_all}:
  sum{v in VESSELS, (arc_i,arc_j) in ARCS[v]: arc_i==i}
    x[v,i,arc_j] = 1;
```

```
# (5.6) Ensures that the vessel which visited the pick-up node also visits the
# corresponding delivery node
c16{v in VESSELS, i in nodes_pickup[v]}:
  sum{(arc_i,arc_j) in ARCS[v]: arc_i==i} x[v,i,arc_j]
  - sum{(arc_i,arc_j) in ARCS[v]: arc_i==i+N_cargos} x[v,i+N_cargos,arc_j] == 0;


# SHIP LOCATION CONTRATINS
# (5.7) A ship must arrive at a node before it can leave it
c05{v in VESSELS, i in nodes_cargo[v]}:
  sum{(arc_i,arc_j) in ARCS[v]: arc_i==i} x[v,arc_i, arc_j]
  - sum{(arc_i, arc_j) in ARCS[v]: arc_j==i} x[v,arc_i, arc_j] == 0;

# (5.8) A ship must arrive at a sink node exactly once
c06{v in VESSELS}:
  sum{(arc_i, arc_j) in ARCS[v]: arc_j == node_sink} x[v,arc_i, arc_j] == 1;


# SHIP LOAD CONTRAINTS
# (5.9) Total load when leaving source node
initial_load_c07{v in VESSELS, (arc_i, arc_j) in ARCS[v]: arc_i == node_source}:
  l[arc_i , v] == 0;

/*
# (5.10) Total load on ship after leaving a node (non-linear)
c08{v in VESSELS, (i, j) in ARCS[v]: j in nodes_cargo[v]}:
  x[v,i,j]*(l[i, v] + node_quantity_to_load[j] - l[j, v]) == 0;


# TIME CONTRAINTS
# (5.11)  (non-linear)
c14{v in VESSELS, (i, j) in ARCS[v]}:
   x[v,i,j] * (
     t[i,v]
     + service_time[i]
     + arc_dist[v,i,j]/(fixed_sail_speed[v]*hours_per_time_unit
     )
   - t[j,v]) <= 0;
*/

# (5.12) No delivery node can be visited prior to its corresponding pick-up node
c17{v in VESSELS, i in nodes_pickup[v]}:
  t[i, v] + service_time[i]
  + arc_dist[v, i, i+N_cargos]/( fixed_sail_speed[v]*hours_per_time_unit )
  - t[i+N_cargos, v] <= 0;


# (5.13) (Linearization of 5.10)
c08{v in VESSELS, (i, j) in ARCS[v]: j in nodes_cargo[v]}:
  l[i, v] + node_quantity_to_load[j] - l[j, v]
  + x[v,i,j]*Deadweight_Summer[v] <= Deadweight_Summer[v];


# (5.14) (Linearization of 5.10)
c09{v in VESSELS, (i, j) in ARCS[v]: j in nodes_cargo[v]}:
  l[i, v] + node_quantity_to_load[j] - l[j, v]
  - x[v,i,j]*Deadweight_Summer[v] >= - Deadweight_Summer[v];

# (5.15) (Linearization of 5.12)
c14{v in VESSELS, (i, j) in ARCS[v]}:
   t[i,v] + service_time[i] + arc_dist[v,i,j]/(fixed_sail_speed[v]*hours_per_time_unit)
   - t[j,v] + x[v,i,j]*node_service_window_close[j] <= node_service_window_close[j];
```

9.2 Speed optimization problem in AMPL

```
# speed_optimization.mod

param vessel symbolic;

set ARCS_ROUTE dimen 2 default {} ;
set origin_nodes_in_ROUTE := setof{(i,j) in ARCS_ROUTE: i in nodes_cargo[vessel]} i;
set NODES_IN_ROUTE := {node_source} union origin_nodes_in_ROUTE union {node_sink};


# (5.3) - Time at which service begins at node for vessel
var t_sop{i in NODES_IN_ROUTE, VESSELS}
        >= node_service_window_open[i],
        <= node_service_window_close[i];

var speed{VESSELS, ARCS_ROUTE} >= v_min_speed[vessel], <= v_max_speed[vessel]; #(5.16)

# Extra variable because "Gurobi can't handle nonquadratic nonlinear constraints"
var sail_time{ARCS_ROUTE} >= 0;

# (5.17)
minimize fuel_consumption_as_square_funtion_of_speed{v in VESSELS}:
  sum{(i,j) in ARCS_ROUTE}  arc_dist[vessel,i,j] * (speed[vessel,i,j]**2)
  ;


subject to
# (5.18)
time_of_service_start_at_nodes {(i, j) in ARCS_ROUTE}:
  t_sop[i, vessel] + service_time[i] + sail_time[i,j] - t_sop[j, vessel] <= 0;

# Extra contraint because "Gurobi can't handle nonquadratic nonlinear constraints"
def_sail_time{(i, j) in ARCS_ROUTE}:
  sail_time[i,j]*(speed[vessel,i,j] * hours_per_time_unit) - arc_dist[vessel,i,j] = 0;



/*
# Replaced nonquadratic nonlinear constraint
time_of_service_start_at_nodes {(i, j) in ARCS_ROUTE}:
   t_sop[i, vessel]
   + service_time[i]
   + (arc_dist[vessel,i,j]/(speed[vessel,i,j]*hours_per_time_unit))
   - t_sop[j, vessel]
   <= 0;
*/
```

9.3 Run configuration in AMPL

```
reset;
reset options;
model main_problem.mod;
data fra_excel.dat;

let N_cargos := 12;
let N_vessels := 12;

let hours_per_time_unit := 24;
let time_start_N_days_before_first_laycan_start := 0;
let time_max_N_days_after_pickup_before_delivery := 30;

let fuel_price := 500;

for {v in VESSELS} {
  let fixed_sail_speed[v] := v_max_speed[v]*0.9;
  let fuel_consumption_at_fixed_speed[v] := v_fuel_at_90pst_speed[v];
  for {i in NODES[v]} {
  let vessel_dependent_port_charges[v, i] :=
    if i in nodes_cargo[v] then Deadweight_Summer[v]*0.1
    else 0
    ;
    }
  }

option omit_zero_rows 1, display_width 100, presolve_eps 0.001; option solver gurobi;
option gurobi_options 'outlev=1 presolve=2 mipgap=0.05'; # (If nonlinar : nonconvex=2)
printf '\n \n \n \n \n \n \nStarting solver: \n';
solve;
fix x;


printf  "%s\t%s\t%s\t%s\t%s\t%s\t%s\t%s\t%s\t%s\t%s\t%s\t%s\t%s\t%s\n" ,
        'Vessel',
        'Arc orig port',
        'Arc dest port',
        'Cargo weight loaded at arc orig',
        'Commodity',
        'Total cargo load (weight) at arc',
        'Arc orig node',
        'Arc dest node',
        'Arc distance',
        'Service start at arc orig',
        'Service time at orig',
        'Service start at arc dest',
        'Arc sail speed',
        't_orig_main',
        't_dest_main'
         > output2.txt;

param break_repeat_loop default 0;
param current_node;
set final_arc dimen 2 default {} ;

option solver gurobi;
option gurobi_options 'nonconvex=2 outlev=1 presolve=2 mipgap=0.05';
option solver_msg 1;
model speed_optimization.mod;
```

```
param del_1_infeasible default 0;
if solve_result = "infeasible" then {let del_1_infeasible := 1;}

for {v in VESSELS} {
  if del_1_infeasible = 1 then {break;}
  let break_repeat_loop := 0;
  print v;
  let current_node := node_source;
  let ARCS_ROUTE := {};
  repeat {
    for{(i,j) in ARCS[v]: x[v,i, j]=1} {
      if i == current_node then {
        let final_arc := {(i,j)};
        let ARCS_ROUTE := ARCS_ROUTE union final_arc;
        let current_node := j;
        }
      if current_node == node_sink then {
        for {(i,j) in final_arc} if i != node_source then {
          display ARCS_ROUTE;
          let vessel := v;
          objective fuel_consumption_as_square_funtion_of_speed[v];
          solve;
          for {(arc_voy_i,arc_voy_j) in ARCS_ROUTE} {
            printf  "%s\t%s\t%s\t%s\t%s\t%s\t%s\t%s\t%s\t%s\t%s\t%s\t%s\t%s\t%s\n" ,
               v,     #'Vessel',
               node_port_relation[v,arc_voy_i],     #'Arc orig port',
               node_port_relation[v,arc_voy_j],     #'Arc dest port',
               node_quantity_to_load[arc_voy_i],    #'Cargo weight loaded at arc orig',
               node_commodity_relation[arc_voy_i],     #'Commodity',
               l[arc_voy_i,v],     #'Total cargo load (weight) at arc',
               arc_voy_i,     #'Arc orig node',
               arc_voy_j,     #'Arc dest node',
               arc_dist[v, arc_voy_i , arc_voy_j],     #'Arc distance',
               t_sop[arc_voy_i,v],     #'Service start at arc orig',
               service_time[arc_voy_i],     #'Service time at orig',
               t_sop[arc_voy_j,v],     #'Service start at arc dest',
               speed[v,arc_voy_i,arc_voy_j],     #'Arc sail speed',
               t[arc_voy_i,v],     #'t_orig_main',
               t[arc_voy_j,v]     #'t_dest_main'
               >> output2.txt;
            }
          }
        let break_repeat_loop := 1;
        }
      }
    if break_repeat_loop = 1 then {
      break
      }
    } # End repeat
  }

close; # Closing output file
display _ampl_elapsed_time;
display _total_solve_elapsed_time;
display _ampl_elapsed_time + _total_solve_elapsed_time;




param del_1_infeasible default 0;
if solve_result = "infeasible" then {let del_1_infeasible := 1;}

for {v in VESSELS} {
```