Multi-step Optimization of Logistics Networks: Strategic, Tactical and Operational Decisions

Maarten Hendriks

Contents

Chapter 1

Introduction

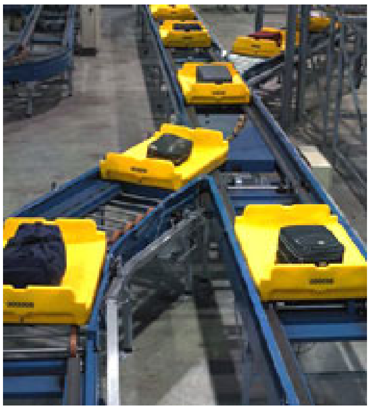
In the last decades a remarkable growth is noticeable in the number of people and goods that have to be transported from one place to another by different kinds of resources, e.g. buses, trains, airplanes and ships, but also transport belts, cranes, elevators and robots. A set of these resources linked together, is called a logistics network. Such a logistics network is usually run by a logistics provider, who faces the problem of delivering the right amount in the right place at the right time. Due to the ever-growing complexity of these networks, a logistics provider needs efficient tools to support its decisions leading to a maximal service level at minimal costs. The decision making can typically be classified into three levels: strategic, tactical and operational. The combined decisions on these inter-related levels, or even on one level specifically, are often too complex to be solved at once. A possible approach is to cut the total decision making problem into several subproblems, which then are solved step by step and where necessary alternatingly.

This dissertation proposes, develops and tests multi-step optimization methodologies to support the decisions of logistics providers in two particular logistics networks. Case studies suggest the suitability of the developed methodologies for practical applications. For each of the two logistics networks considered, the following generic conceptual approach is applied: we first reason about how to cut the overall problem into appropriate steps and compare our choice to existing methods. Next, for each chosen step, a mathematical model is constructed, which is optimized for a real-life data set. Finally, the results of the individual steps are discussed separately and the performance of the multi-step procedure as a whole is evaluated.

1.1 Logistics Networks

Logistics networks are present in every-day life, in all sorts and sizes. A chip assembly facility, the bus network in a town, and the luggage handling system at an airport are just a few examples among many. Figure 1.1 shows some pictures of (parts of) typical logistics networks.

A logistics network is commonly run by a logistics provider, who aims to deliver people and/or goods in the right place at the right time. In order to meet these goals at minimal costs, intelligent decisions at different levels of the logistics networks are required. In the past, a logistics provider could easily keep track of all decisions to be made. However, size and complexity of present logistics networks have grown that large, that intelligent decision support systems are required to help the provider in making the right decisions. This dissertation develops methodologies, insights and advices to support



(a) Luggage handling.

(b) Railway switch region.

(c) Parcel service.

Figure 1.1: Examples of logistics networks.

service providers in their decision making in the following two particular logistics networks (see also Figure 1.2):

1. A supply and demand distribution network,
2. A multi-terminal container port.

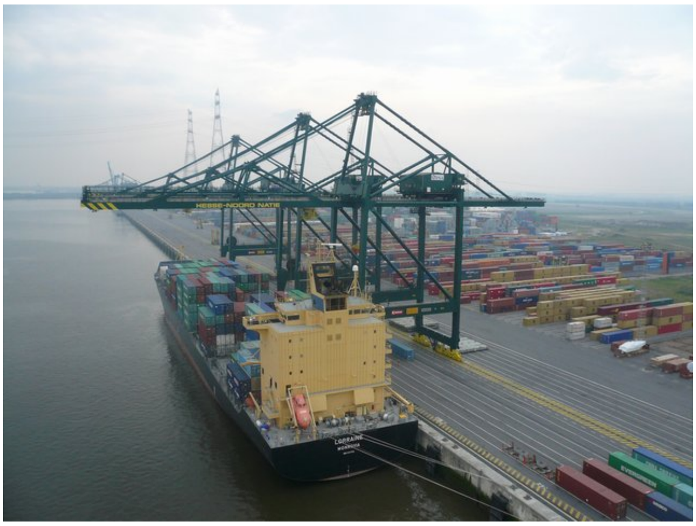
In the next section, we first discuss the common categorization for the decision making in logistics networks. Then we discuss these levels in detail for the two logistics networks addressed in this dissertation.

1.2 Decision levels in logistics networks

The three levels of decision making are categorized based on their time scale and the extent of influence they have on the network's performance:

- The strategic level is concerned with the number, location and size of particular network nodes (e.g. facilities and resources). Since the establishment of such a node (e.g. an airport or a train station) usually requires a lot of money and effort, these decisions have a long lasting effect on the network performance. Although

(a) Distribution truck.



(b) Container terminal.

Figure 1.2: Parts of the two logistics networks addressed.

strategic moves are continuously considered, actual changes are only executed after a number of years. Hence, these decisions are defined to be long term decisions.

- On the tactical level, a logistics provider has to decide on the design of i) the network topology, i.e. how the established nodes should be connected, and ii) the detailed layout of the particular nodes, i.e. the location of resources and products at one facility. To preserve continuity for employees and customers, and to restrict the network complexity this network design should not change too frequently. Decisions regarding the tactical level are therefore reconsidered after a number of months and defined as medium term decisions.

- The operational level is concerned with short term decisions, i.e. every day, every minute or every second (re)routing and (re)scheduling of people or goods through the established network.

The above described categorization in the decision making is now applied to the two particular logistics networks addressed in this dissertation.

1.2.1 Decision levels in a distribution network

A distribution network consists of a production company, which supplies different products at various facilities, and several retailers, which have a demand for these products. Dependent on the size of the network, production facilities and retailers are spread over a region, country or continent. Mismatch in supplies and demands might lead to a surplus or shortage of a particular product at a particular time. Since neither the supplier nor the retailers want to deal with buffering this variability in supply and demand, they hand the job to a third-party logistics service provider (LSP). His task is therefore to ship the right amounts from production facilities to retailers, possibly using intermediate warehouses for i) temporary storage of products to compensate the variability in supply and demand, and ii) consolidation of products so as to leverage on the economy of scales principle. The above described three level decision making can be applied to such a distribution network as follows:

For an LSP, the number, locations and capacities of suppliers and retailers in the distribution network are given. On a strategic level, the LSP has to decide on the number, location and capacity of intermediate warehouses. While making these decisions, not only has the current suppliers and retailers, but also potentially new markets have to be considered. These decisions have a long lasting effect on the performance of the distribution network and are therefore reconsidered on the long term time scale.

Once the location and sizes of all warehouses are known and fixed, the LSP has to make tactical decisions on the links/connections between these facilities, i.e. the network topology. In the distribution network considered, these links are the so-called line hauls being the roads, rails and/or waterways involved in the movement of freight between two facilities. According to the economy of scales principle, a line haul's efficiency increases if more products flow through it. The average line haul utilization can simply be increased by decreasing the total number of line hauls in the topology. Hence, an LSP strives to construct a topology with a small number of line hauls, which still performs well on an operational level. A line haul however cannot be established and removed on a short term basis, since it has a large impact on the operations of both involved facilities. Namely, after establishing a line haul, routings and schedules have to be modified and employees have to change their tasks according to these modified plans. Hence, to provide consistency, a network topology should not change too frequently. The establishment of line hauls is therefore reconsidered on the medium term time scale.

The operational tasks involve short term decisions: day-to-day routing and scheduling of the shipments along the chosen line hauls between the different facilities. Given the network topology and given supply and demand for a limited time horizon, the LSP has to decide on how much to send through which link facing costs for transportation, storage and penalties for early and late delivery.

1.2.2 Decision levels in a container port

Since 1960, containerization has grown rapidly and nowadays annually up to 114 million TEU's (1 TEU, Twenty feet Equivalent Unit, is a container of length 20 feet, width 8 feet and height 8 feet) are transported all over the world. In this world-wide network, a container port not only serves as a connection between land and sea container transportation, but also as a transshipment hub for forwarding containers between vessels.

A terminal operator coordinates and performs the logistics processes of discharging, loading, transporting and storing containers of various vessel lines in a particular terminal. Each vessel line owns a vessel fleet to maintain several repetitive loops along ports all over the world. Commonly, the number and phasing of the vessels of one loop are such that one vessel calls on each of its ports exactly once a week. Hence, the terminal operator has to service each customer (line) according to a cyclic timetable, which is repeated week after week. One can compare such a timetable with a bus or train schedule. The general levels of decision making in logistics networks can be applied to a multi-terminal container port as follows:

Dependent on its assets, a terminal operator provides its services in a certain number of terminals along the world. Once a new terminal is to be built or an existing one is to be overtaken somewhere, a terminal operator can take place in a competitive bidding procedure for operating this terminal in the future. These are long term decisions and have a large impact on the overall turnover of the operator. The expansion of the number of terminals is very expensive, however necessary to cope with the exponential growth of container transport. Hence, an operator has to continuously anticipate on the future market while considering the services in another terminal. Next, strategic decisions have to be made on the way to operate a terminal, e.g. which kind of resources (straddle carriers vs. trucks and stacking cranes) are used to transport the containers between quay and yard and how many quay cranes are required at the quay.

In many multi-terminal container ports, various operators take care of the logistics processes for container handling. Commonly, the tasks are divided such that one terminal operator is responsible for one terminal (or at least for the major share of one terminal). In an increasing number of ports (e.g. Singapore, Rotterdam and Antwerp) however, one terminal operator is responsible for multiple terminals. Given the quay lengths and storage capacities of the terminals, and given the load compositions of the calling loops, the first tactical problem is then to i) allocate a terminal and ii) a berthing time interval to each of the loops. Secondly, a berth position has to be allocated to each loop and an appropriate yard layout (which containers to stack where) has to be constructed. Together this results in a tactical timetable, which is reconsidered on the medium term time scale.

The tactical timetable depicts the allocation if all vessels arrive perfectly in time. However in practice, vessels are sometimes early or late (e.g. due to breakdown or bad weather conditions), and may have different call sizes and/or compositions each week. Moreover, quay cranes and other resources may brake down for an unknown period of time. The daily operational tasks of a port operator involve the management of the disrupted system to serve the vessel lines as good as possible at minimal costs. First, each vessel has to be allocated to a specific berth position within its terminal. The reference berth position of a vessel is usually taken closest to the position of its export containers in the stack. In this way, the travel distance of container carriers between vessels and stacks is reduced. Second, a schedule for quay cranes along with resources and its drivers has to be constructed to process a vessel within the agreed service time. These decisions are usually made every eight hours (one shift), and sometimes even a replanning takes place after four hours (half a shift). The detailed sequence of actual discharging, loading, transporting and stacking containers is updated at every container pick-up and drop-down.

1.3 Optimization of logistics networks

From the previous section it becomes clear that a logistics provider faces many decisions so as to run a logistics network efficiently. To satisfy his customers and to maximize his own profit, he has to strive for decisions that lead to a maximum customer satisfaction (service level) at minimum costs. These two objectives however are conflicting and the provider continuously considers a higher service, at the expense of additional resources (equipment and personnel). Customer satisfaction can be quantified easily, since logistics providers are usually charged by their customers when services are not provided in time. This enables the logistics provider to explicitly trade off his service level against his investment costs.

An appropriate way to optimize the decisions is by constructing and optimizing a mathematical model of (a part of) the logistics network. First, one has to select a vector of decision variables x, which have to be decided on by the, in this case, logistics provider.

Next, a suitable objective function $f(x)$ has to be constructed, which describes how the system performance (total costs) depends on the decision variables x. Finally, physical limitations or theoretical bounds on (parts of) the system have to be incorporated. To model this, one distinguishes between inequality constraints $g(x) \leq 0$ and equality constraints $h(x) = 0$ represented by functions $g(x)$ and $h(x)$ of x that are bounded by or fixed to zero, respectively.

An optimization tool can then be used to optimize the model with respect to the objective function $f(x)$ (e.g. consisting of investment costs and costs for not delivering in time). The output of the optimization is a set of optimal decisions that leads to minimal objective costs. The generic mathematical form of an optimization problem looks as follows:

$$\begin{array}{rlcl} \min\limits_{x} & f(x) & & \\ \text{subject to} & g(x) & \leq & 0 \\ & h(x) & = & 0. \end{array}$$

1.3.1 Illustrative example

To illustrate the translation of a logistics problem into a mathematical problem, we apply the above described approach to a simple, every day example. Imagine, it is Monday afternoon 17:55 and suddenly you realize today is the last day to return a number of borrowed books to the library without paying a fine of 0.20 euro per book. The library closes at 18:00 and is located 4 km outside your home town, so going by bike today is not an option. You can take the bike tomorrow, but then you have to pay the 0.20 euro fine per book. Furthermore, there is the restriction that you are able to carry only 10 books on your bike and are not willing to bike the route more than once. If you take the car right away, you will deliver the books just in time, but spend 0.19 euro on gasoline per kilometer. What do you do? It is clear that the books cannot be returned without having some costs, no matter what decision you make: if you go by bike, you pay the fine, if you go by car, you pay the gasoline.

Assume you take the bike and you have to return a number of N books. The fine for one book is 0.20 euro, which makes the fine for N books $N \cdot 0.20$ euro. Moreover, there is the condition that you can only carry up to 10 books on your bike and you do not want to have to ride the bike to the library more than once. This means that you can only choose between the bike and the car if no more than 10 books are to be returned, in mathematical terms if $N \leq 10$.

Now assume you go by car and pay 0.19 euro per kilometer on gasoline. To drive to the library and back, a distance of 8 kilometers has to be covered, which will cost you $8 \cdot 0.19 = 1.52$ euro. Now dependent on your decision on the transport and the number N of books, you pay either $N \cdot 0.20$ euro or 1.52 euro.

The question here is, given these conditions how to incur minimal costs. Mathematically, the choice of your transport can be represented by a variable, which we denote by x. Dependent on the choice, this variable x is given a different value. We define the variable x to be 1 if you take the car, and zero if you take the bike. This can be represented as follows:

$$x = \begin{cases} 1 & \text{if you drive your car today,} \\ 0 & \text{if you drive your bike tomorrow.} \end{cases} \tag{1.1}$$

The costs can now be written as a function $f(x)$ of this decision variable x:

$$f(x) = 0.20 \cdot N \cdot (1 - x) + 1.52 \cdot x. \tag{1.2}$$

If you would decide to take the car ($x = 1$), then the term $(1 - x)$ in (1.2) becomes 0 and costs for fines are excluded. In this case, the only cost contribution remaining is the gasoline costs equal to $1.52 \cdot 1$. If you would decide to take the bike ($x = 0$), then you only pay the fine of $0.20 \cdot N$ and 0 for gasoline $(1.52 \cdot 0)$.

Finally, the restriction on the number of books N you can carry on your bike has to be formulated dependent on the decision variable x. We already saw that if $N \leq 10$, you can still decide on taking the bike or the car, and hence x can be either zero or one. If $N > 10$, you have to take the car, and hence for this case we should enforce x to be one. Mathematically, this can be achieved as follows. We propose the next inequality constraint:

$$g(x) = -Sx + (N - 10) \leq 0, \tag{1.3}$$

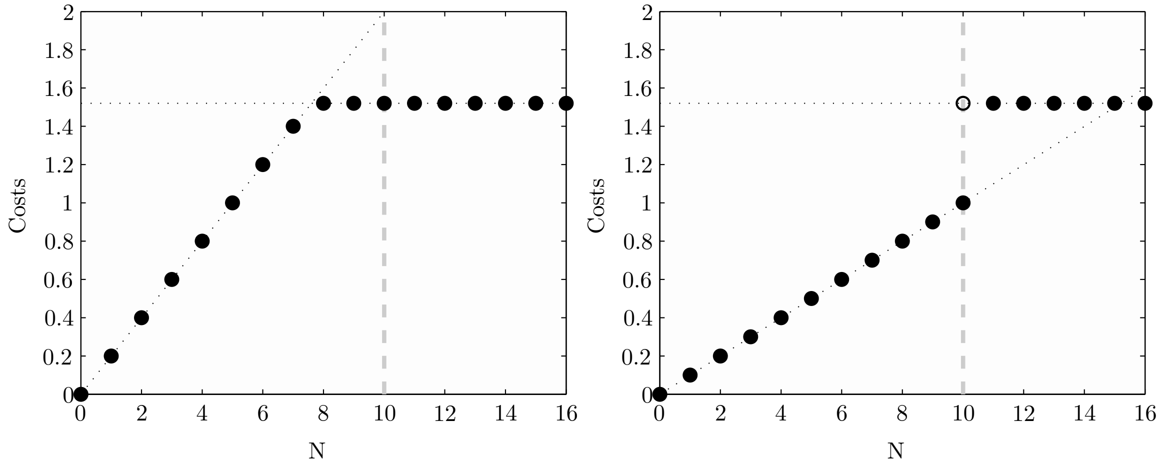
where S is a number still to be determined. The inequality constraint $g(x)$ in (1.3) implies that the term $-Sx+(N-10)$ has to be smaller than or equal to zero. For the case where $x = 0$ this is valid only if $N \leq 10$, for the case where $x = 1$ this is valid if $S \geq N - 10$. Hence, we define $S \geq N - 10$, since then independent on N, you can always choose the car ($x = 1$) and still satisfy (1.3). However, to satisfy (1.3) for $x = 0$, it has to be that $N \leq 10$, and this is exactly what we strived for.

Now, a decision on x has to be made such that the costs represented by the objective function $f(x)$ are minimal, and the inequality constraint $g(x)$ is satisfied. This can be depicted in the general form of an optimization model as follows:

$$\begin{array}{rl} \min\limits_{x} & 1.52 \cdot x + 0.20 \cdot N(1 - x) \\ \text{subject to} & -Sx + (N - 10) \leq 0 \end{array} \tag{1.4}$$

For the given parameter values, the dots in Figure 1.3a depict the minimal costs as a function of the number of books N you have to return. The thin black dotted lines represent the individual contributions of costs for going either by car or bike. Furthermore, the vertical grey dotted line depicts the maximum number of books you can carry on your bike. From this figure, the following is noticed: for a small number of books, the minimum costs equal the fine costs (you should go by bike), but from the break-even point on, the minimum costs follow the gasoline costs (you should go by car). Apparently, for a small number of books (up to seven), it is less costly to take the bike the next day ($x = 0$). If you need to return between 8 and 10 books it is cheaper to drive the car right away ($x = 1$). Finally, if you have to carry more than 10 books you have to use your car anyway ($x = 1$).

The parameter values have a crucial effect on the best choice and the corresponding minimal costs. Figure 1.3b for instance depicts the minimal costs for exactly the same problem, except that now the fine per book is 0.10 euro. From this figure it can be



(a) Fine per book is 0.20 euro.

(b) Fine per book is 0.10 euro.

Figure 1.3: Minimal costs as a function of the number of books to be returned N.

concluded that it is cheaper to drive the bike if you have to return up to fifteen books. However, if you have to deliver more than 10 books you have to take the car, resulting in a large cost increase at $N = 11$.

In the above described example, only one decision has to be made, which can easily be optimized in one's head. One can imagine however that in real-life logistics networks, many decisions have to be made facing a multi-objective function, subject to many constraints. Hence, finding the optimal decisions is far from trivial. For these kinds of systems, optimization by means of a mathematical model can be very helpful. This dissertation considers the decision making in two complex logistics networks, defines practically relevant and interesting subproblems, formulates appropriate mathematical optimization models and applies them to real-life case studies.

1.4 Multi-step optimization

As has been mentioned before, the considered logistics networks are of such a size, that the combined decisions on the strategic, tactical and operational levels or, even at one level, cannot be solved at once within the time allowed. Operational decisions for instance have to be made every second or minute of the day. A model that runs for an hour to propose these operational decisions is useless. The approach in this dissertation is to cut the overall problem into a number of subproblems each of which i) is practically interesting in its own right, and ii) can be solved within the time allowed at the level(s) concerned. The proposed models for instance enable to construct decisions on the long and medium term time scale within hours, while short term decisions can be constructed within minutes.

The price we pay by solving the subproblems sequentially is that the solutions found are no longer guaranteed to be optimal. In this dissertation, limited attention is paid to quantify the (expected) deviations between the found solution and an optimal solution. The focus is on the gains that can be achieved with respect to the solutions as currently applied in practice. Although some currently applied solutions might result from managerial decisions or negotiations that are not covered by our models, a quantification of the induced additional costs is made explicitly. This provides insights in the costs of modification and can support a logistics provider in his future decision making.

Solving the multiple subproblems can be done either i) sequentially, i.e. the output of one subproblem is fixed and used as an initial condition for the next subproblem, or ii) alternatingly, i.e. one alternates between different subproblems where the output of one subproblem is the input for the other and vice versa. In this dissertation, dependent on the subproblems discussed either a sequential or a alternating solution approach is applied.

Several studies on multi-step optimization for large systems can be found in literature, where either sequential or alternating solution approaches are applied. The study in [12] for instance discusses several approaches to solve problems in fluid and solid mechanics, in which coarse scale phenomena influence local phenomena and vice versa. These studies cut the global problem into different time and/or space scales to retrieve tractable subproblems, which can be solved sequentially or alternatingly. In this dissertation as well, most subproblems are formulated based on the different scales/decision levels involved. However, sometimes a problem is cut into two or more subproblems, which are all at the same scale/decision level. Therefore, the terminology "multi-step optimization" is used rather than "multi-scale optimization".

Some of the subproblems we consider are, to some extent, similar to problems addressed individually in literature, others, however, are not and result from the cuts in our overall multi-step approach. In the next two subsections, we first highlight the overall problem that is considered for the distribution network (Section 1.4.1) and the multi-terminal container operation (Section 1.4.2). Moreover, the multiple steps and solution methods for the corresponding network are highlighted and main results are summarized. Relevant references are cited, however a detailed literature review for each subproblem is only given at the beginning of the chapter that discusses the concerning subproblem.

1.4.1 Multi-step optimization of a distribution network

The three-level decision making in a distribution network has been discussed before and can be summarized as follows: i) Strategic decisions involve the number, location and capacity of warehouses, ii) on a tactical level one has to decide on the network topology, i.e. which links (highways, railways, waterways) to use between suppliers, warehouses and retailers and iii) the daily routing of vehicles (trucks, trains, vessels) through the chosen topology are considered to be operational decisions. In Figure 1.4, an illustration of the distribution networks considered is depicted. Production facilities, warehouses and retailers of different sizes are, in this example, spread among the Netherlands. The red lines represent the main highways connecting all facilities.

In this study, the strategic decisions are assumed to have been made and hence the number, location and capacity of warehouses are fixed. Furthermore, the number, location and average supply rate of production facilities, and the number, location and average demand of retailers are assumed to be given. Additionally, the actual daily supplies and demands per product type for a limited future horizon ahead (couple of days) are assumed to become available each day. Having this information, a third-party logistics provider deals with the joint problem of i) the generation of a tactical network topology on the medium term time scale and ii) the daily scheduling and storing of different types of products in the generated network.

Looking at Figure 1.4 again, this means that the provider has to select i) which of the red links (highways) to use for a longer time period and ii) how much to send through the

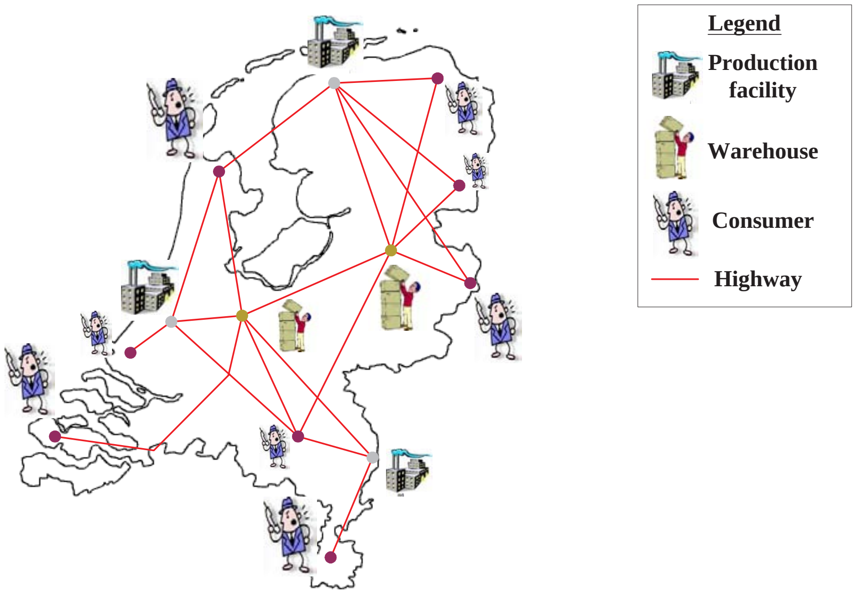


Figure 1.4: Illustration of a distribution network.

selected links each day. The LSP strives for a topology that has i) a small number of line hauls to have a high link utilization so as to leverage on the economy of scales principle and ii) a fixed number of links to provide consistent routings and schedules and reduce the organizational complexity, while still the operational performance is satisfactory. For real-life distribution networks, however, it is very time consuming to evaluate the operational performance of each possible topology to find the one that fits best these two objectives best. Hence, the complex problem of tactical and operational decisions is cut into two problems, i) the topology design problem and ii) the daily routing problem, which are solved in an alternating fashion: a coupled bi-level optimization method is proposed that evaluates only a very limited number of topologies to find one with a small number of links that still has a satisfactory operational performance.

A mathematical model is constructed to describe the product flows as a function of the daily operational decisions given a topology and given supply and demand on a restricted future horizon. This model is used to determine the minimal operational costs for that particular topology and these particular supply and demand time series. The bi-level method alternates between the tactical level of decision making and the operational level of decision making. The method is based on iteratively dropping links from a topology on the tactical level dependent on its performance at the operational level.

Results of a case study suggest that this method is very fast and still yields accurate solutions. Moreover, results suggest that the constructed network topologies are insensitive to changes in second and higher order moments of supply and demand distributions. The topology however appears to be sensitive to changes in the means of supplies and demands.

Many theories and solution approaches on the design of logistics networks and supply chains can be found ([11], [18], and [44]). The problem considered in this dissertation however investigates a specific and practically interesting problem, which to our knowledge has not yet been explored.

1.4.2 Multi-step optimization of a multi-terminal container operation

The three level decision making in a container operation can be summarized as follows: On the strategic level, the operator considers the enlargement of the number of terminals, the kind of operations within each particular terminal and the number of quay cranes in each particular terminal. Given the terminals and their capacities, the operator can service a certain number of periodically calling container vessels, and has to construct an appropriate tactical time table and berth allocation plan. Since in practice, container operations are heavily disturbed due to bad weather conditions and breakdowns, the terminal operator continuously needs to reschedule on an operational level in order to return to the tactical plan as soon as possible.

This study addresses decision problems faced by a terminal operator on the strategic, tactical as well as on the operational level. Similar to the distribution network problem, the overall problem cannot be solved to optimality and is for this case cut into four subproblems, which are then solved step by step (either sequentially or alternatingly). Figure 1.5 shows a graphical illustration of the decisions at the various levels. The colored dotted rectangles indicate the selected subproblems, which are shortly discussed in the next subsections.

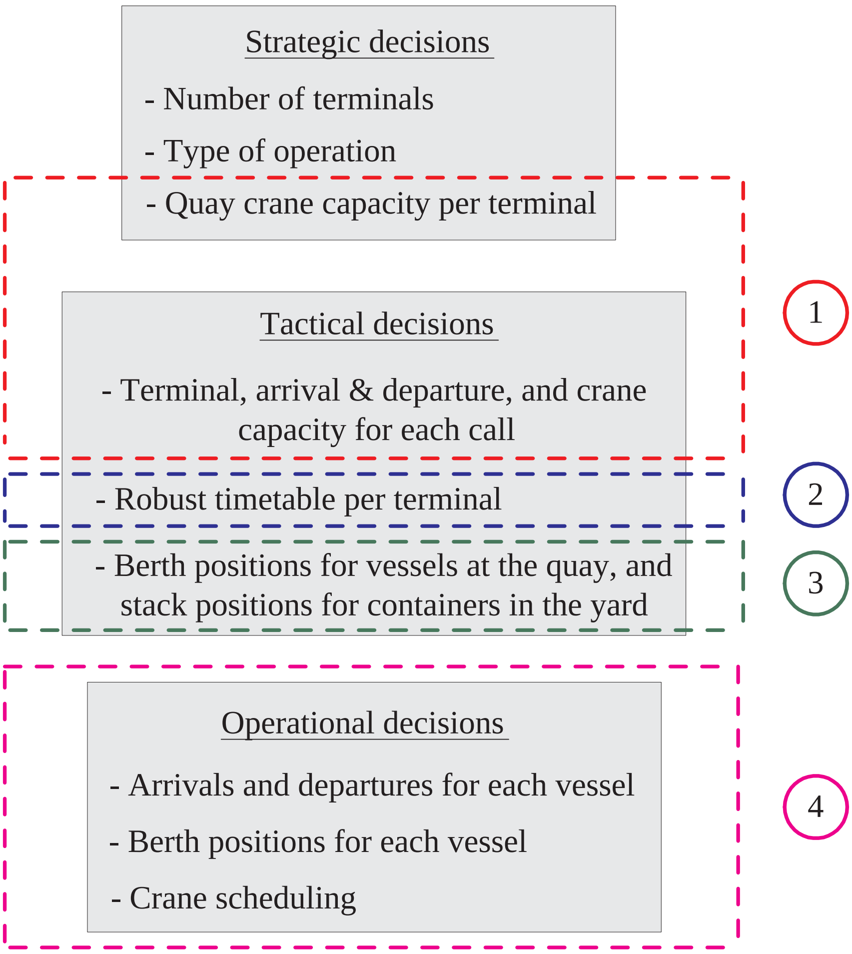


Figure 1.5: Illustration of the chosen subproblems for the container operation.

Since the decisions on the strategic level induce the highest costs and have the largest impact on the system's performance, it makes sense to start solving the subproblem(s) at the strategic level, and use the results as inputs for solving the subproblem(s) at the tactical level. These solutions in turn are input for the solution methods for the subproblem(s) at the operational level.

In literature many studies can be found on container operations, some of which discuss problems that to some extent are similar to one of the four main steps addressed here. The authors in [45], [46], [49] and [43] give extensive overviews of the main issues in container ports and discuss existing solution approaches. Comparisons between our studies and existing ones are made in the next subsections.

Step 1: Terminal, arrival and departure, and crane capacity allocation to vessels

Nowadays, mega-container ports consist of multiple terminals, which are relatively close to each other. In most of these ports, the logistics terminal operations are performed by multiple container operators. Typically, one terminal operator takes care of the logistics processes in one terminal. However, in an increasing number of ports (e.g. Singapore, Rotterdam and Antwerp), one terminal operator is responsible for a number of terminals. The logistics problems in such ports can no longer be considered per terminal for two main reasons. Firstly, peaks and troughs in quay crane utilization should be avoided and vessel calls should be spread evenly over the various terminals. Secondly, transshipment containers will very likely generate inter-terminal traffic resulting in costs that should be taken into consideration. All possible flows of containers through a number of terminals managed by one operator are depicted in Figure 1.6.

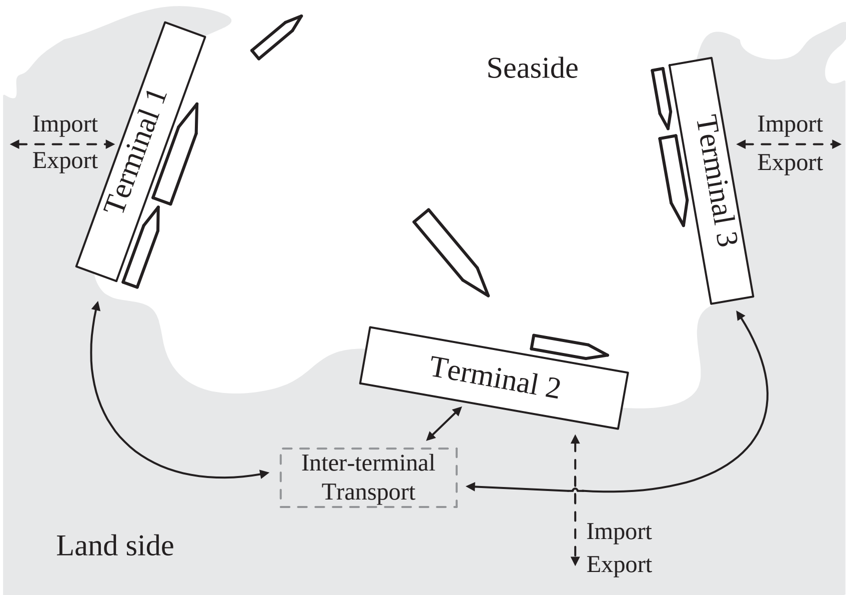


Figure 1.6: Container flows in a cluster of multiple terminals.

This research is supported by the company of PSA HNN, that operates a number of container terminals in Antwerp, Belgium. The terminals' length, and the set of current customers (shipping lines) calling at this port are. The actual remaining question on the strategic level is hence about the number of quay cranes minimally required, or in particular: "Can we do more business with the same number of quay cranes and where is this spare capacity?".

The current policy of the terminal operator is to strive for satisfying the preferred tactical arrival and departure times and preferred berthing terminal of shipping lines to provide a high service level. It is interesting to determine the potential quay crane spare capacity and the potential reduction in inter-terminal transport if small modifications to the shipping lines' preferences are allowed. An optimization model that allocates a terminal, arrival and departure times, and crane capacity to a set of cyclically arriving vessels with known call sizes can already yield nice insights in potential spare capacity and cost reductions. This actually is the first in the sequence of optimization steps with respect to container operations. The allocation of exact positions of the vessels within a terminal and the actual quay cranes assigned to them is not relevant yet and is left to be determined in a subsequent subproblem. In this way, we are left with an practically interesting, well-defined problem that is still tractable from a computational point of view. Note that in this first step, we strive for improvements at the strategic level (spread work evenly, identify spare capacity) as well as improvements on the tactical level (reduction in carriers for inter-terminal transport).

An appropriate mathematical optimization model is proposed, which includes decisions regarding i) which terminal to call, ii) arrivals and departures and iii) crane capacities and incorporates flow equations for the container transport between the different terminals. The multi-objectives are to minimize i) deviations from preferred arrival and departure times, ii) the minimally required crane capacity, and iii) the amount of inter-terminal container transport. For randomly generated problem instances, the proposed model can be solved substantially faster than a more common approach. Subsequently, representative data, provided by PSA HNN, is used to set the parameter values in the model. Results of a case study suggest that significant reductions in both minimally required quay crane capacity and inter-terminal transport can be gained if only small modifications to the current allocations of only a part of the shipping lines were allowed. Many studies exist on the berth allocation problem within one terminal. Surprisingly, to our knowledge, there is no study on the allocation to a cluster of inter-related terminals so far. A recommendation in [39] however indicates the growing need for the optimization of a multi-terminal container port.

Step 2: Increasing the robustness of a tactical timetable per terminal

The first optimization step results in a cyclic timetable per terminal. If all vessels arrive accordingly, this timetable can be executed over and over again. However, the arrival times of container vessels in a terminal are heavily disturbed by delays in other terminals in their loop and by bad weather conditions during travel. Therefore, a terminal operator cannot expect the vessels to arrive exactly in time. On the other hand, a shipping line cannot demand immediate service if it is delayed for instance one day. As a compromise, the terminal operator and each shipping line agree on two kinds of arrivals: i) within and ii) out of a so-called arrival window, which is placed around the scheduled arrival time and typically has a width of eight hours. If a vessel arrives within its window, the operator guarantees to operate the vessel within an agreed process time. If a vessel arrives out of its window, the terminal operator is not bound to any process time at all.

The goal of the second step optimization is to slightly modify a terminal's tactical berth plan, determined in the first step, into a tactical berth plan that is robust to all scenarios where all vessels arrive anywhere within their arrival window. In our definition, a berth plan is robust with respect to a given set of arrival windows if a feasible solution exists for each arrival scenario where all vessels arrive within their windows. The price for achieving this robustness is then the additional crane capacity reservation that is required in the worst case arrival scenario where all vessels arrive within their windows. The problem is hence to construct a window-based berth plan that minimizes the maximally required crane capacity for all scenarios where vessels arrive within their arrival windows. Arrival windows, and quay and crane capacity reservations have to be allocated to each of the vessels in the set. Note that although sufficient quay and crane capacities are reserved, the actual vessels' berth positions and quay cranes operating are left to be allocated in a subsequent subproblem. However, still we end up with a tractable problem that is practically interesting on its own right.

A mathematical optimization model is proposed that constructs a window-based plan and minimizes the maximally required crane capacity. Results on representative data provided by PSA HNN suggest that slightly modifying a berth plan, generated in the first optimization step, yields a significant reduction in the maximally required crane capacity. As a particular case, the optimization model finds a nominal berth plan, i.e. a berth plan that neglects disturbances and hence ignores the arrival window agreements. Results suggest that the window based plan requires slightly more crane capacity if vessels' arrivals would only slightly deviate from the scheduled arrivals for zero and narrowly bounded arrivals¿ However, the window based plan is much more robust than the nominal plan in case of relatively large deviations (which are still within the arrival window bounds).

The concept of building in pro-active robustness in tactical timetables has already been introduced in airline applications [7], [1] and [29], and railway applications [5], [50] and [5]. To the best of our knowledge, only one study [37] addresses the stochastic berth allocation problem by building in some kind of pro-active robustness. Given a timetable and delay distributions, the study in [37] allocates berth positions to vessels to minimize overlaps of vessels (i.e. two vessels at the same place at the same time) and deviations from preferred berth positions. In contrast to this study, we explicitly guarantee no overlaps and do use the flexibility of modifying i) the timetable and ii) crane capacity allocation to increase the robustness.

Step 3: Allocation of vessels' berth and containers' stack positions

Once the (window-based) berth plan has been constructed for each terminal and hence the tactical arrival and departure times are known, the next step is to allocate berth positions to the vessels. Containers to be loaded onto a vessel arrive at the terminal between a few weeks up to a few hours prior to the vessel's departure time. These containers are temporarily stored in designated areas in the terminals' yard. Once the vessel has been positioned along the terminal quay, carriers in between the quay and the yard start moving containers from the vessel to designated areas (unloading process) and from designated areas to the vessel (loading process). The berth position of a vessel and the position of its designated container area(s) hence determine the distance that has to be covered by the carriers. An illustration of a container terminal and the relevant logistics activities are depicted in Figure 1.7.

Given the tactical timetable and average call sizes, the goal of the third optimization step is to allocate i) vessels' berth positions at the quay and ii) container area positions

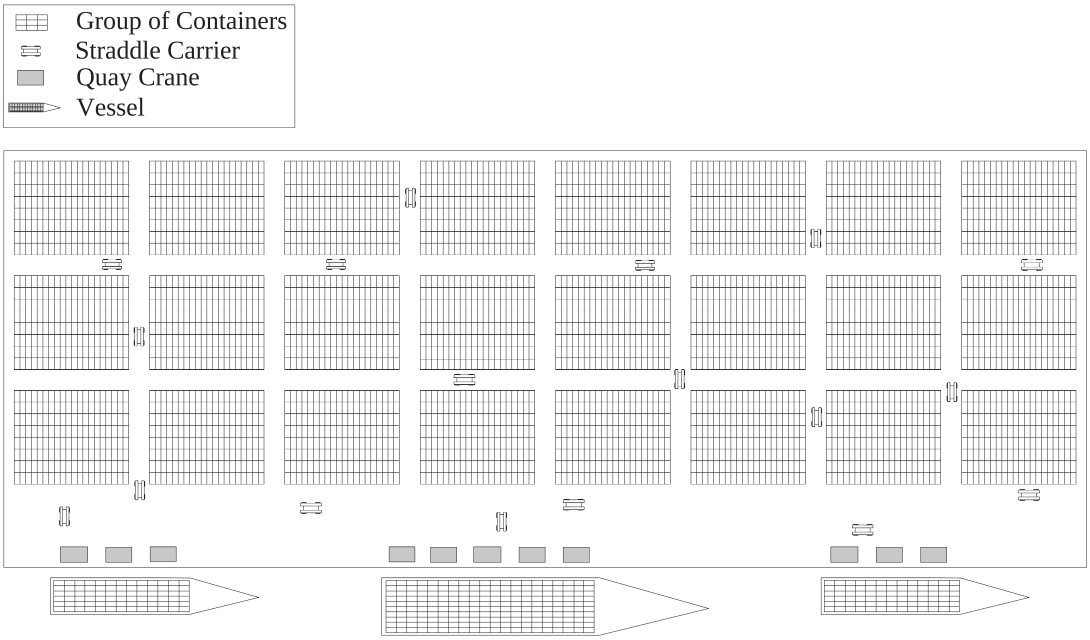


Figure 1.7: Logistics activities in a container terminal.

in the yard such that the total carrier travel distance is minimized. A reduction in this travel distance would not only lead to a smaller fuel consumption of the carriers, but also to a possible decrease in the idle time of the quay cranes and consequently to a certain amount of spare crane capacity.

For computational complexity reasons, this problem in itself is split into two subproblems, being the vessels' berth position problem and the container area position problem. For both these problems, we formulate independent mathematical optimization problems, which are coupled in the objective function that minimizes the total carrier travel distance. Due to this structure, the problem can be efficiently solved in an alternating fashion: we start solving the berth position problem for chosen initial values of the container areas positions. Next, the generated values of the berth positions are passed to the container area position problem and fixed as parameters. Subsequently, the container area position problem is solved and generated values are passed back as parameters to the berth position problem. This procedure is repeated until the objective value does no longer decrease. Since this alternating method finds a local minimum and heavily depends on the initial condition, an additional optimization model is proposed that finds a proper initial condition. Starting from this initial condition, the alternating procedure finds a local minimum that outperforms the best of all solutions found by starting from an extensive number of random initial conditions.

Results of a case study on a representative timetable of one of Antwerp's terminals depict vessels' positions and a yard lay out that induce a substantially smaller travel distance than the one as currently applied. Although a feasible solution in the second step optimization might turn out to be infeasible at this third level, not one of many experimental instances for typical quay utilizations in the world's busiest terminals reveals this problem. Although many studies on the individual berth allocation problem and the individual yard lay-out design problem can be found, ([15] and [33], respectively), a study that addresses the joint problem has, to the best of our knowledge, not yet been conducted.

Step 4: Online reallocation of a terminal under disturbances

The results of the former three steps together generate i) a robust timetable, ii) vessels' berth positions and iii) container positions in the yard. Such a tactical cyclic allocation is commonly reconsidered on a medium term time scale. However, as mentioned before, a container operation is exposed to all kinds of disturbances, which requires an online management system that observes the disturbances and reacts to them by reallocating the affected vessels and resources.

In the fourth optimization step, an approach is proposed to act upon disturbances on vessels' arrivals, changes in load compositions and break-downs of quay cranes. The approach is similar to the one constructed for the operational routing of product flows through the distribution network: In each iteration (time) step of the approach, (expected) parameter realizations over a limited future horizon are taken into consideration while determining the operational decisions for the current time step.

Each iteration step in itself consists of three sequential steps. In the first step, the expected arrivals and the expected call sizes of all vessels within the horizon are considered. Crane capacities are allocated such that vessels expected to arrive within their windows depart in time and vessels expected to arrive outside their windows are processed as fast as possible (without spending too much additional resources). Once the start and end berth times of each vessel within the horizon have been determined, berth positions at the quay are allocated in the second step, considering the call sizes and call compositions and the position of the containers in the yard. In the third step, the actual quay crane scheduling is performed for all vessels in a more limited future time horizon. After solving these three steps sequentially, the operational decisions of only the current time step are actually executed. Then, the sequence of the three optimization steps is performed for the next time step.

Since each iteration step can be solved within a couple of minutes, this approach is very suitable for practical setting. Namely, in practice, an operational plan is typically updated each hour. Simulation experiments are performed to trade off the carrier travel distance against the deviations from preferred berthing positions. Namely, dependent on the actual load compositions of the vessels, the optimal berth position (optimal in the sense that the carrier travel distance is minimum) of a vessel might deviate from the one derived in the tactical timetable.

Several studies consider the dynamic berth allocation problem in which vessels arrive in a terminal while work is in progress (see [45] and [15] for overviews). A rolling horizon approach like in this study, that observes and reacts on stochastic arrivals, stochastic load compositions as well as crane break downs however cannot be found so far.

1.5 Outline

The outline of the dissertation is as follows: In the chapters 2 through 6, the distribution network problem and the four main steps for the multi-terminal container operations are dealt with. For all of these chapters, the structure is the same. Firstly, a detailed literature review is given and the considered problem is positioned within existing studies. Secondly, the problem is formally phrased, and assumptions and model parameters are properly arranged. Third, a solution approach is proposed and mathematically formulated. Fourth, experiments on random instances or representative data are performed and results are discussed. Finally, conclusions and recommendations are given. In Chapter 7, the main findings from the various chapters are summarized and conclusions and drawbacks on the overall performance of the multi-step approaches in the two considered logistics networks are discussed.

Chapter 2

Design of a Distribution Network Topology

2.1 Introduction

The logistics networks considered in this chapter consist of production facilities, warehouses and consumers, which are geographically connected by links, e.g. roads, railways, waterways. In long-distance transportation networks, the distribution of products is often performed by a third-party logistics service provider (LSP). As opposed to the common supply chain studies, the problem addressed here is not to control the amount of products in the supply chain. Instead, this study addresses one of the typical services an LSP provides: supply and demand cannot be influenced by the LSP, but are simply a (stochastic) reality, that is revealed only a few days in advance. A supplier pushes its products from several production facilities to the logistics provider. By the same token, consumers pull products from the network. The logistics provider then has to decide whether to store the products in a warehouse or to immediately match them with a consumer demand.

The decisions regarding design and operation of distribution networks can typically be classified into three levels: The strategic level, the tactical level and the operational level. The strategic level deals with decisions regarding the number, location and capacities of warehouses. These decisions have a long-lasting effect on the system's performance. The tactical level includes decisions on which transportation links to actually use, i.e., the design of the network topology. To preserve continuity for employees and to restrict the complexity of organizational tasks such a network topology should not change too frequently. These tactical decisions are therefore reconsidered somewhere between once a month and once a year. The operational level refers to day-to-day decisions such as scheduling and routing of the shipments given the tactical topology.

In this chapter, we focus on the interplay of the operational level and the tactical level, i.e., we are interested to determine the topology of the network for a given operational strategy and given operational parameters. We specifically strive for a robust topology, which can be established for a relatively long period of time (months or years) and is still cost-effective when operational parameters (supply and demand distributions) change.

This research is supported by the LSP "Koninklijke Frans Maas Groep", one of the leading logistics service providers in Europe. One of the typical services this LSP provides can be illustrated by the following scenario: a big company produces different types of products at various production facilities throughout Europe. These products are consumed by all kinds of industries at various plants throughout Europe. Average consumption of each type of product by each consumer is known, but the daily demands fluctuate. On the other hand, due to production in batches, machine break down etc., also the supply of the different types of products fluctuates on a daily basis. Regardless of this, the consumers still expect, dependent on the type of product, a certain level of in-time delivery on a daily basis, where it does not matter from which production facility a consumer receives its products. Neither the consumers nor the supplier want to deal with the logistics of buffering the variability of supplies and demands and decide to hand this task to an LSP. Its task therefore is to supply the desired quantities at the desired day. The LSP can compensate for the stochasticity in supplies and demands by temporarily storing products in warehouses rather than shipping them directly from supplier to consumer. Furthermore, shipping via a warehouse enables product mixing so as to leverage on the economy of scales. On the other hand shipping through a warehouse introduces additional delay due to the handling activities, e.g. (un)loading and consolidation. The task of the LSP is therefore to decide which transportation links to use on the long run and how much of which product(s) to ship through them each day, such that total costs, including transportation and storage costs and penalties for early and late deliveries, are minimized.

In this chapter, we address the above described problem and determine the structure of the tactical network topology assuming the underlying operational control problem is approximately solved via a rolling horizon approach. We specifically strive for a network with a very small number of links that still has close to minimal operational cost. Reducing the number of links (or, equivalently, assuming unit cost per link) is a surrogate for a more detailed optimization at the tactical level for which the associated costs are hard to quantify:

- A reduction in the number of links reduces the complexity of the network and with that the complexity of the organizational tasks of an LSP,
- Each link involves a fixed cost due to contracts and overhead,
- Reducing the number of links leads to thicker flows per link.

The overall problem can thus be characterized as a bi-level bi-objective joint network design and operation problem. The upper (tactical) level chooses the links and the lower (operational) level, for given choice of links, chooses the material flow. Our goal is to provide an efficient method to approximate the trade-off front of link cost versus operational cost for a given time series of supply and demand, and in particular to determine a network topology that is (i) approximately optimal in both objectives (small number of links and minimal operational costs) and (ii) robust with respect to stochasticity (second and higher order moments) of supply and demand. An extensive amount of experiments is performed to confirm these properties.

2.1.1 Related Work

In [34], [9], [11], [25], [30], [52], [53], [2] networks for different applications are designed based on demand and supply. For instance in [34], properties of logistics networks structures are determined by optimizing the structural design on a strategic level: given the location of a plant producing one type of product and given the number and location of the consumers, who have to be satisfied, the number and locations of warehouses is determined. In [34] robustness is the extent to which the system is able to carry out its functions despite some damage done to it, such as the removal of some of the nodes and/or edges in the network. Results show that if the maximal robustness level is increased, more warehouses are present in the optimal network, decreasing the network efficiency and increasing its complexity. These studies all assume the deterministic problems (i.e. supply and demand are constant). In this chapter, however, we consider the distribution of multiple products and take stochastic supply and demand into consideration while designing a network for an LSP. Our definition of robustness is then to which extent the network topology is able to minimize operational costs despite some changes in supply and demand distributions.

A common setting for stochastic supply chain problems can be found in [48], [28], [6] [18] and [44]. These studies investigate different policies for supply chain management: orders are placed to manufacturing facilities to keep inventory positions in warehouses at a desired level and satisfy consumer demand. Only a restricted number ([18], [44]) mimic transportation costs by a stepwise cost-function dependent on the capacity of a transportation device (economy of scales). Such an approach reflects practice more accurately than a linear model at the expense of higher computational time. In this chapter, a heuristics is proposed that generates approximately similar close to optimal network topologies for both the stepwise and the linear model for medium size problems. For large network sizes, which become intractable with the stepwise model, we have to rely on the approach with linear transportation costs. Again, we have to stress that the decision space in a supply chain is crucially different from that of an LSP. Namely, in a supply chain, each member can place orders to the one upstream. For an LSP however, both supply and demand are given and cannot be influenced. Despite this difference, some of the supply chain studies mentioned above are related to ours as they use similar approaches on a tactical and/or strategic level.

In [48] for instance, the strategic design of a multi-echelon, multi-product supply chain network under demand uncertainty is studied. Given demand, decisions have to be made on the production amount at each supply facility. The objective is to minimize total costs taking infrastructure as well as operational costs into consideration. The authors consider a steady-state form of this problem and thus all flows between nodes are considered to be time-averaged quantities. A set of (only) three demand scenarios is generated and the objective function is expanded by adding weighted costs for meeting demand in each of these three scenarios. In this chapter, however, the amount of supply at the production facilities is given as well as the consumers' demands and cannot be controlled. Moreover, we do consider time-variant supply and demand of multiple products and use a rolling horizon approach to decide on the product flows on an operational level. Dependent on the link usage during a large number of time steps in this rolling horizon approach, a close to minimal number of links is established to construct a close to optimal topology. This topology is robust to changes in second and higher order moments of the supply and demand distributions.

In [28] the number and locations of transshipment hubs (strategic level) in a supply network is determined dependent on the product flows. The authors start from a network in which all potential hubs are present and minimize the costs for the operational activities in the network. Then the decrease in costs is evaluated for each of the cases where one hub is deleted from the network (establishing a hub introduces fixed costs). Next, the hub that causes the largest decrease is deleted from the network. This process is repeated and terminates whenever deleting a hub does not significantly affect the costs. In this chapter, a similar approach on a tactical level is applied: we start from a fully connected network, run the operational decision making policy for a certain time period (100 time steps) and in the end delete a least used link over this period. This procedure is repeated until the network is minimally connected. Additionally, experiments suggest that this heuristic generates a topology with a small number of links that is robust to changes in supply and demand distributions.

In [6], models and algorithms for a one product, multi-stage stochastic distribution problem with recourse are developed. In the first stage, the amount of products to be supplied by the plants to warehouses has to be decided on, without knowing the demand. Then in the second stage, the demand becomes available and products have to be shipped from the warehouses to the consumers. In this chapter, we consider the distribution of multiple products by an LSP and investigate the robustness of generated network topologies. Since supply and demand over a restricted time horizon in the future are known, we can apply a rolling horizon approach rather than a multi-stage approach. As we noticed from practice, such a rolling horizon approach is often used by an LSP.

To our knowledge, only a restricted number of studies consider the viewpoint of an LSP where supply and demand are both stochastic and uncontrollable. Of particular interest is the study in [47]: A company owns several production plants and has to distribute only one type of product to different regional markets. In each time period, a random (uncontrollable) amount of products becomes available at each of these plants. Before the random demand becomes available, the company has to decide which proportion of the products should be shipped directly and which proportion should be held at the production plants. Linear costs are assigned to transportation, holding and backlog. A look ahead mechanism is introduced by using approximations of the value function and improving these approximations using samples of the random quantities. It is numerically shown that this dynamic programming method yields high quality solutions. One of the results of [47] is that most improvements on the operational costs are made when a part of the consumers is served by two plants, rather than one. The study in [47] investigates the distribution of only one type of product for only one network configuration with linear transportation costs. In this chapter, however, we consider the distribution of multiple products and start from considering a stepwise cost function to represent the economy of scales principle. Hence, our methodology takes the possible consolidation of high and low value products to ship full trucks into consideration. Another difference is that in the setting we consider, the exact supplies and demands over a restricted future horizon are known and we thus can apply a rolling horizon approach. This is another major difference with the study in [47], where decisions on how much to send or store have to be made before the actual demand becomes available (recourse model). Furthermore, we expand our study by investigating the robustness of the constructed network topologies, i.e., the dependency of the operational performance of the constructed topologies on supply and demand distributions.

We apply the proposed bi-level optimization method to several real-life networks. The results suggest that in each considered case a distribution network with only a few links provides a large portion of the operational efficiency of the fully connected network. These results confirm the findings of other studies on different kinds of two-echelon networks. Results in [26] for instance suggest that, for a small theoretical problem, limited flexibility in manufacturing processes (i.e., each plant builds only a few products) yields most of the benefits of total flexibility (i.e., each plant builds all products). However, the authors mention that for more realistic cases they have no guidelines or general approach to add flexibility in an arbitrary network. In this chapter, this issue is addressed by developing a heuristic that for more realistic cases and for more than a two echelon system generates a network with a small number of links that still performs well. Moreover, the study in [26], does not investigate the robustness of the constructed networks to the system's parameters. In this chapter, besides proposing a generic design methodology, we also investigate robustness properties of the designed topologies.

2.1.2 Contributions

We proceed in Section 2.2 by first addressing the lower-level operational problem (Section 2.2.1) of controlling the material flow for a given network topology. A stepwise cost function is introduced in the flow model to represent the costs for transportation by trucks (economies of scale). The supplies and demands of each product type are scaled to the trucks' capacity. Storage costs per unit per day are (therefore) the same for each product type. Costs for early and late delivery do depend on the product type. A model predictive control with rolling horizon (MPC) [13] is used, which determines a sub-optimal operational routing schedule for a particular topology and particular supply and demand realizations over a certain time period (in this case 100 time steps). This amounts to solving a sequence of dependent non-linear minimum-cost network flow problems.

In Section 2.2.2, the upper level is addressed, which in turn refers to the tactical decision making. We present a branch-and-bound method that computes the exact trade-off front of the bi-objective problem for given multiple-product supply and demand time series and the given control policy. As the method is impractical for larger instances, we propose a heuristic that works by iteratively dropping least used links (after determining a routing schedule for a certain time period) to generate an approximation of the trade-off front. The heuristic is accurate and much faster than the branch and bound, however still large real-life networks instances are intractable due to the model's non-linearity. Hence, we replace the stepwise cost model by a linear cost model. Since in the heuristic the link usage depends on the product type(s) through it (the higher the demand for just-in-time delivery, the higher the weight factor), links with products that do not require just-in-time delivery will be dropped first. In the end, these products are then shipped through another link together with products that do require just in time delivery. In this way, different product types are consolidated even in the linear approach. We compare the results to those obtained with the stepwise cost model and find similar results in both the costs curve and the topology structure. For the heuristic approximation, which is very efficient, we can usually provide a network with a small number of links and close to minimal operational costs. From that we can conclude that the topology is also close to optimal.

Results of the branch and bound method and the heuristics approach for small problems are compared in Section 2.3. Next, the results for the heuristics applied to the stepwise and the linear model for medium size problems are compared. Moreover, results of the heuristics approach applied to the linear model for several large network configurations are shown in Section 2.3. These results suggest the validity of the proposed approximative bi-level optimization. Finally, we perform a large amount of experiments to formulate generic insights in the robustness of the heuristic network topology. Since this heuristic network topology results from a particular supply and demand time series, we are interested in the performance of this topology for different time series. Interestingly, the experimental results suggest that the heuristic network topology is only sensitive to the first moment of supply and demand distributions. Namely, as long as the heuristic network topology is run operationally for a different supply and demand time series with the same means, the operational costs are still close to minimal (relative to the operational costs in the fully connected network). However, if the means are changed, the operational costs grow large (relative to the operational costs in the fully connected network). Having reasonable forecasts about the individual means of supply and demand we would a priori be able to construct a cost-effective network topology, robust to any time series with these means. We end with conclusions and recommendations for future work in Section 2.4.

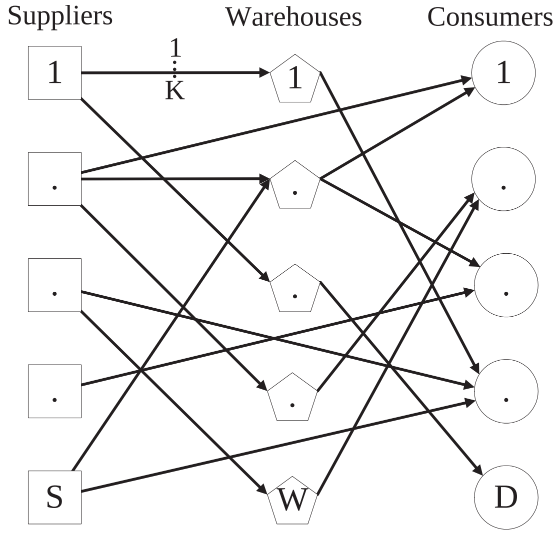


Figure 2.1: Illustration of the considered logistics networks.

2.2 Bi-level Network Design Problem

We consider the distribution of K types of products by trucks through a network with S production facilities, W warehouses, and D consumers. Unless stated differently, we use the indices $s \in \{1, ..., S\}$ for the production facilities, $w \in \{1, ..., W\}$ for the warehouses, $d \in \{1, ..., D\}$ for the consumers, and $k \in \{1, ..., K\}$, the product types. Products can be sent either directly from supplier(s) to consumer(s) or indirectly via a warehouse, assuming that each consumer can receive its products from each (and more than one) of the production facilities. Shipments from one warehouse to another are not taken into account, but could easily be incorporated into this framework as well. An example network for this type of three-echelon multi-item distribution system is depicted in Figure 2.1. Even though each link in the graph has a certain length representing the geographical distance between the corresponding nodes, Figure 2.1 does not show the spatial positioning of the nodes for ease of presentation.

Each production facility supplies one or multiple product types, and each consumer demands one or multiple product types, at each time step (day) $t \in \{1, ..., T\}$. The supply and demand time series $\mathcal{S}_i^k(t)$ (supply of product type k by supplier i at day t) and $\mathcal{D}_j^k(t)$ (demand of product type k by consumer j at day t) are randomly distributed according to a joint distribution that is defined by the following sampling procedure: for each combination of product type, supplier and consumer, we assume a uniform distribution on an interval $[\bar{\sigma}_i^k(t) - \hat{\sigma}_i^k(t), \bar{\sigma}_i^k(t) + \hat{\sigma}_i^k(t)]$ with given mean $\bar{\sigma}_i^k(t)$ and half-length $\hat{\sigma}_i^k(t)$. Next, T preliminary samples out of each of the $K \cdot (S + D)$ distributions are taken. The final samples are then determined by adding a constant either to each supply or each demand sample for each product type to enforce that for each product type, the sum of T supplies equals the sum of T demands over the horizon.

On a given network, the operational task of the LSP is to decide the amount of products to be transported over each link, which incurs transportation cost on the links, storage cost in the warehouses, and penalty costs for early or late delivery at the consumers. On the tactical level, the task of the LSP is to choose a subset of links such that it can carry out its operational task well, e.g., with minimal expected costs. Of course, there is an immediate trade-off between the number of links and the operational costs, as additional links can only improve the operational costs due to the added flexibility. In the following, we present an approach to approximate the trade-off front of this bi-objective bi-level problem numerically, and in particular to determine a network structure with a close to minimal number of links and close to minimal operational costs.

2.2.1 Operational Decision Making

To formalize the operational task of the LSP, we define the operational decision variables $x_{ij}^k(t) \geq 0$ as the amount of products of type k transported from node i to node j on time step t, where $(i, j) \in \mathcal{U}$, the set links in the given network. On the suppliers side, we require that all available products $\mathcal{S}_i^k(t)$ have to picked up in the same period such that

$$\mathcal{S}_i^k(t) = \sum_{j \in P_i} x_{ij}^k(t) \qquad \forall k, i, t \tag{2.1}$$

where the set P_i contains the indices of all the nodes to which production facility i is connected. On the consumers side, both early and late deliveries might occur and have to be accounted for. For this we introduce state variables $b_j^k(t)$ as the backlog of product type k for consumer j on day t, whose dynamics is

$$b_j^k(t) = b_j^k(t-1) + \mathcal{D}_j^k(t) - \sum_{i \in Q_j} x_{ij}^k(t) \qquad \forall k, j \tag{2.2}$$

where the set Q_j contains the indices of all the nodes connected to consumer j. In the warehouses, several processes (e.g. inbound, consolidation) between arriving at and storing in a warehouse cause delay. The total delay is captured in a constant $\tau_w \in \mathbb{N}$ so that products arriving in warehouse w at time t can be shipped out τ_w time steps later at the earliest. To model this time delay, we introduce additional state variables $y_w^k(t)$ as the inventory of product k in warehouse w on day t, whose dynamics is

$$y_w^k(t) = y_w^k(t-1) + \sum_{i \in I_w} x_{iw}^k(t - \tau_w) - \sum_{j \in O_w} x_{wj}^k(t) \qquad \forall k, w, t \tag{2.3}$$

where the set I_w contains the indices of all nodes that are sources of warehouse w and the set O_w contains the indices of all nodes that are destinations of warehouse w.

The operational cost for time step t is the sum of transportation cost, storage cost, and backlog cost. It can be expressed as a function of the decision variables $x_{ij}^k(t)$ and state variables $b_j^k(t)$ and $y_w^k(t)$ at time t as

$$\alpha(x_{ij}^k(t), b_j^k(t), y_w^k(t)) \equiv \sum_{(i,j) \in \mathcal{U}} c_{ij} \phi(\sum_{k=1}^{K} x_{ij}^k(t)) + \sum_{k=1}^{K} \sum_{j=1}^{D} \beta^k {b_j^k}^2(t) + \sum_{w \in \mathcal{W}} h_w \sum_{k=1}^{K} y_w^k(t) \tag{2.4}$$

where c_{ij} represents the geographical distance between nodes i and j, h_w represents the inventory holding costs in warehouse w, and β^k represents the "value" of product k for consumer j. The reason for not using the absolute value $|b_j^k(t)|$ as a penalty for early or late delivery is that in case of shortages, the quadratic function favors equal distribution of shipments among consumers rather than shipping everything to one consumer. The function $\phi(\cdot)$ expresses quantity-dependent transportation costs. We consider two cases, (i) $\phi(x) \equiv \lceil \frac{x}{V} \rceil$ to express stepwise-constant transportation costs due to the use of trucks with unit capacity V, and (ii) the identity $\phi(x) \equiv x$ to express the simplest case of linear transportation costs.

We further assume that suppliers and consumers commit to their real supplies and demands for a constant number of time steps Ω in advance. Thus, the exact supplies and demands for the current and the next $\Omega - 1$ time steps are known and can be taken into account when deciding the transportation quantities. By using state augmentation we can include this information in the current state of the distribution system at time t. Consequently, the operational problem can be posed as a discrete-time stochastic optimal control problem, with the objective to minimize the expected operational cost over the considered fixed time horizon T. Under the assumptions given above it is reasonable to assume that there exists an optimal state-feedback control policy $\{\mu^*(t)\}_{t=1}^T$ that maps the augmented state to its corresponding optimal transportation quantities at each time step t.

The standard method to solve this type of stochastic optimal control problem is stochastic dynamic programming. Unfortunately, the state space dimension in our case is very large, so that a numerical solution, even with a very coarse state discretization, seems intractable. We therefore have to resort to solving the operational control problem only approximately.

One principle method for approximate dynamic programming is to use a limited look-ahead scheme in a rolling horizon fashion, also called model predictive control (MPC). The basic idea is that optimization is only performed over a limited horizon where more information about uncertain data is available, or even well representable by deterministic or nominal values, and to choose some reasonable approximation of the value function for the time steps outside the horizon. Here, we use a horizon of Ω steps as the exact supply and demand data is known over this time frame, and approximate the value function of the state at the horizon with the constant zero. Thus, we obtain an approximate control policy $\hat{\mu}$ by solving the following time-expanded network flow problem:

$$\text{minimize} \sum_{q=t}^{t+\Omega-1} \left[\sum_{(i,j)\in\,\mathcal{U}} c_{ij}u_{ij}(q)) + \sum_{k=1}^{K}\sum_{j=1}^{D} \beta^k [b_j^k(q)]^2 + \sum_{w\in\,\mathcal{W}} h_w \sum_{k=1}^{K} y_w^k(q) \right] \tag{2.5}$$

subject to

$$\begin{array}{lcll}
\mathcal{S}_i^k(q) & = & \sum\limits_{j\in\,P_i} x_{ij}^k(q) & \forall k,i,q \\
y_w^k(q) & = & y_w^k(q-1) + \sum\limits_{i\in\,I_w} x_{iw}^k(q-\tau_w) - \sum\limits_{j\in\,O_w} x_{wj}^k(q) & \forall k,w,q \\
b_j^k(q) & = & b_j^k(q-1) + \mathcal{D}_j^k(q) - \sum_{i\in\,Q_j} x_{ij}^k(q) & \forall k,i,j,q \\
V \cdot u_{ij}(q) & \geq & \sum\limits_{k=1}^{K} x_{ij}^k(q) & \forall k,i,j,q \\
x_{ij}^k(q) & \geq & 0 & \forall k,i,j,q \\
y_w^k(q) & \geq & 0 & \forall k,w,q
\end{array}$$

Minimization is performed over the variables $x_{ij}^k(q)$, $u_{ij}(q)$, $b_j^k(q)$, and $y_w^k(q)$, where $q \in \{t,...,t+\Omega-1\}$. The additional variables $u_{ij}(q)$ are introduced to linearize the transportation cost function $\phi(\cdot)$ and represent the number of trucks of capacity V necessary to transport the total amount of products over link (i,j) at time step q. To model the stepwise constant transportation cost, these variables have to be restricted to integer values; otherwise a continuous relaxation can be used to model linear transportation costs.

For all time steps prior to t, the values of the state variables and decision variables are known so that they serve as deterministic data in the above limited look-ahead problem. The same holds for the values of supplies $\mathcal{S}_i^k(q)$ and demands $\mathcal{D}_j^k(q)$ within the considered horizon $\{t,...,t+\Omega-1\}$. Consequently, the problem is a mixed-integer quadratic program (in case of stepwise-constant transportation costs) or a quadratic program (in case of linear transportation costs). The approximate policy $\hat{\mu}$ is now given by the mapping of the current (augmented) state at time t to the values $x_{ij}^k(t)$ in an optimal solution of the above optimization problem.

2.2.2 Tactical Decision Making

In a given collection of suppliers, warehouses, and consumers, let $\mathcal{L}$ be the union of all potential links (directed edges) from suppliers to warehouses, warehouses to consumers, and suppliers to consumers. The tactical problem can then be defined as choosing a subset $\mathcal{U} \subseteq \mathcal{L}$ of a given cardinality L with smallest expected operational cost. In doing so, we tacitly assume that there exists some mapping γ_μ that assigns the expected operational costs $\gamma_\mu(\mathcal{U})$ to the network of chosen links $\mathcal{U}$ when using a particular control policy μ. As it is unclear how this expectation can be computed exactly in the present set-up, we resort to a variant of sample average approximation by Monte Carlo simulation, where we simulate the process for a fixed number of time steps while applying the given policy μ. Here, we use the the approximate limited look-ahead policy $\hat{\mu}$ defined above, and refer to the resulting estimator of the expected operational cost as $\hat{\gamma}_{\hat{\mu}}(\mathcal{U})$.

In a bi-objective formulation, this tactical problem is to be solved for any value of $L \in \{L_{\min}, \ldots, |\mathcal{L}|\}$, where $L_{\min}$ is the smallest number of links such that no node is isolated and each warehouse is connected to at least one supplier and one consumer. Our conjecture about this trade-off between the number of chosen links and the operational costs is that the operational costs are only marginally sensitive to a reduction of the number of links from the fully connected network until a critical value, after which the costs increase sharply. In this case, the trade-off frontier would form a pronounced knee. Hence, our particular goal is to determine a representative network from this region, as this would constitute in a sense a best-possible bi-objective approximation. Such a network could be considered "cost-effective" as it would yield close-to-minimal operational costs with a close-to-minimal number of links.

To determine the trade-off frontier, we first develop a bi-objective branch and bound method that computes the minimal cost topology for any given number of links simultaneously. As this method still searches a large part of the solution tree, we additionally propose a heuristic, which is able to generate a satisfactory solution for larger instances.

Bi-objective Branch and Bound Method

We develop a branch and bound algorithm that defines a tree search over a set of binary decision variables z_{ij} associated with each link $(i, j) \in \mathcal{L}$, where

$$z_{ij} = \begin{cases} 1 & \text{if the link from facility } i \text{ to facility } j \text{ is present,} \\ 0 & \text{otherwise.} \end{cases} \tag{2.6}$$

A pseudo-code description of the algorithm is given below (Algorithm 1). The algorithm maintains and updates a current approximation to the trade-off frontier whose initial values are given in Table 2.1.

L	$L_{\min}$	$L_{\min}+1$	.	.	.	.	$\mathcal{L}$
opt(L)	∞	∞	∞	∞	∞	∞	∞
$Z_{\text{opt}}(L)$	-	-	-	-	-	-	(1,1,1,...,1)

Table 2.1: Initial state of the table for the minimal costs opt(L) and the corresponding solution (network) $Z_{\text{opt}}(L)$.

During the search, the L-entry is updated whenever a network z with L links is identified to have lower operational cost than the current best network with L links (lines 14-16). At each branching operation (lines 19-27), a natural lower bound is given by setting all undecided variables to one as all solutions in this subtree can only be composed of a subset of those links. The subtree can be pruned if this lower bound is not lower than the current best value for any number of links L for which there are potential solutions in the subtree; otherwise, two children are created by setting the first undecided variable to zero and one, respectively, and added to the list on unexplored nodes A.

Heuristic

Since the branch and bound method evaluates a large number of possible topologies, only small problems can be solved. Hence, we propose a heuristic that evaluates less than $|\mathcal{L}|$ topologies and still yields a very good solution. The heuristic starts from the fully connected network and approximates the operational costs $\hat{\gamma}_{\hat{\mu}}(\mathcal{L})$ when using the control policy $\hat{\mu}$ over the fixed number of time steps. Then the link(s) that are least used over this time period are deleted. In determining the usage of a link, the value of the product type(s) through this link is taken into account by a weight factor. We define the weighted average daily usage P_{ij} of the link between facilities i and j as

$$P_{ij} = \frac{1}{T} \sum_{k=1}^{K} \sum_{t=1}^{T} \frac{b_k}{\max_{b_k}\{b_k\}} \cdot x_{ij}^k(t). \tag{2.7}$$

In this way, links that provide just-in-time delivery of high value products are not (easily) deleted. This procedure of deleting links is repeated until a minimally connected network remains. A pseudo-code description of the algorithm is given below (Algorithm 2).

Algorithm 1 Branch and Bound

let $z := (\star, \star, ..., \star)$
let $\mathrm{LB}(z) := -1$
initialize the list of unexplored nodes A with z
while A is not empty **do**
  pick z as the last element from A
  delete z from A
  let p be the number of ones in z and q be the number of zeros in z
  if $p + q = |\mathcal{L}|$ **then**
    if $\mathrm{LB}(z) = -1$ **then**
      let $c := \hat{\gamma}_{\hat{\mu}}(z)$
    else
      let $c := \mathrm{LB}(z)$
    end if
    if $c < \mathrm{opt}(p)$ **then**
      let $\mathrm{opt}(p) := c$
      let $Z_{\mathrm{opt}}(p) := z$
    end if
  else
    if $\mathrm{LB}(z) = -1$ **then**
      let z' be a copy of z where all $\star$ are replaced by 1
      let $\mathrm{LB}(z) := \hat{\gamma}_{\hat{\mu}}(z')$
    end if
    if $\mathrm{LB}(z) < Z_{\mathrm{opt}}(L)$ for some L with $p \leq L \leq |\mathcal{L}|$ **then**
      let z'' be a copy of z where the first $\star$ is replaced by 0 and append z'' to A
      let z''' be a copy of z where the first $\star$ is replaced by 1 and append z''' to A
      let $\mathrm{LB}(z'') := -1$ and $\mathrm{LB}(z''') := \mathrm{LB}(z)$
    end if
  end if
end while

Algorithm 2 Heuristic

1: let $\mathcal{U}$ be the fully connected network $\mathcal{L}$
2: **while** the network is not minimally connected **do**
3: apply the policy $\hat{\mu}$ during T time steps and determine $\hat{\gamma}_{\hat{\mu}}(\mathcal{U})$
4: determine usage P_{ij} for each link in $\mathcal{U}$
5: delete the link(s) from $\mathcal{U}$ with minimal usage
6: **end while**

2.3 Computational Study

In the previous section, we proposed two algorithms to find a network topology with a close to minimal number of links that is still cost-effective on the operational level. In this section, the performance of both these algorithms is evaluated and compared. Experiments suggest that the heuristic is very effective in constructing a network topology with a small number of links that yields close to minimal operational costs. However, large real-life problems are still intractable. Hence, we replace the stepwise cost function by a linear cost function and find similar results much faster. Some large real-life instances are solved for the linear cost structure and results are presented.

To evaluate the performance of the heuristic, a bi-objective relative approximation factor (ϵ_c, ϵ_L) is used as a performance measure. The factor denotes the relative deviation from the ideal point, the point composed of the single-objective optima. In this case, those values are known, so we can define the two components as

$$\epsilon_c(\mathcal{U}) := \frac{\hat{\gamma}_{\hat{\mu}}(\mathcal{U})}{\hat{\gamma}_{\hat{\mu}}(\mathcal{L})} \qquad \text{and} \qquad \epsilon_L(\mathcal{U}) := \frac{|\mathcal{U}|}{L_{\min}}, \tag{2.8}$$

denoting the operational costs of the considered topology $\mathcal{U}$ relative to the operational costs in the fully connected network, and the number of links $|\mathcal{U}|$ relative to the number of links in a minimally connected network. Even though we are not able to give some a priori performance guarantee of the heuristic, this factor can be used to bound the approximation quality a posteriori.

In addition to evaluating the performance of the heuristic approach, a computational study is performed to investigate the robustness of the so found network topology with respect to changes in the supply and demand distributions. Results suggest that a "heuristic network" with a satisfactory $(\epsilon_c(\mathcal{U}), \epsilon_L(\mathcal{U}))$ value is cost-effective as long as the means of supply and demand distributions remain the same. Finally, we formulate the hypothesis that increasing the initial inventory enables to compensate for the backlog costs embedded in a particular supply and demand time series. The hypothesis is confirmed by experimental results.

For the logistics networks in this chapter, we consider M suppliers, W warehouses, D consumers and the distribution of K product types. The networks considered typically consist of few suppliers and warehouses and a lot of consumers, such that $W < M \ll D$. Furthermore, we generate stochastic time series of $K \cdot S$ supplies and $K \cdot D$ demands from uniform distributions with random means and variances for 100 time steps. We assume that supplies and demands are only known 2 days in advance, so the window size Ω is equal to 3. Moreover, we focus on networks of relatively small geographical extent. Therefore, it is assumed that products can be transported from any arbitrary node to another within one day. In addition, we assume that the delay for going through warehouse n is equal to one day, which implies that $\tau_w = 1$. The costs are chosen such that the proportion of transportation and storage costs $c_{i,j} : h_n$ =1 : 0.6 on average and the early/late delivery costs b^k are randomly chosen between the bounds of 0.001 and 100.

2.3.1 Comparison Branch and Bound versus Heuristic

First, we compare the results of the branch and bound method and the results of the heuristic applied to the stepwise cost model. Hence, we generate different problem instances (supply and demand distributions as well as spatial location of network nodes) for many small network configurations and run the branch and bound method as well as the heuristic. Comparison of the resulting trade-off fronts of operational cost versus number of links show that the heuristic generates a topology that is close to optimal.

As an illustration (for a network with $S = 2$, $W = 1$, $D = 3$ and $K = 1$), we show results for ten different parameter sets being 10 supply and demand time series, each accompanied by a specific spatial location of nodes. Results for these sets for both the branch and bound method and the heuristic are shown in Figure 2.2. As expected, both methods generate equal operational costs for the fully connected network (number of deleted links equals zero). From Figure 2.2, it also appears that the critical value in the efficient frontier found by the heuristic occurs at 5 or 6 deleted links, whereas it always occurs at 6 deleted links for the networks generated by the branch and bound approach. The difference in the number of links in the resulting network for these instances is thus at most 1 link. This suggests that the heuristic finds a satisfactory solution for small network instances. Since the CPU time of the branch and bound grows extremely fast with increasing network size, we have to rely on the heuristic for larger instances.

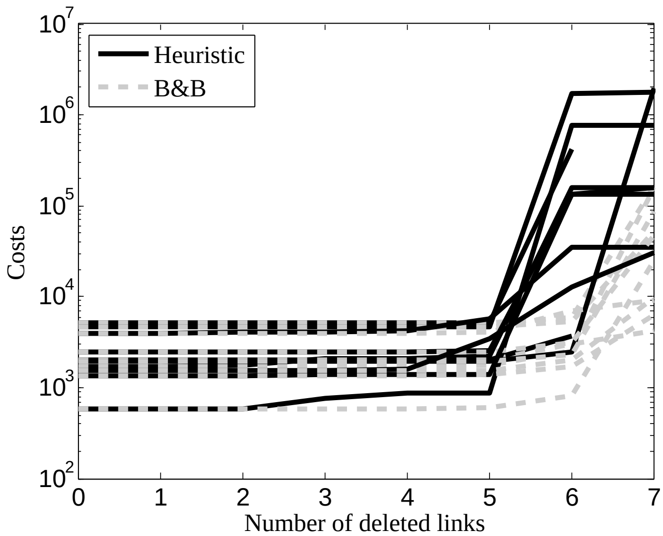


Figure 2.2: Comparison of branch and bound with heuristic for 10 supply and demand instances, $S = 2$, $W = 1$, $D = 3$, $K = 1$, Branch and bound: $\epsilon_c \leq 1.01$, $\epsilon_L = 1.5$, Heuristics: $\epsilon_c \leq 1.01$, $\epsilon_L = 1.63$.

The heuristic, applied to the stepwise cost model, enables to find a close to optimal topology for a network with up to 4 suppliers, 3 warehouses, 25 consumers and 6 product types ($M = 4$, $W = 3$, $C = 25$ and $K = 6$, respectively). Due to the non-linearity, larger real-life problems are still intractable. We therefore remove the integer variables

u_{ij} from the model and introduce linear transportation costs ($\phi(x) \equiv x$). We apply the heuristic to this linear model, which is now able to construct cost-effective topologies for large real-life network sizes. In the next section, results of the heuristic for the stepwise and linear model are compared. Additionally, we present results of the heuristic for the linear model for large real-life network sizes.

2.3.2 Performance of the Heuristic for Large Networks

Results for both the linear and the stepwise model are presented in Figure 2.3 for two different network configurations, respectively. For larger network sizes, we have to rely on the heuristic model with linear costs. Two cost curves for large real-life networks are presented in Figure 2.4. Costs are plotted on a logarithmic scale versus the number of deleted links. In these plots, zero deleted links corresponds to the fully connected network, whereas the maximal number of deleted links corresponds to a minimally connected network.

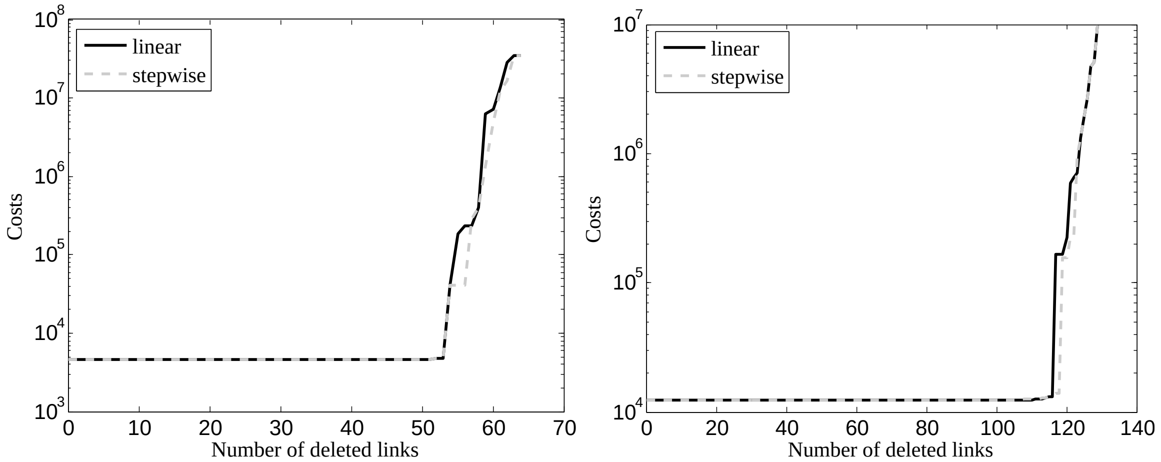
All these curves (in Figures 2.3 and 2.4) exhibit the same typical characteristics from the left to the right: at first, a large number of links can be deleted without affecting the costs significantly: Up to 70% of the links is not used at all in the first heuristic iteration and is thus deleted at once. Still, about 15% of all the links can subsequently be deleted without substantially affecting costs. Then, after about 85% of the total number of links that can be deleted, costs start increasing slightly with the number of deleted links. Finally, when about 90-95% of the total number of links that can be deleted, costs start increasing dramatically. Below the figures the ϵ_L-value are given for an upper bound on the ϵ_c-value of 1.01.

The comparison between the results of the linear and stepwise model show that in the fully connected network, as expected, the costs for the stepwise model are slightly higher than the costs for the linear model (due to the logarithmic scale this can hardly be noticed in Figure 2.3). Furthermore, Figure 2.3 shows that for both models the dramatic cost increase starts at approximately the same number of links. Further analysis (not depicted in the figures here) shows that for the performed experiments, the set of deleted links at the critical value is approximately the same for both models. A large number of experiments, not presented in this chapter, exhibit the same characteristics, suggesting that the heuristic for the linear model is a quite good approximation of the heuristic with the stepwise model.

Apparently, a topology with only a small number of links suffices to gain close to minimal operational costs. Since only a small number of links remain to be deleted we can also conclude that the resulting topology is also close to optimal. In [26], similar characteristics are presented for a manufacturing system with deterministic supplies and demands. The results in [26] suggest that a small flexibility (each plant produces only a couple of product types) can almost achieve the benefits of total flexibility (each plant produces all product types). Furthermore, it should be noticed that the curves in Figures 2.3 and 2.4 exhibit approximately horizontal plateaus. The reason for this is that not one, but a group of links has to be deleted to significantly affect the costs.

2.3.3 Statistical Robustness Analysis

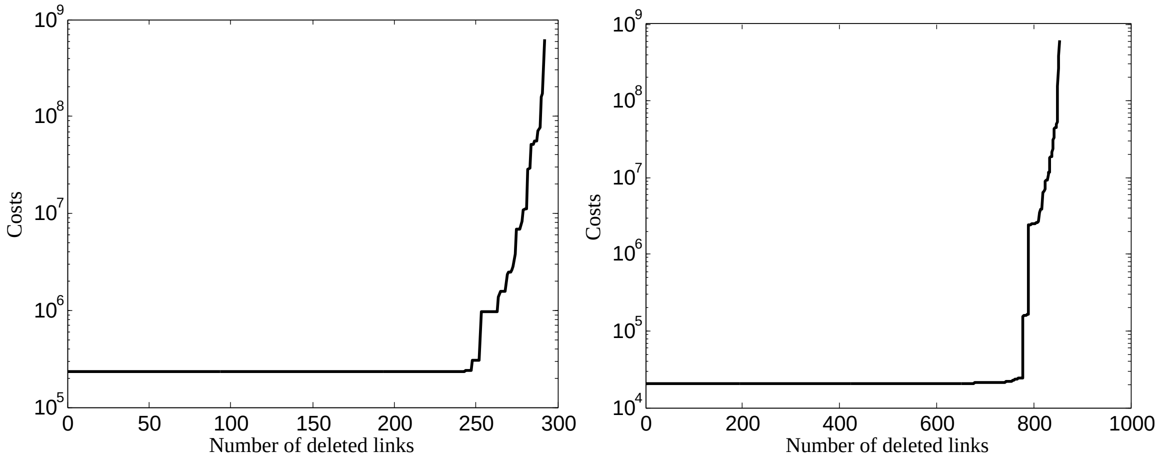
We now formulate the hypothesis that networks generated for a particular supply and demand time series, are insensitive to changes of second or higher order moments of



(a) $S = 3$, $W = 2$, $D = 15$, $K = 5$, $\epsilon_c \leq 1.01$, $\epsilon_L = 1.65$.

(b) $S = 4$, $W = 3$, $D = 20$, $K = 5$, $\epsilon_c \leq 1.01$, $\epsilon_L = 1.57$.

Figure 2.3: Comparison heuristics results for the linear and stepwise model, T=100, τ_w=1, Ω=3.



(a) $S = 5$, $W = 3$, $D = 40$, $K = 6$, $\epsilon_c \leq 1.01$, $\epsilon_L = 2.35$.

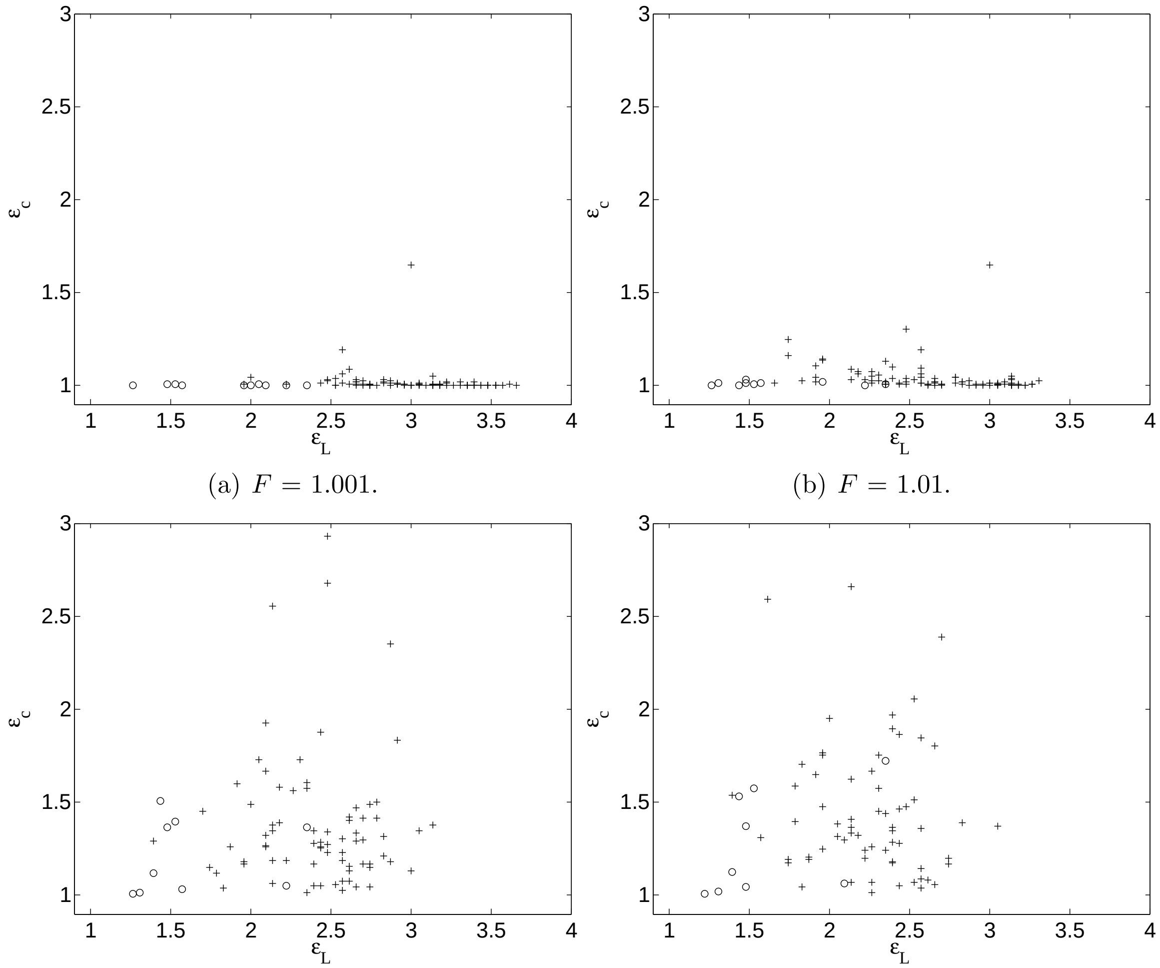
(b) $S = 8$, $W = 4$, $D = 75$, $K = 10$, $\epsilon_c \leq 1.01$, $\epsilon_L = 1.97$.

Figure 2.4: Heuristics results for the linear model, T=100, τ_w=1, Ω=3.

supply and demand distributions. Thus, if information about the individual means of supplies and demands is available a priori, we only need to generate a stochastic time series with these means to find a cost-effective topology for other scenarios with these means. Next, as we optimize over a finite period of time T, we have to address the influence of the initial inventory. We demonstrate that the steady state value of the costs converges to the minimal transportation plus storage costs, as initial inventory increases.

Sensitivity

To allow for an automatic statistical sensitivity analysis we define a representative "heuristic network" to be the smallest network $\mathcal{U}$ discovered by the heuristic, whose operational costs are within a factor F of the operational costs of the fully connected network, i.e., where $\epsilon_c(\mathcal{U}) \leq F$. Deleting one more link will increase the costs above that level. We then perform the following statistical analysis: We choose fixed mean supplies and demands for a network with $S = 4$, $W = 3$, $D = 20$ and $K = 5$ and generate stochastic supply and demand samples (for 100 time steps). Based on this, we determine the "heuristic network" and fix its topology, i.e., its link structure $\mathcal{U}$. For this network $\mathcal{U}$ and the same product mix, we than compute the operational costs for supply and demand time series from 20 different distributions (with different second and higher order moments but same means) and record the resulting approximation factor $\epsilon_c(\mathcal{U})$ in each case. This experiment is repeated for 99 other heuristic networks, each generated from particular supply and demand time series.



(a) $F = 1.001$.

(b) $F = 1.01$.

(c) $F = 1.05$.

(d) $F = 1.10$.

Figure 2.5: Sensitivity "heuristic network" to 2nd and higher order moments. 'o' product mix with 1 high value and 4 low value products, '+' product mix with more than 1 high value product.

Figure 2.5 shows four scatter plots (each for a particular value of F) of the 100 mean values over the 20 ϵ_c samples versus ϵ_L. As expected, the robustness decreases as F increases (since then the number of links in the heuristic network decreases which makes the network more sensitive). We define the network to be robust if the mean value of ϵ_c (over 20 experiments) is at most 1.2 with a frequency of 95%. The scatter plot in Figure 2.5a shows that ϵ_c is strongly clustered near one indicating that, with high

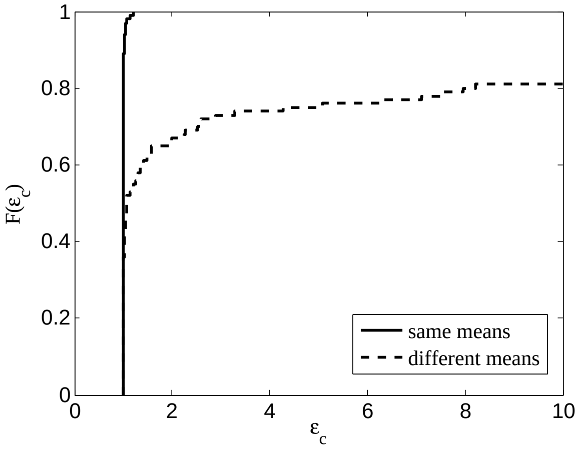


Figure 2.6: Empirical cumulative distributions showing the sensitivity of the heuristic networks to changes in i) variances and ii) means of supplies and demands.

likelihood, our process of choosing a heuristic network (with $F = 1.001$) will find a network that is a "very good" network even when the supply and demand time series vary randomly with respect to higher order moments. The histogram for the case where $F = 1.01$ (Figure 2.5b) is still satisfying since more than 95% of the considered cases results in an ϵ_c value equal to or smaller than 1.2. The results for $F = 1.05$ (Figure 2.5c) and $F = 1.1$ (Figure 2.5c) do not satisfy our definition of a robust network anymore. For these values of F, several means and standard deviations of ϵ_c are even outside the figure range (larger than 3). We choose as "heuristic network" the one with $F = 1.01$, since (i) the accompanying results fulfill our definition of a robust network, and (ii) the generated heuristic networks have less links than the case where $F = 1.001$. These results suggest that the "heuristic network" ($F = 1.01$) indeed is robust to changes in second and higher order moments of supply and demand distributions.

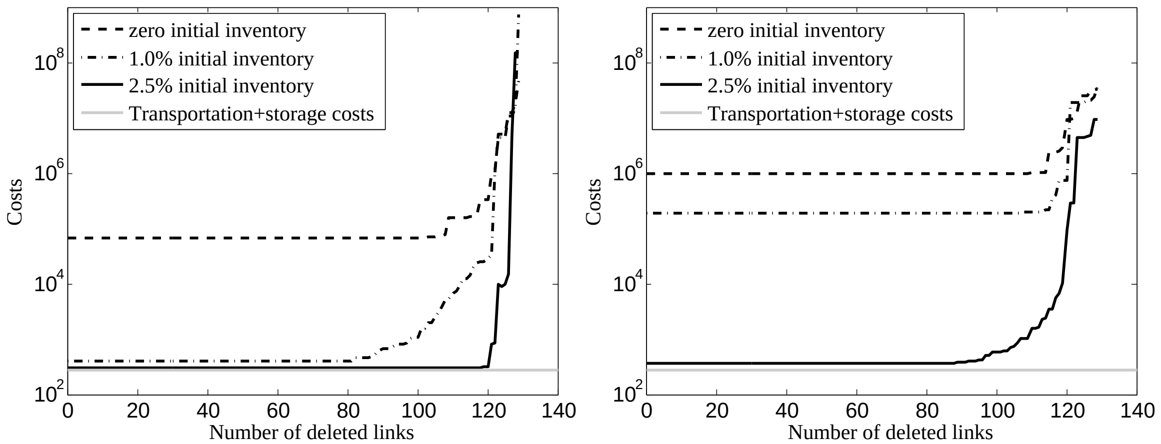
The corresponding values of ϵ_L suggest that the generated heuristic networks on average consist of 2.5 times the number of links in a minimally connected network. This means that each network facility is on average connected to two or three other network facilities. Apparently, this restricted amount of connectivity suffices to provide a large portion of the operational efficiency and robustness of the fully connected network. Another interesting result is that the number of links in the "heuristic network" depends on the product mix. A circle in Figure 2.5 represents an instance with a product mix with only one high value product and in this case 4 low value products, whereas a plus represents an instance with a product mix with at least two high value products. The results suggest that more links are required when more high value products are present. We give the following explanation for this phenomenon: the structure of the "heuristic network" is mostly determined by the number and locations of suppliers and consumers of high value products. When more high value products are present, in general the number of suppliers and consumers of high value products increases, and thus more links have to be established to provide just-in-time delivery.

To check the sensitivity of the heuristic network to the first moments of supply and demand distributions, we perform the following experiment: we take the 100 heuristic networks, where F=1.01, and 100 accompanying results of ϵ_c from the previous experiments where the means of supplies and demands stay the same, but the variances change

(Figure 2.5b). In addition, for each of the 100 heuristics networks we generate supply and demand time series from completely different distributions and determine the corresponding ϵ_c. Figure 2.6 depicts two empirical cumulative distribution functions of 100 ϵ_c values generated from i) time series with different variances, but the same means as the heuristic network originally was generated from (solid line), and ii) time series with different means (dotted line). The solid line again confirms that the heuristic network is robust to the variances since the cumulative distribution grows to 100% very fast starting at $\epsilon_c = 1$. The cumulative distribution function for the instances with different means grows much slower. Furthermore, only 80% of the ϵ_c values are smaller than or equal to 10. These results suggest that indeed the heuristic network ($F = 1.01$) is sensitive to changes in the means of supply and demand distributions.

Inventory Flexibility

In all previously performed experiments the parameter of initial inventory $y_w^k(0)$ has been set to zero, which is typical in practice for an LSP. It turns out that this initial inventory position is the only controllable parameter that can influence the operational costs for running the fully connected network. Consider the black dotted lines in Figure 2.7a and 2.7b, which show results of the heuristic for $S = 4$, $W = 3$, $D = 20$ and $K = 5$, for respectively two different supply and demand time series with equal first moments and $y_w^k(0) = 0$. The fact that the means of supply and demand are equal implies that approximately the same amount of products is shipped along time period T, and thus approximately the same transportation costs are present for both these cases. Nevertheless, a large difference in costs for the fully connected network can be noticed. This difference can therefore only be caused by backlog costs, which are induced by the specific time series. For instance, a time series, which exhibits a peak in demand at the beginning of period T, always leads to backlog costs, even in a fully connected network. Therefore, we pose the hypothesis that the only controllable parameter that influences the costs for the fully connected network is the initial inventory. Increasing the initial inventory will compensate for early backlog, which eventually will lead to convergence to the inevitable transportation costs.



(a) Results time series 1.

(b) Results time series 2.

Figure 2.7: Initial inventory flexibility.

To confirm this hypothesis, we perform the following additional experiments for the two supply and demand time series as mentioned above: we generate a supply and demand time series, set the inventory position in each of the warehouses to zero and determine the operational costs dependent on the number of deleted links (black dotted lines in Figure 2.7a and 2.7b). Next, we perform two additional optimizations for the same supply and demand time series where 1.0% and 2.5% of the total shipped amount of products is present in the warehouses as an initial condition, respectively. The results of these experiments are depicted in Figure 2.7a and 2.7b. As one can see, for both cases, the total costs of the fully connected network indeed converge to the transportation and storage costs as initial inventory increases.

2.4 Conclusions and Recommendations

In this chapter we considered transportation networks from the point of view of a third party logistics provider, who deals with the problem of distributing different types of products from suppliers to consumers via transportation links. Warehouses between the suppliers and consumers may be used to compensate for the stochastic behavior of supplies and demands and to consolidate different products. The amounts of these supplies and demands are assumed to be uncontrollable for the logistics provider. With only information about the supplies and demands a few days in advance, the logistics provider has to decide on which transportation links to use for a long period of time (tactical level) and how much of which products to ship through them each day (operational level).

We formulated a bi-level joint network design and network operation problem: at the upper (tactical) level, the network topology has to be constructed, and at the lower (operational) level routings and schedules for daily shipments have to be decided on. A model predictive control with a rolling horizon (MPC) was used as decision model on this operational level. A heuristic was proposed to construct the topology dependent on the operational decisions and compared to a bi-objective branch and bound method. The consolidation of product types so as to leverage on the scales economies is taken into consideration in this network design heuristic.

Experimental results reveal that the cost of such a near-optimally operated network as a function of the number of links in the network stays almost constant as the vast majority of network links is removed and explodes once the link number has decreased below a critical value. The experiments show that our heuristic determines a network that has close to optimal costs with a very low number of links (typically about 10% of all links that can be deleted before a minimally connected network remains).

Furthermore, experimental results suggested that the resulting topology is insensitive to second and higher order moments of the individual supply and demand distributions. Hence, information about the means of supplies and demands over a certain time period will suffice to generate a close to optimal network topology robust to any supply and demand scenarios with the same means.

The distribution network design problem, faced in this chapter, involves decision making at different levels: a tactical network topology with a minimal number of links and optimal operational performance is to be found. For real-life instances, this problem is cannot be solved to optimality within reasonable time. However, the problem is very suitable to be decoupled into two related problems, one at the tactical and one at the operational level. The proposed bi-level approach is solved very fast for real-life instances and still finds a satisfactory solution. In the next four chapters, we proceed with the approach of multi-step optimization by cutting the combined decision making problem in a container operation into four main subproblems, which then are solved step by step. Like in this chapter, the cuts are chosen such that the remaining subproblems are i) interesting from a practical point of view and ii) solvable within the computation time allowed by the specific decision level. A subproblem for instance that considers only strategic decisions is allowed to run for a couple of days or even a week. An operational subproblem that has to be solved each hour of the day however, should take no more than a couple of minutes such that results can still be analyzed and interpreted.

A recommendation is the extension of the current approach to include multi-echelon networks, allowing shipments between regional and local warehouses in a whole distribution network. In addition, the inclusion of more accurate travel times, including some that are smaller or longer than a day can be considered.

The study on distribution networks as presented in this dissertation only involves tactical and operational decisions. As a recommendation, our bi-level optimization scheme could be extended to allow for different strategic decisions. In particular, while the current research studied the influence of the operational level on the tactical level, we have ignored the strategic level by fixing the number of warehouses in advance. In reality, warehouses can be opened or shut depending on the amount of traffic going through them. Determining the right number and location of potential warehouses for a given source and sink distribution in a logistic network therefore is an interesting extension.

Chapter 3

Strategic Allocation in a Multi-Terminal Container Operation

3.1 Introduction

In the previous chapter, we considered the joint decision making problem at the tactical and operational level in a distribution network. Since the overall problem is too complex to be solved within satisfactory time, it was cut into two problems, which were solved alternatingly. Although this procedure does not find a global optimum, it appears to be very fast and still finds a satisfactory solution. The approach of cutting the overall decision making problem into multiple subproblems is also applied for a multi-terminal container operation. In this chapter and chapters 4, 5 and 6, we subsequently address one of the four chosen subproblems (see Figure 1.5). In this chapter, we start from combined problems at the strategic and tactical decision making levels (step 1 in Figure 1.5). The question at the strategic level is whether the same amount of loops can be processed with less crane capacity. We think this can be achieved by modifying the terminal and time allocation at the tactical level.

In many container ports, a number of terminal operators takes care of the logistics processes for container handling. Commonly, the tasks are divided such that one terminal operator is responsible for one terminal (or at least for the major share of one terminal), at which various vessel lines have one of their loops calling. The berth allocation problem (BAP) then involves the allocation of these loops in time and space in order to minimize a certain objective function ([15], [35], [24], [8], [27], [16], [51], and [36]). All these studies have in common that they consider the berth allocation problem for a single terminal.

In an increasing number of ports (e.g. Singapore, Antwerp and Rotterdam) however, one terminal operator is responsible for multiple terminals. In this dissertation, a multi-terminal container operation in the port of Antwerp is considered, which is run by only one terminal operator being PSA HNN. The overall problem then becomes to allocate i) the loop to which a number of vessels belong, to a terminal, ii) a time interval to a vessel for berthing, iii) a suitable berth position for a vessel within its terminal, and iv) a number of quay cranes to a vessel, taking the cyclic nature of the system into account. The BAP in such ports can no longer be considered per terminal for two main reasons. One is that it makes sense to avoid peaks and troughs in quay crane utilization and to spread vessel calls evenly over the various terminals. The other is that transshipment containers will unavoidably generate inter-terminal traffic, whose costs should be taken into account. All possible flows of containers through such a port are depicted in Figure 3.1.

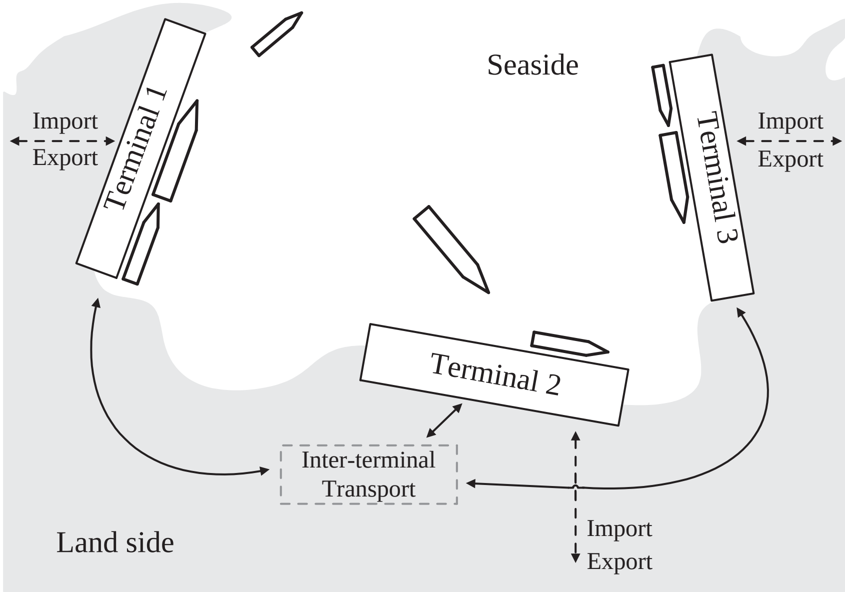


Figure 3.1: Container flows in a cluster of multiple terminals.

The multi-terminal allocation problem falls naturally apart in four subproblems, following the general classification of decision making: The first is a strategic problem, i.e. which terminal and which time interval to allocate to each (vessel in the) loop. This strategic problem is reconsidered occasionally when contracts for new loops are negotiated or existing contracts are renegotiated. The algorithm to solve this problem optimally may run for several hours, if not days.

Second, it is important to build in some pro-active robustness (by means of quay and crane reservations) in these timetables such that the risk of delay propagation is minimized.

The next problem is of a more tactical nature. Given the terminal and berthing interval for each call, the problem is where to berth the vessels along the quay. This is different from the studies mentioned above for two reasons. One is that the time window at this level is no longer a variable of the problem, it is given. Secondly, the objective is not to minimize the makespan of each vessel, but to use the time allowed to optimally serve the vessel. The service to be optimized at the tactical level concerns the carriers in between quay and yard. The objective is to find berth positions at the quay and container areas in the yard such that the total carrier travel distance is minimized given the strategic timetable and expected call sizes and call compositions.

The timetables and yard layout per terminal are kept for a year or two and considered to be the cyclic reference plan. In practice however, all kinds of disturbances take place. The operational level is concerned with the disruption management of the disturbed container operation. This involves the allocation of actual start and end process times to a vessel, the actual berth position to a vessel and the actual quay cranes processing a vessel.

In this chapter, the strategic problem of allocating a terminal, and a time interval of berthing to each vessel is addressed. In Chapter 4, the timetable per terminal, constructed in Chapter 3, is slightly modified to make it more robust to disturbances on vessels' arrivals. The tactical problem of finding berth positions and container areas is considered in Chapter 5. An on-the-fly approach for allocating positions and quay cranes in the disturbed system is proposed and discussed in Chapter 6.

3.1.1 Related Work

Descriptions and classifications as well as solution methods for the main logistics processes in container ports are given in [46], [49] and [43]. These studies determine the so-called berth allocation problem (BAP) as one of the key issues in a container port. Existing studies as discussed below all consider different versions of the single terminal BAP, which allocates a berth position and a berth interval to a set of vessels within one terminal. To our knowledge and as stated in [39], the BAP for multiple interrelated terminals has not yet been considered.

We however address the multi-terminal BAP, which is concerned with the allocation of i) a terminal, ii) a berth interval, iii) a berth position within the terminal, and iv) a number of quay cranes to a set of vessels in a multi-terminal container operation. This problem is solved in multiple steps: in this chapter, a terminal and a berth interval is allocated to vessels. Next in Chapter 4, the constructed timetable per terminal is slightly modified to increase its robustness. Then, appropriate berth positions within the allocated terminals are generated for the given (robust) timetable (Chapter 5). The operational quay crane scheduling is addressed in an online procedure in Chapter 6. Each of the individual subproblems are not only relevant from a practical point of view, but are also solvable within the time allowed at the corresponding decision level. Although the subproblems considered are all typically different from the single terminal BAP, still some principles from the single terminal BAP are relevant and therefore worth discussing below.

In the last two decades intensive research has been conducted on the single-terminal BAP. The single-terminal BAP consists of two interrelated assignment problems: allocate i) a berthing position and ii) a time interval of berthing to each vessel within one terminal such that vessels are not overlapping. The objective in these studies is often to minimize the vessels' turnaround times.

In this dissertation however, the multi-terminal BAP is addressed and solved in multiple steps. Accordingly, in this chapter we only allocate i) a terminal and ii) a berth interval (start and end time of processing) to a set of vessels. Multi-objectives are to minimize i) the container transport between different terminals (by minimizing the allocation of connecting vessels to different terminals), ii) the maximally required crane capacity per terminal (by evenly distributing the workload among the terminals), and deviations from preferred berth times of vessel lines.

In existing studies, the single-terminal BAP is modeled either as a discrete or a continuous problem. In the discrete case, the quay is divided into segments with specific lengths or even points when the quay and vessels' lengths are ignored. The problem can then be modeled as a parallel machine scheduling problem [42] and [32] where each vessel is a job and each berth a machine. However, large segments result in a poor space utilization, whereas small segments might lead to an infeasible solution. The continuous approach, where vessels can berth anywhere along the quay, circumvents these difficulties, however is more complex from a computational point of view.

Although, our problem can be considered as a multiple-job-on-one-processor scheduling problem, where each processor represents a terminal and each vessel a job, we only guarantee non-exceeding of quay and crane capacities, rather than positioning the vessels along the quay in this chapter. Subsequently in Chapter 5, for a given timetable, a continuous position allocation problem is solved and hence vessels are assumed to berth anywhere along the quay. This cut enables to solve both subproblems very efficiently.

Besides the distinction between a continuous and discrete case, existing studies on the single-terminal BAP consider either a static or dynamic case. The static case assumes all vessels to be in the port before the berth allocation is determined. This implies that each vessel can be allocated anywhere in time. The discrete, static berth allocation problem [21] is an assignment problem and is solvable in polynomial time with the Hungarian method [40]. This method assigns jobs to machines by sequentially computing shortest paths until each job is assigned to a machine. In the dynamic case, vessels arrive while work is in progress, in which case there may be idle times between successive vessels. The dynamic BAP is NP-hard for both the continuous and the discrete case [14].

In this dissertation, we address a strategic problem and therefore we first assume that each vessel can be positioned anywhere in time (although deviations from preferred berthing intervals are minimized). In an additional case study, lower and upper bounds are placed around start and end berth times, which restricts the vessel order and makes the problem dynamic. Although the dynamic BAP is NP-hard for both the continuous and the discrete case [14], we can still solve real-life instances of our problem within satisfactory time since we abstract from the position allocation in this chapter.

Studies that present heuristics solution approaches for several versions of the discrete and/or continuous dynamic single-terminal BAP with fixed process times can be found in [10], [37], [51], [15] and [27]. Other studies however assume that the process time of a vessel depends on its berth position [23], [24], [22], [20], [8], [16], and [35]. This assumption results from the fact that carriers have to cover a certain distance between a vessel's berth and the designated stacks for the vessel's containers in the yard.

A limited number of studies addresses a two phase problem of berth allocation and crane scheduling [20] and [41]. The authors in [20] assume that the process time of a vessel depends on its berth position, construct a mixed integer linear program for the joint problem of berth allocation and crane scheduling, and solve it using a genetic algorithm. The authors in [41] take into account that a vessel's process time depends is inversely proportional to the number of cranes assigned to it. The problem is solved in two phases. The first phase determines the vessels' positions and berth times and an integer number of quay cranes in each time segment for each vessel. A sub-gradient optimization technique is applied to obtain a near-optimal solution of the first phase. The second phase constructs a schedule for each individual quay crane guaranteeing non-crossing of cranes.

In this chapter, a vessel's process time is assumed to be inversely proportional to the crane capacity allocated to it. The actual crane scheduling problem is addressed in Chapter 6 of this dissertation. The main difference with [41] is that we consider the multi-terminal BAP (rather than the single-terminal BAP) and cut the problem into four subproblems, rather than two. The first three subproblems are typically different from existing studies. Only the subproblem of crane scheduling is quite similar to the second phase in [41]. We propose a different solution technique for this problem in Chapter 6.

To solve the multi-terminal BAP, it is necessary to incorporate the complex of interacting terminals in one model. This model should take into account the amount of inbound and outbound containers and their corresponding destinations. Inbound containers of an arriving vessel for instance could be partly destined for the hinterland and partly for another vessel. Hence, allocation of the two involved vessels to different terminals implies inter-terminal traffic and thus additional costs. However, due to other objectives and constraints this may still be the best or only solution.

3.1.2 Contributions

In this chapter we consider one terminal operator, who is responsible for a number of terminals in one container port. A vessel (or actually the loop to which a number of vessels belong) is allocated to a terminal for a certain amount of time to be processed by a number of quay cranes. Although we guarantee that terminal quay lengths as well as quay crane capacities are never exceeded, i) the berth position and ii) the actual quay crane scheduling within a terminal are still to be determined at a tactical and operational level, respectively. The tactical level of allocating berth positions to vessels and positions for their containers in the yard is addressed in Chapter 5, while the operational on-the fly vessel allocation and quay crane scheduling are treated in Chapter 6. These cuts into specific subproblems are depicted in Figure 1.5. The strategic subproblem, considered in this chapter, involves a number of dependent, one-dimensional packing problems, which allow capacitated parallel processing. This is a very interesting problem from a practical point of view, which still can be solved within satisfactory time.

Most of the existing studies after the BAP consider a set of vessels within a certain time horizon. The corresponding objective in these researches often reduces to fitting all vessels within a time horizon and minimizing the total weighted handling time for all vessels. However, in practice most vessels run a regular service on their ports, for instance once a week, which makes the system cyclic. Vessels can arrive at the end of the considered time period (cycle) and leave at the beginning of this time period (next cycle). Relating this to the packing problem implies that rectangles (vessels) can be cut into two pieces, where one piece is placed at the end of the time horizon and the other piece at the beginning. The authors in [37] take this into consideration for a single terminal BAP.

The contributions of this chapter are the following:

- We address the strategic problem of allocating vessels to a certain terminal for a certain time interval of processing.
- The model in this chapter takes the cyclic nature of the system into consideration.
- An alternative approach, introduced here, is much faster solvable than the straightforward approach. Using the alternative approach, we are able to construct accurate allocations for real-life problems within a couple of hours. Since such a strategic problem is only reconsidered once each year or once each two years, a run time of a couple of hours or even a day is still satisfactory.
- Applying the alternative approach to a representative data set in a case study suggests that significant reductions in required crane capacities and inter-terminal transport can be achieved.

The chapter is structured as follows: in Section 3.2, the problem considered is formally phrased. Then, we introduce a straightforward mixed integer linear program to solve this problem. In addition, we propose an alternative mixed integer linear program. We compare the performance and discuss feasibility aspects in Section 3.3. In Section 3.4, the alternative approach is applied to a representative data set, provided by the terminal operator PSA HNN in Antwerp. Results suggest that significant reductions in required crane capacities and inter-terminal transport of a representative can be achieved. Finally, in Section 3.5, we draw conclusions and make recommendations for future research.

3.2 Mathematical Models

In this section, we first describe the problem considered in detail. Next, we propose and discuss two mathematical formulations to solve the problem. The first approach is straightforward in the sense that the way of modeling the berthing of vessels is common. The second model significantly reduces the number of variables. The way of modeling the cyclic property of the system is similar for both approaches.

3.2.1 Problem Description

For all of this chapter the following holds, unless stated differently: $t \in \{1, 2, \dots, T\}$, the cluster of terminals, $v \in \{1, 2, \dots, V\}$, the set of vessels, $z \in \{0, 1, 2, \dots, V\}$, the set of container destinations. Furthermore, we assume vessels to call cyclically, where each vessel in the set arrives exactly once each cycle. In general, the cycle length is a week for such a container operation. We consider discrete time k and unless stated differently, $k \in \{1, 2, \dots, K\}$, is the set of discrete time slots within the cycle.

In the cluster of terminals, the set of container vessels has to be unloaded and loaded. Vessel v imports a pre-determined number of inbound containers $I_{vz} \in \mathbb{N}$ with destination(s) z, where $v \neq z$. In this context, $z = 0$ means that containers are destined for the hinterland, whereas $z = 1, 2, \dots, V$ means that containers are destined for vessel $v = 1, 2, \dots, V$ respectively. Besides import containers brought in by vessels, a certain amount of containers H_v with destination v is imported from the hinterland by trucks and trains during the cycle. These containers are distributed among the different terminals dependent on their destination vessels. Furthermore, each vessel v exports a number of outbound containers $O_v \in \mathbb{N}$. Container transport between among the terminals is established by trucks.

Terminal t has a restricted quay length $L_t \in \mathbb{R}^+$ and a number of quay cranes $N_t \in \mathbb{N}$. Once berthing, vessel v occupies a certain amount of quay meters M_v. In addition, this length M_v determines the maximum number of quay cranes $S_v \in \mathbb{N}$ processing vessel v and the efficiency $\eta_v \in [0, 1]$ of the quay cranes on vessel v. In practice, quay cranes with different processing rates are present in the terminals. We do not take the specific allocation of quay cranes to vessels into account yet, but assume the average processing rate $\bar{\lambda}_t \in \mathbb{N}$ to be the processing rate of each quay crane in terminal t. So the handling time of vessel v in terminal t depends on i) the mean processing rate $\bar{\lambda}_t$ in terminal t, ii) the efficiency η_v of quay cranes operating vessel v, iii) the number of quay cranes processing vessel v and iv) the total number of inbound and outbound containers I_{vz} and O_v of vessel v. We assume the processing time of vessel v to be inversely proportional to the first three of these items and proportional to the latter. Furthermore, the number of quay cranes processing vessel v may change from one time slot to another. After the unloading and before the loading, containers can temporarily be stored in the yard of terminal t up to the yard's capacity W_t. The time it takes to transport containers from terminal p to terminal r is defined as $\Delta_{pr} \in \mathbb{N}$, $p, r \in \{1, ..., T\}$. Furthermore, we assume that the total number of time slots vessel v is berthing, is less then the total number of time slots K in the cycle. In addition, we assume that vessels arrive at the beginning of a time slot and depart at the end of a time slot.

Our goal is to minimize the total costs of the system, which consist of three conflicting elements: first of all, costs are associated with the number of quay cranes that have to be installed in the terminals in order to satisfy the proposed schedule. We define c_t to be the average costs of a quay crane in terminal t. Second of all, a fixed amount of money c_{pr} has to be paid for each container that is transported from terminal p to terminal r. Finally, the terminal operator has some contractual agreements with each of the vessel lines with respect to the berthing interval of the corresponding vessels. We define $A_v \in \mathbb{N}$ to be the preferred arrival time of vessel v in the port and $D_v \in \mathbb{N}$ to be the preferred departure time of vessel v. We have to remark that, in our definition, this means that vessel v prefers to depart at the end of time slot $\langle D_v - 2, D_v - 1]$. We assign factors of penalty costs c_v^c and c_v^d if in the constructed allocation vessel v departs earlier and later than its preferred departure time D_v, respectively.

Parameter	Definition
T	Number of terminals in the cluster
V	Number of vessels in the set
K	Number of discrete time slots within the cycle
L_t	Quay length [m]
M_v	Quay length required for vessel [m]
I_{vz}	# inbound containers to be unloaded from vessel v with destination z and $v \neq z$
O_v	# outbound containers to be loaded onto vessel v
H_v	# containers with destination v arriving from the hinterland during the cycle
A_v	Preferred arrival time of vessel v
D_v	Preferred departure time of vessel v
E_v	Parameter to distinguish between the cases $A_v < D_v$ and $D_v \geq A_v$
N_t	# quay cranes available in terminal t
S_v	Maximum # quay cranes, which can process vessel v
$\bar{\lambda}_t$	Mean processing rate of quay cranes in terminal t [containers/time slot]
η_v	Vessel efficiency with respect to quay crane rate [-]
Δ_{pr}	# time slots needed to transport containers from terminal p to r
W_t	# containers that can be stored in terminal t
c_v^a	Factor of penalty costs for vessel v for arriving too late $[\frac{\text{euro/container}}{\text{time slot}}]$
c_v^c	Factor of penalty costs for vessel v for departing too early $[\frac{\text{euro/container}}{\text{time slot}}]$
c_v^d	Factor of penalty costs for vessel v for departing too late $[\frac{\text{euro/container}}{\text{time slot}}]$
c_{pr}	Factor of transportation costs from terminal p to r [euro/ container]
c_t	Factor of costs for required equipment in terminal t [euro/ quay crane]

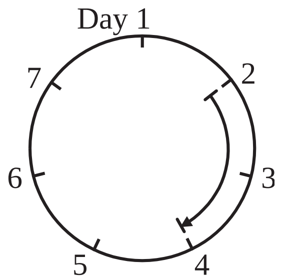
Table 3.1: Model parameters

With respect to the cyclic property of the considered system, we have two additional remarks. First, we require conservation with respect to the arrival and departure of containers:

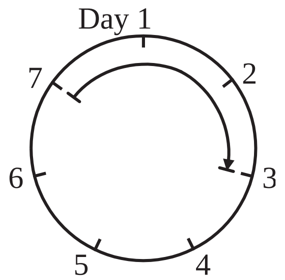
$$\sum_{i=1}^{V} I_{iv} + H_v = O_v \qquad \forall v \tag{3.1}$$

Second, we notice that both $A_v \geq D_v$ and $A_v < D_v$ are possible, since we model cyclically arriving container vessels (see also Figure (3.2)). Therefore, we introduce an auxiliary parameter E_v, which explicitly distinguishes between both cases:

$$E_v = \begin{cases} 1 & \text{if } A_v \geq D_v, \\ 0 & \text{if } A_v < D_v. \end{cases} \qquad \forall v$$



(a) $A_v = 2$ and $D_v = 5$: $E_v = 0$.



(b) $A_v = 7$ and $D_v = 4$: $E_v = 1$.

Figure 3.2: Possible arrivals and departures of vessel v in terminal t in a cyclic system with $K = 7$.

In the next section it becomes clear why we need this parameter. The sets and parameters discussed above are conveniently arranged in Table 3.1.

3.2.2 Straightforward MILP

Binary variables

$$a_{tv}(k) = \begin{cases} 1 & \text{if in terminal } t \text{ vessel } v \text{ berths during time slot } \langle k-1, k], \\ 0 & \text{otherwise.} \end{cases}$$

$$d_{tv}(k) = \begin{cases} 1 & \text{if in terminal } t \text{ vessel } v \text{ departs during time slot } \langle k-2, k-1], \\ 0 & \text{otherwise.} \end{cases}$$

Auxiliary binary variables

$$b_{tv}(k) = \begin{cases} 1 & \text{if in terminal } t \text{ vessel } v \text{ is berthing during time slot } \langle k-1, k], \\ 0 & \text{otherwise.} \end{cases}$$

$$e_{tv} = \begin{cases} 1 & \text{if } a_{tv}(k_a) = 1 \text{ and } d_{tv}(k_d) = 1 \text{ and } k_a > k_d, \\ 1 & \text{if in terminal } t \text{ vessel } v \text{ is continuously berthing,} \\ 0 & \text{otherwise.} \end{cases}$$

$$e_v^a = \begin{cases} 1 & \text{if } a_{tv}(k_a) = 1 \text{ and } k_a < A_v, \\ 0 & \text{if } a_{tv}(k_a) = 1 \text{ and } k_a \geq A_v. \end{cases}$$

Continuous variables

Integer variables

Constraints

Vessel v can arrives once each cycle at exactly one terminal:

$$\sum_{t=1}^{T} \sum_{k=1}^{K} a_{tv}(k) = 1 \qquad \forall v \tag{3.2}$$

$m_{tv}(k)$	=	Amount of quay meters consumed in terminal t by vessel v during time slot $\langle k-1, k]$ [m]
$q_{tv}(k)$	=	Amount of quay processing vessel v in terminal t during time slot $\langle k-1, k]$
$h_{tv}(k)$	=	Amount of containers from hinterland transported into terminal t with destination v during time slot $\langle k-1, k]$ [containers/ time slot]
$f_{prv}(k)$	=	Amount of containers transported from terminal p to terminal r with destination v during time slot $\langle k-1, k]$ [containers/ time slot], $p \neq r$
$w_{tv}(k)$	=	WIP in terminal t with destination v at time k
n_t	=	Number of quay cranes required in terminal t

Δ_v^a	=	Number of time slots vessel v berths too late
Δ_v^c	=	Number of time slots vessel v departs too early
Δ_v^d	=	Number of time slots vessel v departs too late

Furthermore, vessel v departs once each cycle from the same terminal it arrives at:

$$\sum_{k=1}^{K} a_{tv}(k) - \sum_{k=1}^{K} d_{tv}(k) = 0 \qquad \forall t, v \tag{3.3}$$

If vessel v arrives at and departs from terminal t, terminal t is occupied by vessel v only between its arrival and departure time.

Now assume that vessel v berths at terminal $t = 1$, arrives at time slot k_a and departs and time slot k_d, where the cycle length $K = 10$. Since the system is cyclic, we can distinguish three different cases with respect to the berthing of a vessel:

1. $k_a < k_d$
   This means that the arrival and departure of vessel v take place in the same cycle. Assume $k_a = 3$ and $k_d = 7$. The corresponding values for $a_{1v}(k)$, $d_{1v}(k)$ and $b_{1v}(k)$ are depicted in the table below:

k	1	2	3	4	5	6	7	8	9	10
$d_{1v}(k)$	0	0	0	0	0	0	1	0	0	0
$a_{1v}(k)$	0	0	1	0	0	0	0	0	0	0
$\sum_{k'=k+1}^{K} d_{1v}(k')$	1	1	1	1	1	1	0	0	0	0
$\sum_{k'=k+1}^{K} a_{1v}(k')$	1	1	0	0	0	0	0	0	0	0
$b_{1v}(k)$	0	0	1	1	1	1	0	0	0	0

   To model the dependency between $a_{tv}(k)$ and $d_{tv}(k)$ on one hand, and $b_{tv}(k)$ on the other for the case where $k_a < k_d$, we propose the following constraint:

$$b_{tv}(k) = \sum_{k'=k+1}^{K} \big(d_{tv}(k') - a_{tv}(k')\big) \qquad \forall t, v, k \tag{3.4}$$

2. $k_a > k_d$
   This means that vessel v arrives in one cycle and departs in the next one. Assume

$k_a = 8$ and $k_d = 4$. The corresponding values for $a_{1v}(k)$, $d_{1v}(k)$ and $b_{1v}(k)$ are depicted in the table below:

k	1	2	3	4	5	6	7	8	9	10
$d_{1v}(k)$	0	0	0	1	0	0	0	0	0	0
$a_{1v}(k)$	0	0	0	0	0	0	0	1	0	0
$\sum_{k'=k+1}^{K} d_{1v}(k')$	1	1	1	0	0	0	0	0	0	0
$\sum_{k'=k+1}^{K} a_{1v}(k')$	1	1	1	1	1	1	1	0	0	0
$b_{1v}(k)$	0	0	0	-1	-1	-1	-1	0	0	0

To model the dependency between $a_{tv}(k)$ and $d_{tv}(k)$ on one hand, and $b_{tv}(k)$ on the other for the case where $k_a > k_d$, we propose the following constraint:

$$b_{tv}(k) - 1 = \sum_{k'=k+1}^{K} \big(d_{tv}(k') - a_{tv}(k')\big) \qquad \forall t, v, k \tag{3.5}$$

3. $a_v = d_v$

   This means that vessel v i) arrives and departs during the same time slot in the same cycle (no berthing at all) or ii) arrives in a certain cycle and departs in the next time cycle (continuously berthing).

   For the case where no berthing takes place, the accompanying values for $b_{tv}(k)$ (see table below) follow from (3.4).

k	1	2	3	4	5	6	7	8	9	10
$b_{tv}(k)$	0	0	0	0	0	0	0	0	0	0

   For the case where vessel v is continuously berthing, the accompanying values for $b_{tv}(k)$ (see table below) follow from (3.5).

k	1	2	3	4	5	6	7	8	9	10
$b_{tv}(k)$	1	1	1	1	1	1	1	1	1	1

We introduce the auxiliary binary variable e_{tv} to distinguish between the cases as given in the definition of e_{tv}, resulting from the cyclic nature of the system. The principle used here is similar to the one for the auxiliary parameter E_v (see also Figures 3.2a and 3.2b).

$$b_{tv}(k) - e_{tv} = \sum_{k'=k+1}^{K} \big(d_{tv}(k') - a_{tv}(k')\big) \qquad \forall t, v, k \tag{3.6}$$

Furthermore, vessel v can only berth at one terminal:

$$\sum_{t=1}^{T} b_{tv}(k) \leq 1 \qquad \forall v, k \tag{3.7}$$

The sum of lengths of all vessels berthing at terminal t during time slot $\langle k-1, k]$ should be less than or equal to the total quay length in terminal t:

$$\sum_{v=1}^{V} M_v \cdot b_{tv}(k) \leq L_t \qquad \forall t, k \tag{3.8}$$

Vessel v can be operated by up to S_v quay cranes in terminal t during time slot $\langle k-1, k]$, only if this vessel is berthing in terminal t during time slot $\langle k-1, k]$:

$$q_{tv}(k) \leq S_v \cdot b_{tv}(k) \qquad \forall t, v, k \tag{3.9}$$

Vessel v has to be fully processed during the cycle:

$$\sum_{t=1}^{T} \sum_{k=1}^{K} \eta_v \bar{\lambda}_t \cdot q_{tv}(k) = \sum_{z=0}^{Z} I_{vz} + O_v \qquad \forall v \tag{3.10}$$

We want to minimize the maximum number of quay cranes in terminal t ever required during the cycle. Therefore, we introduce an auxiliary variable n_t, which is a soft upper bound on the number of quay cranes in terminal t. This variable n_t is present in the objective function:

$$\sum_{v=1}^{V} q_{tv}(k) \leq n_t \qquad \forall t, k \tag{3.11}$$

The maximum number of quay cranes ever required in terminal t during the cycle cannot be larger than the number of quay cranes actually available in terminal t:

$$n_t \leq N_t \qquad \forall t \tag{3.12}$$

The sum over the cycle's time slots of the number of containers with destination v, transported from the hinterland into the different terminals, should be equal to the total number of containers with destination v arriving from the hinterland during the cycle.

$$\sum_{k=1}^{K} \sum_{t=1}^{T} h_{tv}(k) = H_v \qquad \forall v \tag{3.13}$$

Since the system is cyclic, the storage level in the terminals and the inter-terminal transport during time slot $\langle k-1, k]$ should equal the storage level in the terminals and the inter-terminal transport during time slot $\langle k-1+\alpha K, k+\alpha K]$, where $\alpha \in \mathbb{N}$:

$$w_{tv}(k) = w_{tv}(k+\alpha K) \qquad \forall t, v, k \tag{3.14}$$

and

$$f_{prv}(k) = f_{prv}(k+\alpha K) \qquad \forall p, r, v, k \tag{3.15}$$

We assume that inbound containers with destination 0 ("hinterland") are transported into the hinterland directly after they arrive in the terminal and are not counted as stack.

The amount of containers in terminal t with destination v during time slot $\langle k-1, k]$ is equal to the amount of containers in terminal t with destination v during time slot $\langle k-2, k-1]$ plus all incoming flows (inbound containers from vessels, containers from other terminals and containers from the hinterland) minus all outgoing flows (outbound containers to vessels and containers to other terminals). We assume that loading and unloading of containers from vessel v with different destinations is divided proportionally among the time slots vessel v is actually berthing. To this, we first define the constants $\beta_{vz} = \frac{I_{vz}}{\sum_{z=0}^{Z} I_{vz}+O_v}$ and $\gamma_v = \frac{O_v}{\sum_{z=0}^{Z} I_{vz}+O_v}$, and derive appropriate constraints:

$$\begin{aligned} w_{tv}(k) &= w_{tv}(k-1) + \sum_{i=1}^{V} \beta_{iv}\eta_i\bar{\lambda}_t \cdot q_{ti}(k) - \gamma_v\eta_v\bar{\lambda}_t \cdot q_{tv}(k) + h_{tv}(k) + \\ & \sum_{r=1}^{T} f_{rtv}(k-\Delta_{pr}) - \sum_{r=1}^{T} f_{trv}(k) \qquad \forall t, v, k \end{aligned} \tag{3.16}$$

If we start with bringing containers into the yard during time slot $\langle k-1, k]$, the following constraint has to be satisfied:

$$\sum_{v=1}^{V} \Big(w_{tv}(k-1) + \sum_{i=1}^{V} \beta_{iv}\eta_i\bar{\lambda}_t \cdot q_{ti}(k) + h_{tv}(k) + \sum_{r=1}^{T} f_{rtv}(k-\Delta_{pr}) \Big) \leq W_t \qquad \forall t \tag{3.17}$$

If we start with taking away containers from the yard during time slot $\langle k-1, k]$, the following constraint has to be satisfied:

$$w_{tv}(k-1) - \gamma_v\eta_v\bar{\lambda}_t \cdot q_{tv}(k) - \sum_{r=1}^{T} f_{trv}(k) \geq 0 \qquad \forall t, v, k \tag{3.18}$$

Whatever order is applied during the cycle, (6.9) and (3.18) guarantee that never too much and never too less (negative amount of) containers are in the yard.

We use the additional integer variables Δ_v^a, Δ_v^c and Δ_v^d to model the number of time slots vessel v berths too late, departs too early and departs too late, respectively. We assume vessel v arrives in the port at time A_v, which means that the actual berth time can only take place exactly at or later than A_v. We assume that the costs Δ_v^a for late arrival of vessel v depend linearly on the difference between the preferred arrival time A_v and the actual arrival time. Due to the cyclic nature of the system, both $a_{tv}(k_a) = 1$, where $k_a < A_v$ and $k_a \geq A_v$ are possible. Hence, a jump in the cost function occurs at A_v as depicted in Figure (3.3).

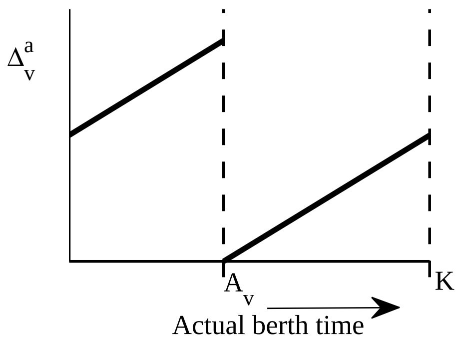


Figure 3.3: Costs for berthing too late.

To model this jump, we introduce the auxiliary binary variable e_v^a in an additional constraint:

$$\Delta_v^a = \sum_{t=1}^{T}\sum_{k=1}^{K} \Big(-\big(A_v - k\big) \cdot a_{tv}(k) \Big) + K \cdot e_v^a \qquad \forall v \tag{3.19}$$

where

$$K \cdot e_v^a \geq \sum_{t=1}^{T} \sum_{k=1}^{K} \big(A_v - k\big) \cdot a_{tv}(k) \qquad \forall v \tag{3.20}$$

With respect to departing too early or too late one can distinguish $4! = 24$ permutations (of a_v, A_v, d_v and D_v. It turns out that with help of the introduced auxiliary binary variables e_v^a and e_{tv}, and the auxiliary parameter E_v as defined in Section 3.1, we are able to construct appropriate constraints for Δ_v^c and Δ_v^d to satisfy each of the 24 cases:

$$\Delta_v^c \geq \sum_{t=1}^{T} \sum_{k=1}^{K} \Big(\big(D_v - k\big) \cdot d_{tv}(k) \Big) - K \cdot e_v^a + K \cdot E_v - \sum_{t=1}^{T} K \cdot e_{tv} \qquad \forall v \tag{3.21}$$

where

$$\Delta_v^c \geq 0 \qquad \forall v \tag{3.22}$$

and

$$\Delta_v^d \geq \sum_{t=1}^{T} \sum_{k=1}^{K} - \Big(\big(D_v - k\big) \cdot d_{tv}(k) \Big) + K \cdot e_v^a - K \cdot E_v + \sum_{t=1}^{T} K \cdot e_{tv} \qquad \forall v \tag{3.23}$$

where

$$\Delta_v^d \geq 0 \qquad \forall v \tag{3.24}$$

Finally, some of the continuous variables have to be lower-bounded:

$$\begin{aligned} q_{tv}(k) &\geq 0 && (3.25) \\ h_{tv}(k) &\geq 0 && (3.26) \\ f_{prz}(k) &\geq 0 && (3.27) \\ w_{tz}(k) &\geq 0 && (3.28) \end{aligned}$$

Objective function

Linear penalty costs are assigned when vessel v berths later than its arrival time and/ or when vessel v departs too early or too late (c_v^a, c_v^c, c_v^d respectively). Furthermore, a linear unit penalty cost is assigned when containers are transported from one terminal to another (c_{pr}). Finally, linear costs are assigned to the number of required quay cranes in terminal t (c^t). The decision variables are represented in a vector $\vec{u}(k) = [a_{tv}(k), d_{tv}(k), h_{tv}(k), q_{tv}(k), f_{prv}(k)]^T$ and the objective function is formulated:

$$\min_{\vec{u}(1),\ldots,\vec{u}(K)} \quad \sum_{v=1}^{V} \big(c_v^a \Delta_v^a + c_v^c \Delta_v^c + c_v^d \Delta_v^d \big) + \sum_{k=1}^{K} \sum_{p=1}^{T} \sum_{r=1}^{T} \sum_{z=1}^{Z} c_{pr} f_{prz}(k) + \sum_{t=1}^{T} c_t n_t \tag{3.29}$$

Remark: In the solution of this MILP it could be that an arbitrary amount of containers is stored in a certain terminal during the entire cycle. This could be prevented by assigning a small cost for each stored container.

3.2.3 Alternative MILP

In the previous section, we introduced a straightforward approach of modeling the problem. The cyclic nature of the system is taken into account by (4.9), (4.6) and (4.13), and (3.19) through (3.24). In this straightforward formulation, we used binary variables $a_{tv}(k)$ and $d_{tv}(k)$, which indicate whether or not a vessel berths at or departs from terminal t during time slot $\langle k-1, k]$. In the alternative approach we split these variables into the integer variables a_v and d_v and the binary variable x_{tv}. Here, a_v and d_v denote the time slots vessel v berths and departs, respectively. In our definition d_v means that the processing of vessel v ends at the end of time slot $\langle k-2, k-1]$. Additionally, x_{tv} denotes the terminal in which vessel v berths. Consequently, some of the constraints of the straightforward approach have to be adapted and even some new constraints have to be introduced to describe the same problem. In the end, however, the alternative way of modeling uses only a fraction $\frac{T+K+2}{3TK+T+1}$ of the number of binary variables in the straightforward way of modeling.

Continuous variables

$$
\begin{array}{lcl}
a_v & = & \text{Actual berth time slot of vessel } v \text{ (start of processing vessel } v\text{)} \\
d_v & = & \text{Actual departure time slot of vessel } v \text{ (processing of vessel } v \text{ ends at the end of} \\
& & \text{time slot } \langle k-2, k-1]
\end{array}
$$

The rest of the continuous variables are equivalent to the continuous variables as described in Section 3.2.

Binary variables

$$
x_{tv} = \begin{cases} 1 & \text{if in terminal } t \text{ vessel } v \text{ berths,} \\ 0 & \text{otherwise.} \end{cases}
$$

Auxiliary binary variables

$$
\begin{aligned}
b_v(k) &= \begin{cases} 1 & \text{if vessel } v \text{ is berthing during time slot } \langle k-1, k], \\ 0 & \text{otherwise.} \end{cases} \\
e_v &= \begin{cases} 1 & \text{if } a_v > d_v, \\ 0 & \text{if } a_v < d_v, \\ 1 & \text{if } a_v = d_v \text{ and vessel } v \text{ is continuously berthing,} \\ 0 & \text{if } a_v = d_v \text{ and vessel } v \text{ does not berth at all.} \end{cases} \\
e_v^a &= \begin{cases} 1 & \text{if } a_v < A_v, \\ 0 & \text{if } a_v \geq A_v. \end{cases}
\end{aligned}
$$

Constraints

Vessel v berths at only one terminal t:

$$
\sum_{t=1}^{T} x_{tv} = 1 \qquad \forall v \tag{3.30}
$$

The arrival and departure times (a_v and d_v respectively) of vessel v are within the cycle:

$$1 \leq a_v \leq K \qquad \forall v \tag{3.31}$$

and

$$1 \leq d_v \leq K \qquad \forall v \tag{3.32}$$

Since the system is cyclic, we can distinguish three different cases with respect to the berthing of a vessel:

1. $a_v < d_v$
   This means that vessel v arrives and departs in the same cycle. To better explain this, we present the next example: assume $K = 10$, $a_v = 3$ and $d_v = 7$. This means that, according to its definition, the variable $b_v(k)$ should have the following values:

k	1	2	3	4	5	6	7	8	9	10
$b_v(k)$	0	0	1	1	1	1	0	0	0	0

   To model the dependency between a_v and d_v on one hand, and $b_v(k)$ on the other for the case where $a_v < d_v$, we propose the following constraints: first of all, the number of berthing time slots should be equal to the difference between the departure and arrival time slot:

   $$\sum_{k=1}^{K} b_v(k) = d_v - a_v \tag{3.33}$$

   Second of all, after the departure of vessel v, vessel v is not berthing anymore:

   $$k \cdot b_v(k) \leq d_v - 1 \tag{3.34}$$

   Finally, before the arrival of vessel v, vessel v is not berthing yet:

   $$(K - k) \cdot b_v(k) \leq K - a_v \tag{3.35}$$

2. $a_v > d_v$
   This means that vessel v arrives in a certain cycle and departs in the next one. To better explain this, we present the next example. Assume $K = 10$, a_v=8 and $d_v = 4$. In the table below, we not only depict corresponding values of $b_v(k)$, but also of $b_v(k) - 1$.

k	1	2	3	4	5	6	7	8	9	10
$b_v(k)$	1	1	1	0	0	0	0	1	1	1
$b_v(k) - 1$	0	0	0	-1	-1	-1	-1	0	0	0

   The last row in this table enables to derive constraints similar to the ones we derived in the first case. First of all, the number of berthing time slots should be equal to the difference between the departure and arrival time slot plus K:

   $$\sum_{k=1}^{K} \big(b_v(k) - 1\big) = d_v - a_v \tag{3.36}$$

Second of all, after the arrival of vessel v, vessel v is berthing:

$$-k \cdot \big(b_v(k) - 1\big) \leq a_v - 1 \tag{3.37}$$

Finally, before the departure of vessel v, vessel v is berthing:

$$-(K-k) \cdot \big(b_v(k) - 1\big) \leq K - d_v \tag{3.38}$$

3. $a_v = d_v$
   This means that vessel v i) arrives and departs during the same time slot in the same cycle (no berthing at all) or ii) arrives in a certain cycle and departs in the next time cycle (continuously berthing).

   For the case where no berthing takes place, the accompanying values for $b_v(k)$ (see table below) follow from (4.10).

k	1	2	3	4	5	6	7	8	9	10
$b_v(k)$	0	0	0	0	0	0	0	0	0	0

   For the case where vessel v is continuously berthing, the accompanying values for $b_v(k)$ (see table below) follow from (4.14).

k	1	2	3	4	5	6	7	8	9	10
$b_v(k)$	1	1	1	1	1	1	1	1	1	1

We want to derive generic constraints, which relate a_v and d_v to $b_v(k)$ as well as $b_v(k)$ to a_v and d_v for each of the upper three cases. To this we introduce the binary variable e_v and combine (4.10) and (4.14) into

$$\sum_{k=1}^{K} \big(b_v(k) - e_v\big) = d_v - a_v \qquad \forall v \tag{3.39}$$

and

$$1 - a_v \leq k \cdot \big(b_v(k) - e_v\big) \leq d_v - 1 \qquad \forall v, k \tag{3.40}$$

and

$$d_v - K \leq \big(K - k\big) \cdot \big(b_v(k) - e_v\big) \leq K - a_v \qquad \forall v, k \tag{3.41}$$

Vessel v requires an amount of quay meters M_v at a terminal t during time slot $\langle k-1, k]$, iff the vessel is actually berthing during time slot $\langle k-1, k]$ at terminal t.

$$m_{tv}(k) \leq M_v \cdot x_{tv} \qquad \forall t, v, k \tag{3.42}$$

and

$$\sum_{t=1}^{T} m_{tv}(k) = M_v \cdot b_v(k) \qquad \forall v, k \tag{3.43}$$

Furthermore, the sum of lengths of all vessels berthing at terminal t during time slot $\langle k-1,k]$ should be less than or equal to the total quay length of terminal t:

$$\sum_{v=1}^{V} m_{tv}(k) \leq L_t \qquad \forall t,k \tag{3.44}$$

Vessel v can only be operated in terminal t iff vessel v is berthing in terminal t. Furthermore, a maximum number of quay cranes S_v can be assigned to vessel v:

$$q_{tv}(k) \leq S_v \cdot x_{tv} \qquad \forall t,v,k \tag{3.45}$$

and

$$q_{tv}(k) \leq S_v \cdot b_v(k) \qquad \forall t,v,k \tag{3.46}$$

Next, constraints (4.14) through (3.18) from the straightforward MILP are also valid in this formulation. Furthermore, we have to slightly adapt constraints (3.19) through (3.24):

$$\Delta_v^a = -\big(A_v - a_v\big) + K \cdot e_v^a \qquad \forall v \tag{3.47}$$

where

$$K \cdot e_v^a \geq A_v - a_v \qquad \forall v \tag{3.48}$$

and

$$\Delta_v^c \geq (D_v - d_v) - K \cdot e_v^a + K \cdot E_v - K \cdot e_v \qquad \forall v \tag{3.49}$$

where

$$\Delta_v^c \geq 0 \qquad \forall v \tag{3.50}$$

and

$$\Delta_v^d \geq -\big((D_v - d_v) - K \cdot e_v^a + K \cdot E_v - K \cdot e_v\big) \qquad \forall v \tag{3.51}$$

where

$$\Delta_v^d \geq 0 \qquad \forall v \tag{3.52}$$

All 24 permutations for the sequence of a_v, d_v, A_v and D_v in the cycle and the corresponding value of Δ_v^d are depicted in Figures 3.4 and 3.5.

Objective function

The decision variables are represented in a vector $\vec{u}(k) = [x_{tv}, a_v, d_v, h_{tv}(k), q_{tv}(k), f_{prz}(k)]^T$ and the objective function is equivalent to the one in (3.29).

3.3 Results

In the previous chapter, two approaches have been formulated, which model the described system. As a next step, both approaches are coded in Matlab and solved using CPLEX. Results for a large set of randomly generated instances suggest that the models find the same solution, but significantly differ in CPU time. In this section, we statistically compare the CPU times of both models dependent on the number of vessels V in the set. Results suggest that the alternative approach convincingly outperforms the straight-forward approach. Next, we investigate the CPU time of the alternative approach dependent on the number of time slots K in the considered cycle. Finally, we suggest that from generated terminal and time window allocations of realistic problems, we are able to construct feasible and satisfactory i) two-dimensional packing solutions and ii) quay crane allocations.

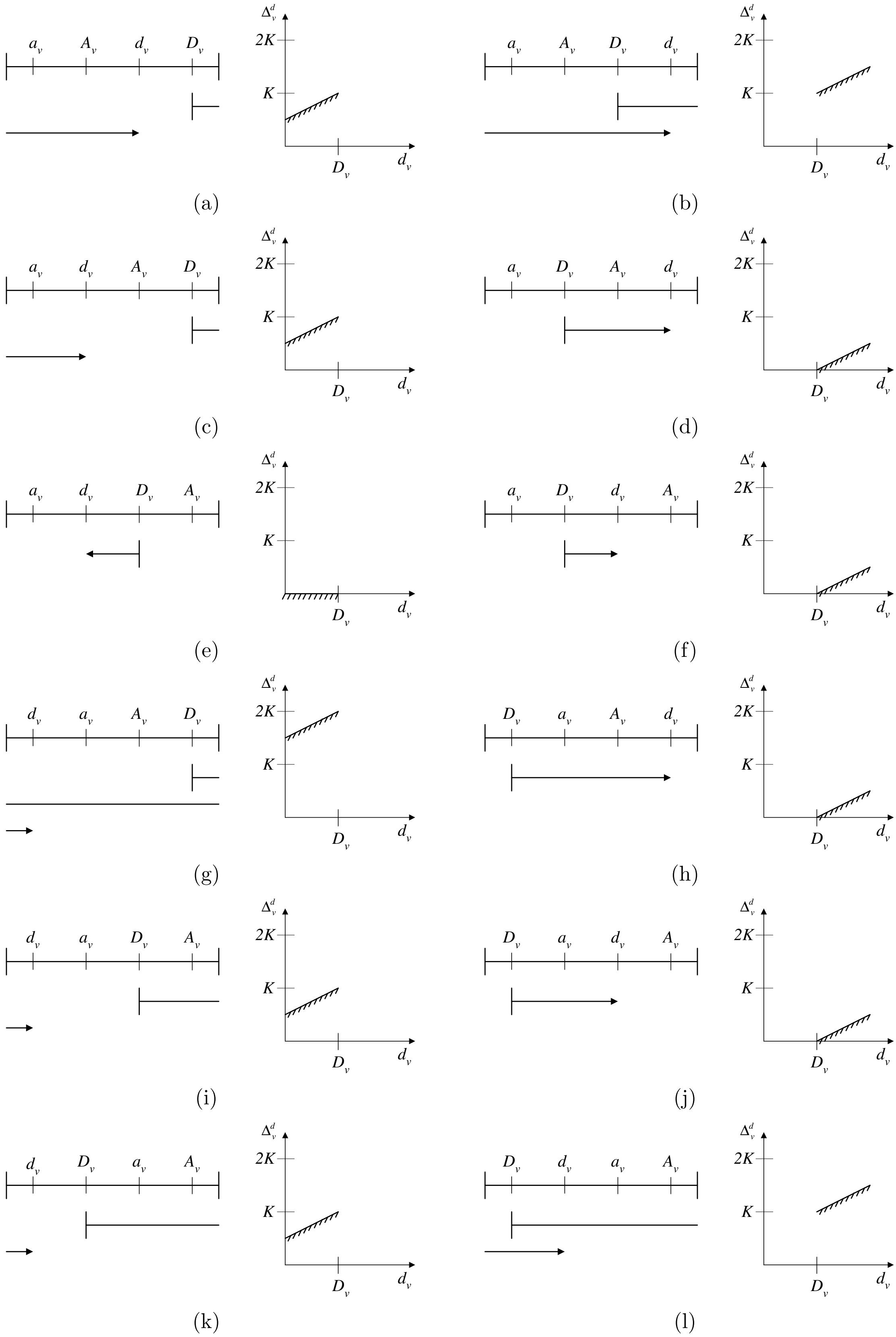


Figure 3.4: Twelve permutations for the sequence of a_v, d_v, A_v and D_v in the cycle and the corresponding delay in departure Δ_v^d.

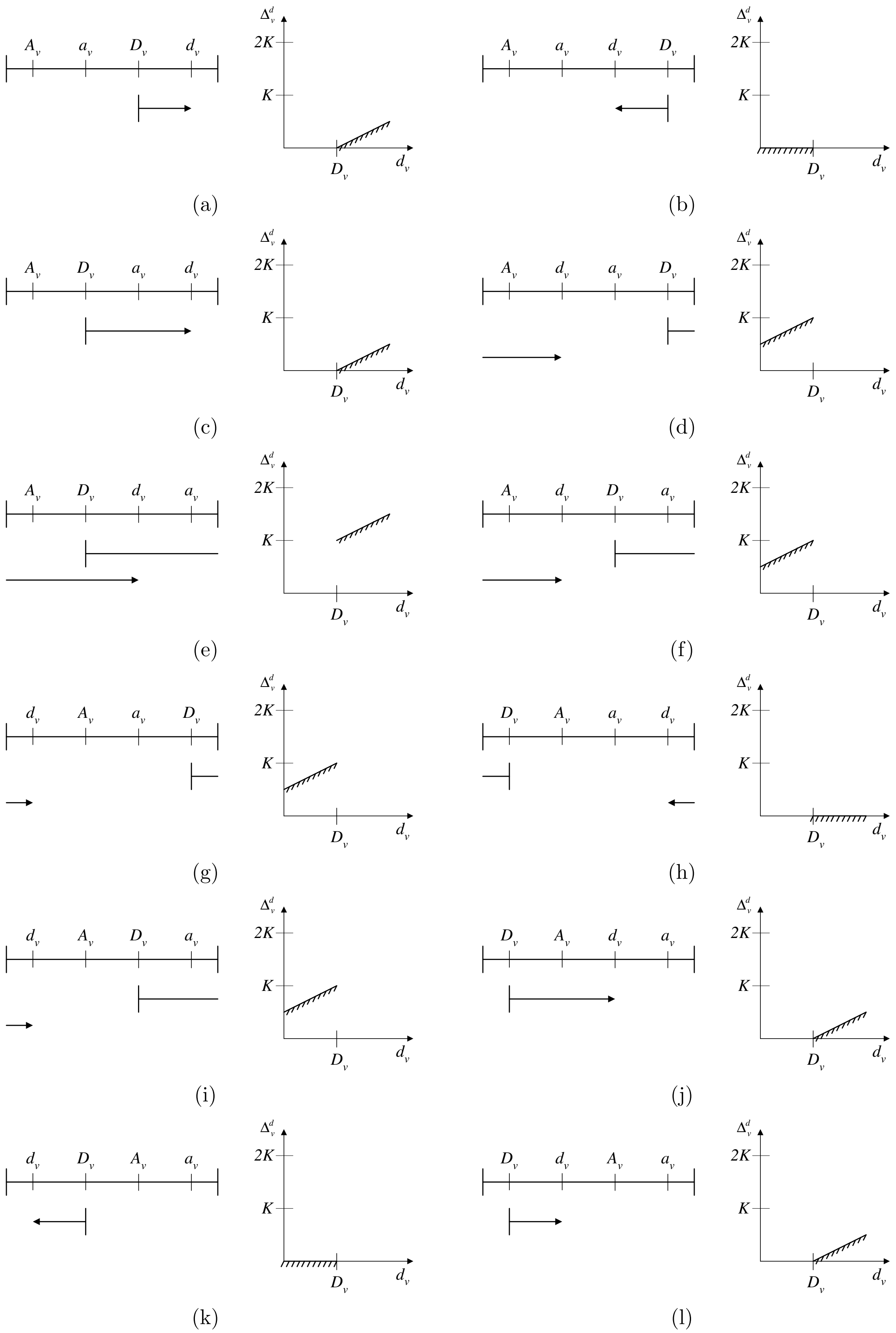
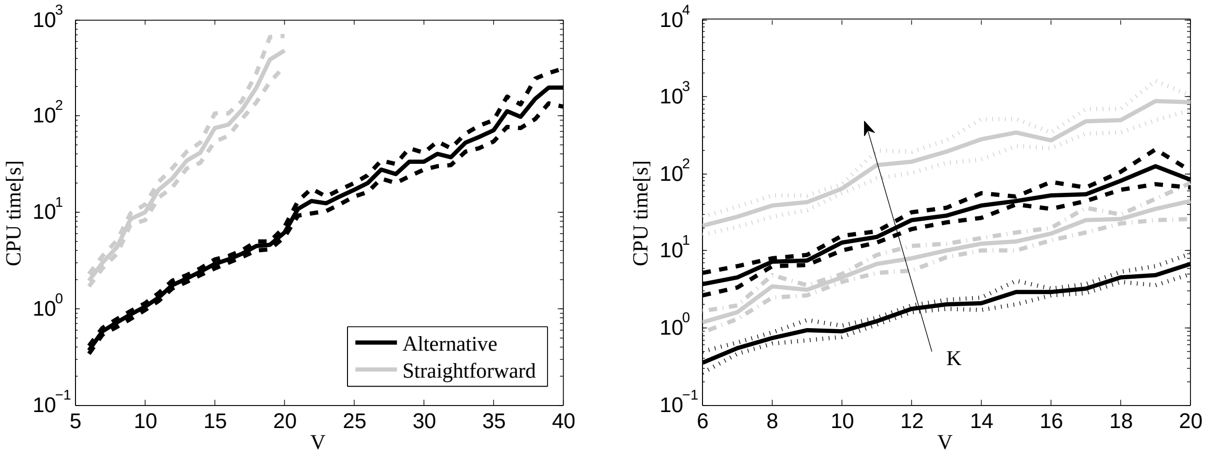


Figure 3.5: Twelve permutations for the sequence of a_v, d_v, A_v and D_v in the cycle and the corresponding delay in departure Δ_v^d.

3.3.1 Performance Analysis

In this section, we first compare the CPU time of both approaches dependent on the number of vessels in the set. We consider a problem with a cluster of three terminals $T = 3$ and a one-week-cycle, where each time slot has a width of one day, hence $K = 7$. The rest of the parameters, as described in Section 3.1, are randomly generated. The parameter set is used as input data and, together with the model, fed into CPLEX. The mixed integer optimization is terminated as soon as it has found a feasible integer solution proven to be within 5% of optimal. For each value of $V \in \{6, 7, 8, ..., 40\}$, we randomly (within reasonable bounds following from practice) generate 60 parameter sets and solve both approaches for these instances. Furthermore, for each of these optimizations, we monitor the time CPLEX's CPU time. Figure 3.6a shows the corresponding geometric means and 95% confidence intervals of the CPU time dependent on V for both approaches. Due to relatively large CPU times when using the straightforward approach, V is varied between six and twenty only in this case. The relative difference between the optima of both approaches for each parameter set are found to be within the 5% optimality gap, which suggests that indeed both models describe the same problem.



(a) Straightforward versus Alternative approach.

(b) Alternative approach for $K \in \{7, 14, 21, 42\}$.

Figure 3.6: CPU time analysis.

From Figure 3.6a it is obvious that the alternative approach significantly outperforms the straightforward approach: First of all, the mean CPU time of the alternative approach is significantly shorter than the mean runtime of the straightforward approach for each point in the considered interval. Moreover, the fraction between the CPU time of the straightforward approach and the CPU time of the alternative approach increases exponentially with the number of vessels in the set. Furthermore, the confidence interval of the straightforward approach starts diverging at a smaller V value than the confidence interval of the alternative approach. We now assume that the order of diverging remains the same for larger V values. A real-life instance, where the number of vessels in the set is forty, then could CPLEX take weeks or even months if the straightforward approach is applied and only a couple of minutes if the alternative approach is applied. Therefore, in the remainder of this chapter, only the alternative approach is used.

The final goal is to consider a real-life case and to improve its current allocation by applying the alternative approach. In order to generate a more and more detailed schedule, the width of a time slot should be decreased, at the expense of larger CPU times. Nowadays, planners of such container operations construct a schedule composed of time slots with 2 or 4 hours width. Therefore, we are interested in the CPU time dependent on the number of time slots K in the cycle. Again, we consider a port with three terminals $T = 3$ and a one-week-cycle. Additionally, for each value of $V \in \{6, 7, 8, ..., 20\}$, we generate 10 parameter sets and solve the model for $K \in \{7, 14, 21, 42\}$, sequentially. The geometric means and the 95% confidence intervals are shown in Figure 3.6b, where K increases along the direction of the arrow. As expected, the CPU time increases as K increases. Furthermore, the four graphs for the geometric means exhibit approximately the same slope, which suggests that, independent of K the CPU time grows with the same exponential rate as V increases. Additional experiments suggest that a problem of real-life size ($V = 40$) can be solved within about 6000 seconds for the case with 42 time slots (for the one-week-cycle, the width of a time slot is then 4 hours). This suggests that with the alternative approach a rather accurate allocation can be constructed for a real-life problem within a couple of hours.

3.3.2 Feasibility

Although the proposed method allocates a terminal, a number of quay cranes and a time interval of berthing to a vessel, the actual position within that terminal as well as the actual quay cranes processing that vessel are not specifically generated (see also Figure 1.5). This cut allows to solve the model relatively fast, at the expense of possible infeasibility of the found solution on an operational level. If the terminals would continuously be utilized against their quay lengths capacities, this could lead to a situation where (4.11) is fulfilled, however a feasible two-dimensional packing solution does not exist. In practice however, ports require a significant utilization margin to compensate for disturbances (e.g. late arrivals and departures) on an operational level. Results in Chapters 4 and 6 suggest that strategic allocations, generated in the first optimization step, always yield feasible position and quay crane allocation in the following optimization steps.

3.4 Case study

We consider three interacting terminals ($T = 3$) in the port of Antwerp, where thirty-seven vessel lines have one of their vessels processed exactly once a week ($V = 37$). Furthermore, we assume that each vessel line has a preferred terminal and a preferred arriving and departure time, which fit best to their schedule. The current policy of PSA HNN is commercially driven and aims to satisfy these preferred allocation as good as possible. The induced costs for the required number of quay cranes and inter-terminal traffic are of lower priority in this policy.

We are interested in potential reductions in required crane capacity and inter-terminal traffic if we allow small modifications to a representative allocation, constructed while applying the policy of satisfying shipping lines' preferences. A simple visual analysis (without any computation at all) of the port learns the following: each terminal shows heavy fluctuations in the workload distribution over the shifts of the week. At busy shifts, the entire quay crane capacity is totally occupied, whereas at quiet shifts not one crane is working at all. A second observation is the relatively large amount of transshipment containers that have to be transported from one terminal to another due to the allocation of connecting vessels to different terminals. Both these phenomena result from the discussed policy of the terminal operator.

As an illustration, Figure 3.7 depicts the scaled number of quay crane activities in one of the terminals for each hour in a a week cycle ($K = 168$) according to an allocation, which is representative for the situation in Antwerp. The black line represents the scaled mean crane capacity usage per hour. From the high fluctuations in crane usage, the following can be concluded: i) during a couple of time slots, a high amount of the crane capacity is needed and ii) during a lot of time slots a large percentage of this amount is simply not used.

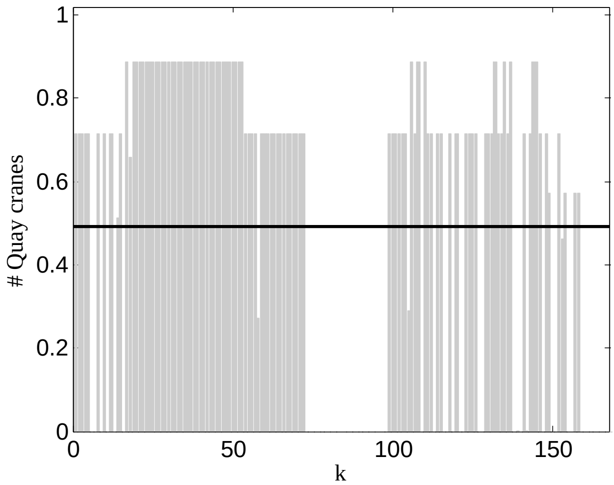


Figure 3.7: Current quay crane usage in terminal 1 during the hours of a one-week cycle.

We are interested in the benefit of flexibility, i.e. the potential reduction of the required number of quay cranes and the costs for inter-terminal transport when both the current terminal and time allocation are allowed to be adapted within certain bounds. We slightly expand the alternative MILP, such that for an arbitrarily chosen level of flexibility, the required quay crane capacity and inter-terminal transport is minimized.

Nowadays, the allocations in ports are constructed on a one or two hour(s) time grid. Since these sizes of time slots lead to large computation times when solving the MILP, the optimization is performed in two steps: First, the MILP is solved for time slots of eight hours. For a weekly cycle this means $K = 21$. The model can then be solved within minutes while generated allocations are still quite accurately, since port employees work in shifts and vessels commonly berth during multiples of a shift. In a second step, a similar MILP is built to refine the constructed allocation per terminal to a one-hour time grid. Solving this MILP takes less than a tenth of a second.

The resulting two-step optimization approach enables us to efficiently investigate the dependency of the cost savings on the level of flexibility: a number of vessels and the maximal modification in their berthing interval can arbitrarily be selected. Next, these settings can easily be implemented in the model to determine possible savings in quay crane and inter-terminal transport costs. In Section 3.4.1, we discuss the two-step optimization in detail and apply it in two experiments presented in sections 3.4.2.2 and 3.4.3.

In the first experiment, our hypothesis is that a slight adaptation of the allocation of only a couple of vessels already leads to large cost reductions. Hence, the terminal allocation of a small part of the vessels is chosen to be flexible and the time position of the berthing interval can be shifted between some pre-specified bounds. For different values of these bounds, we construct Pareto frontiers of the number of required cranes versus the inter-terminal transport costs, which confirm the hypothesis. One of the points from one of the Pareto-frontiers is highlighted to further illuminate the hypothesis. This point represents the allocation after selecting less than one third of the vessels and allowing i) a change in their terminal allocation and ii) a shift in the time position of the berthing interval of these vessels of maximal 24 hours ($G_v = 24$). The results of the two-step optimization suggest that about 25% of the number of quay cranes can be saved while at the same time the inter-terminal transportation costs are reduced by 3%.

In the second experiment, we assume that all vessel lines prefer to stay with their current time position of the berthing interval ($G_v = 0$), while they allow a different terminal allocation. Consequently, the terminal allocation of all vessels is chosen to be flexible while the time positioning of the berthing interval is fixed to the existing one. Results suggest that with the same number of quay cranes as in the current allocation, 40% of the costs for inter-terminal transport can be saved by adapting the terminal allocation.

3.4.1 Two-step approach

Step 1

We want to minimize the number of quay cranes required for the current throughput, and at the same time reduce the costs for inter-terminal transport by adapting the current terminal and time allocation. We expect that changing the current terminal allocation of a couple of vessels and slightly shifting their current berthing interval in time already leads to a significant reduction in both these objectives. Hence, we select 11 vessels (out of the total set) from the busy peaks arbitrarily and define two sets: the set $\mathcal{S}$ to be the set of 11 vessels, which current terminal and berthing allocation can be adapted, and the set $\mathcal{F}_{ij}$ to be the set of index pairs of all the other vessels together with their current terminal. We have to specify additional constraints for the vessels in the different sets. First of all, for the vessels in $\mathcal{F}_{ij}$, the time-position of the berthing interval should be fixed:

$$a_i = A_i, \qquad i \in \mathcal{F}_{ij} \tag{3.53}$$

$$d_i = D_i, \qquad i \in \mathcal{F}_{ij} \tag{3.54}$$

Furthermore, the current terminal allocation for the vessels in $\mathcal{F}_{ij}$ should be fixed:

$$x_{ij} = 1, \qquad \forall i, j \in \mathcal{F}_{ij}, \tag{3.55}$$

The vessels in set $\mathcal{S}$ on the other hand are free to be allocated to any of the terminals according to (3.30). Additionally, we allow some freedom in the time allocation of their berthing interval, while the length of the berthing interval remains equal to P_v according to:

$$\sum_{k=1}^{K} b_v(k) = P_v, \qquad \forall v, \tag{3.56}$$

where $b_v(k)$ is 1 iff vessel v berths during time interval $[k, k+1\rangle$ and 0 otherwise. Furthermore, the time position of the berthing interval of the vessels in $\mathcal{S}$ can be placed maximally G_v time slots earlier or G_v time slots later with respect to its current time position as shown in Figure 3.8.

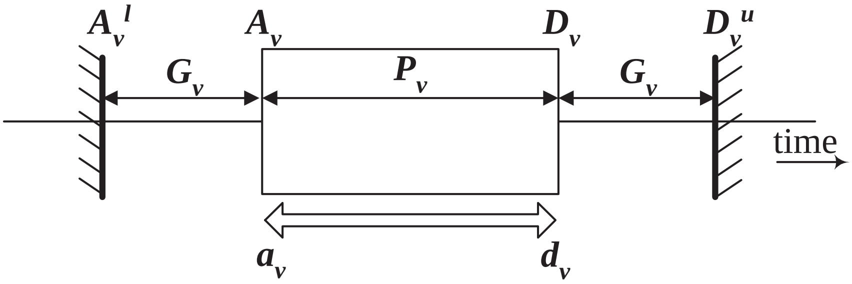


Figure 3.8: Flexibility in the time-position of the berthing interval P_v of vessel $v \in \mathcal{S}$.

The corresponding lower and upper bounds for berthing of vessel $v \in \mathcal{S}$, A_v^l and D_v^u respectively, are then given by:

$$A_v^l = \begin{cases} A_v - G_v & \text{if } A_v - G_v \geq 1, \\ A_v - G_v + K & \text{otherwise} \end{cases} \qquad \forall v \in \mathcal{S}, \tag{3.57}$$

$$D_v^u = \begin{cases} D_v + G_v & \text{if } D_v + G_v \leq K, \\ D_v + G_v - K & \text{otherwise} \end{cases} \qquad \forall v \in \mathcal{S}. \tag{3.58}$$

Due to the cyclic property of the system, we distinguish two cases and derive appropriate constraints for the vessels in $\mathcal{S}$ accordingly:

1. $A_v^l < D_v^u$:

$$e_v = 0, \tag{3.59}$$
$$a_v \geq A_v^l, \tag{3.60}$$
$$d_v \leq D_v^u. \tag{3.61}$$

2. $D_v^u < A_v^l$ (additional binary variables e_v^a and e_v^d are introduced):

$$K \cdot e_v^a \geq A_v^l - a_v \tag{3.62}$$
$$-K \cdot (1 - e_v^a) \leq A_v^l - a_v \tag{3.63}$$

$$K \cdot e_v^d \geq D_u^l - d_v \tag{3.64}$$
$$-K \cdot (1 - e_v^d) \leq D_u^l - d_v \tag{3.65}$$

$$d_v \geq a_v + K \cdot e_v^a - K \cdot e_v^d. \tag{3.66}$$

Experiments show that, with the resulting MILP, allocations on a grid of eight-hours time slots are generated within minutes. Further decreasing the width of the time slots turns out to lead to large computation times. Hence, as a first step, the model is built of eight hours time slots. The actual berthing time is therefore rounded up to a multiple of a shift to express the parameter P_v of vessel v in the model.

Step 2

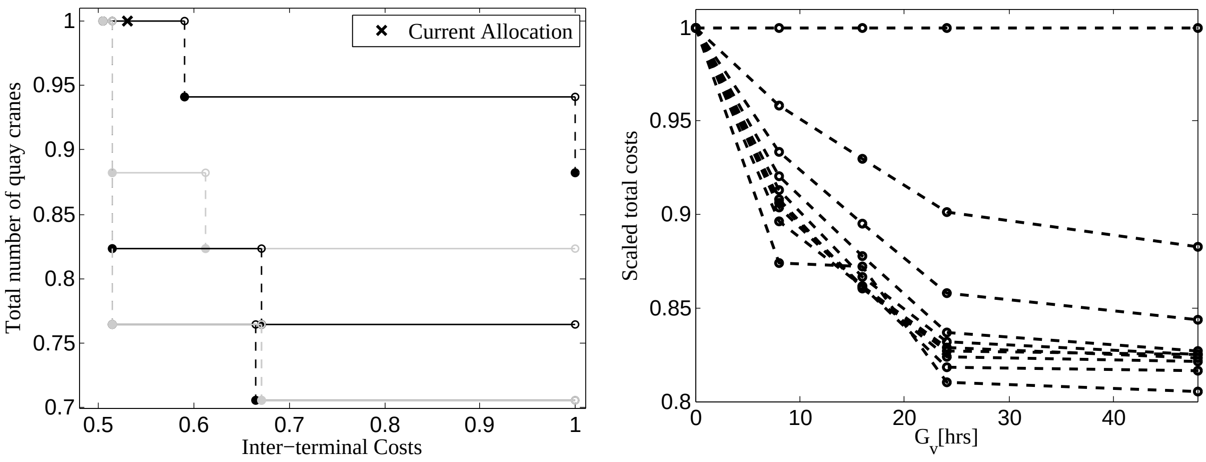
Since in today's ports most allocations are constructed on a two-hours or even on a one-hours grid, we introduce a second step to refine the constructed allocation per terminal from eight-hours time slots into one-hour time slots. We define p_v^n to be the berthing time of vessel v on the refined time grid and thus $p_v^n \leq P_v$. Additionally, we require that the refined berthing time interval is positioned between the allocated arrival and departure time (optimal values $a_v^\star$ and $d_v^\star$ of the first step optimization) on the coarse time grid. Hence, we introduce the variables a_v^n and d_v^n to be the arrival and departure time of vessel v on the refined time grid, respectively. In the second step, we build an MILP with similar constraints as given in (3.59) through (3.66), where A_v^l is substituted by $a_v^\star$, D_v^u by $d_v^\star$, a_v by a_v^n, d_v by d_v^n, and e_v^a and e_v^d by e_v^{na} and e_v^{nd}, respectively. Since the vessels are already distributed among the terminals, the remaining objective is to minimize the required crane capacity. The computation time turns out to be less then a second per terminal.

3.4.2 Experiment 1

For each value of $G_v \in \{0, 8, 16, 24, 48\}$ (in hours) for the vessels in $\mathcal{S}$, a Pareto frontier of the total number of required quay cranes versus the costs for inter-terminal transport is constructed. Each point in a frontier results from a single two-step optimization with a specific ratio between costs for quay cranes and costs for inter-terminal transport. The results are depicted in Figure 3.9a. The cross represents the state of the allocation currently applied in Antwerp. From Figure 3.9a the following can be concluded:

- For $G_v = 0$ for $v \in \mathcal{S}$ yet the costs for inter-terminal transport or the number of required quay cranes can be reduced. Apparently, an adaptation in the terminal allocation of the vessels in $\mathcal{S}$ suffices to achieve this.
- For $G_v = 0$ for $v \in \mathcal{S}$ yet a reduction in the number of quay cranes is possible at the expense of higher inter-terminal costs.
- The improvements going from $G_v = 0$ to $G_v = 8$ are relatively large, whereas the improvements going from $G_v = 24$ to $G_v = 48$ are approximately zero.
- All fronts intersect (the upper left point) where the crane costs are zero. Apparently, the inter-terminal costs cannot be further reduced even if G_v grows and the maximum number of cranes is used.
- The grey bullet suggests that if $G_v = 24$ for $v \in \mathcal{S}$ the number of required quay cranes can be reduced by almost 25% and the costs for inter-terminal by about 3%. This means that besides a possible change in terminal allocation, the time allocation of only 11 vessels has to be shifted one day maximally to gain significant improvements.

The quay crane usage of the allocation, represented by the grey bullet in Figure 3.9a, is depicted in Figures 3.10b, 3.10d and 3.10f. The results are scaled to the quay crane usage in the current allocation as shown in Figures 3.10a, 3.10c and 3.10e. The black lines represent the mean quay crane usage per hour in the different terminals. If we compare Figures 3.10a, 3.10c, 3.10e with Figures 3.10b, 3.10d, 3.10f, the following can be noticed:



(a) Pareto frontiers.

(b) Costs vs flexibility level G_v for different cost ratios.

Figure 3.9: Improvements of introducing a flexibility level.

- The workload in the generated allocation is better balanced than in the current allocation. This results in the previously mentioned reduction of almost 25% of the required quay cranes.

- At some points in time still some quay cranes are not working in the generated allocation. Introducing either a higher level of flexibility (by increasing G_v, $v \in \mathcal{S}$) or including more vessels into the set $\mathcal{S}$ would probably fill up these gaps and lead to an even better workload balance and a smaller number of required cranes.

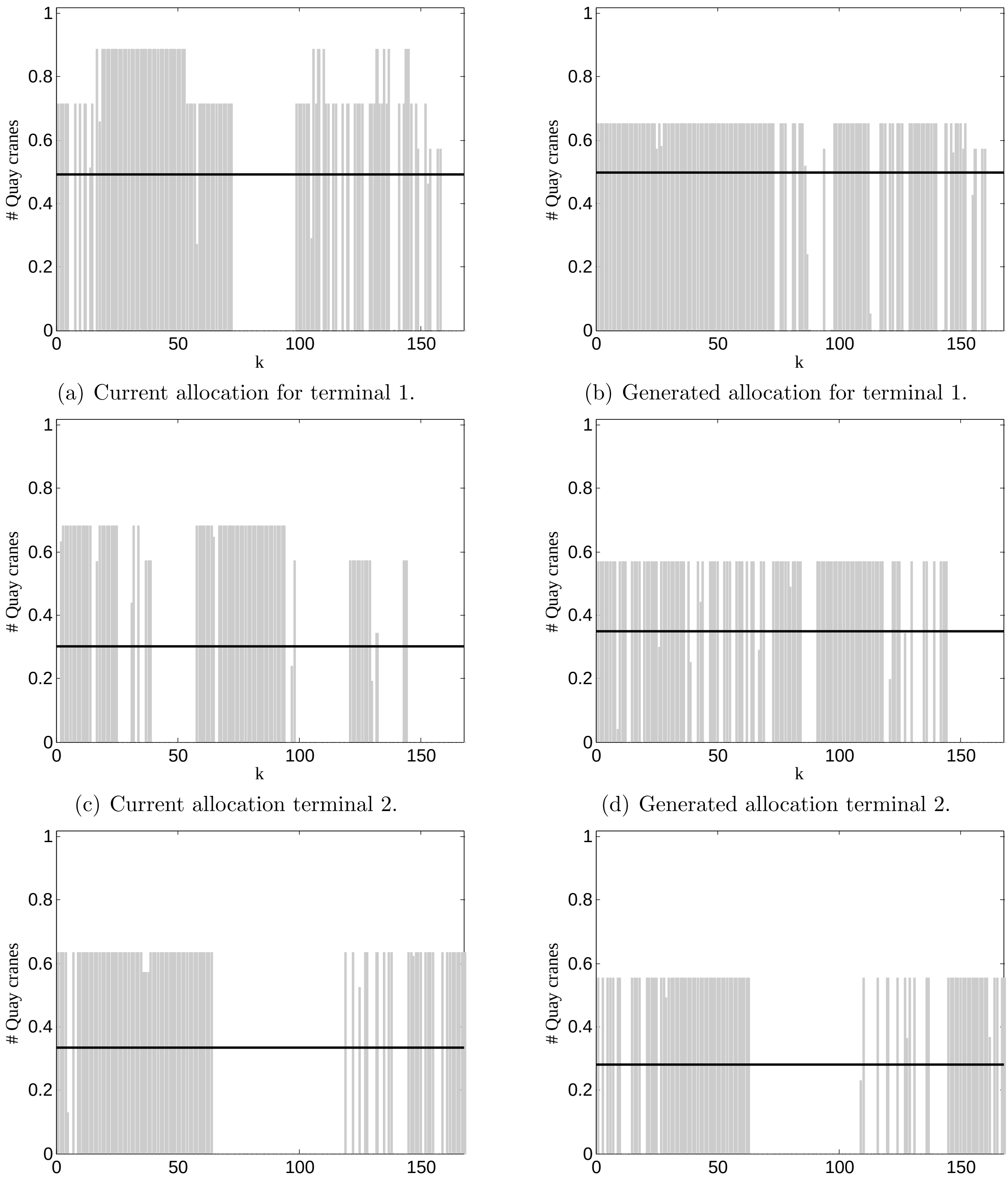
- The mean quay crane usage in a specific terminal can differ for the current allocation and the generated allocation. This can be explained by a difference in terminal allocation of the vessels in $\mathcal{S}$. The total quay crane usage however is equal for both allocations.

Additionally, we depict the benefit of modification in a different way. For a constant ratio of quay crane costs and inter-terminal costs, the scaled total costs are plotted versus the level of flexibility G_v. Figure 3.9b presents the results for 10 ratios ($\frac{c_t}{c_{pr}} \in \{0, 20, 40, 60, 80, 100, 120, 140, 160, \infty\}$). From this figure, the following is noticed:

- For the ratio equal to 0 (no costs for quay cranes), the total costs (for this ratio only inter-terminal costs) are not affected as the level of flexibility increases. Apparently, the inter-terminal costs are not affected by G_v, because the terminal length and crane capacity are not binding at $G_v = 0$, so that the 'best' (lowest inter-terminal traffic) solution is already obtained.

- For all ratios larger than zero the total costs decrease as the level of flexibility increases.

- As the ratio grows between zero and forty, the total costs-decrease becomes relatively large. Apparently, these are the sensitive ratios, where a slide increase of the crane costs already leads to a significant reduction in quay cranes. This suggests that the current number of quay cranes is far from minimal.



(a) Current allocation for terminal 1.

(b) Generated allocation for terminal 1.

(c) Current allocation terminal 2.

(d) Generated allocation terminal 2.

(e) Current allocation for terminal 3.

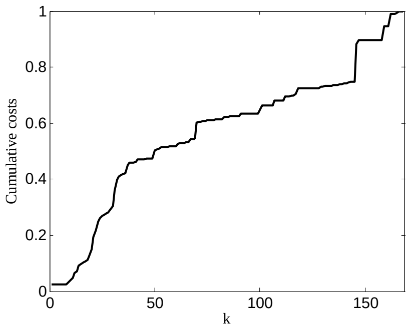
(f) Generated allocation for terminal 3.

Figure 3.10: Current and generated quay crane usage during the hours of a one-week cycle.

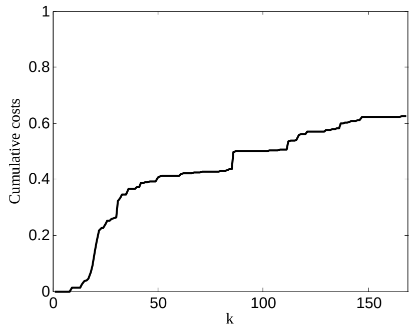
- After the cost ratio of forty, the scaled cost curve approximately stays in steady-state. For these ratios, the relative quay crane costs are that large, that further increasing them does not significantly affect the costs.

3.4.3 Experiment 2

In this experiment we assume that none of the vessel lines is prepared to change their berthing times in Antwerp, however a change in terminal allocation is allowed by all lines. We are interested in decreasing the current costs for inter-terminal transport, while the current berthing times are remained. Additionally, we require that the number of quay cranes needed is at most equal to the number of quay cranes required for the current allocation. Hence, we allow a terminal adaptation for each vessel and fix its berthing interval in time to the current allocation ($G_v = 0$). Figure 3.11 shows the cumulative costs for inter-terminal transport for the current allocation and the generated allocation for each hour in the weekly cycle. The costs are scaled to the total costs in the current allocation. These results suggest that, with the same number of quay cranes, about 40% of the costs for inter-terminal transport can be saved.



(a) Inter-terminal costs current allocation.



(b) Inter-terminal costs generated allocation.

Figure 3.11: Current and generated cumulative inter-terminal costs for a one-week cycle.

3.5 Conclusions and Recommendations

In this chapter, we considered a port consisting of a number of inter-acting terminals, operated by one terminal operator. The strategic problem faced by the operator is about allocating vessels to a terminal for a certain time interval. We abstracted from position and quay crane allocation (problems faced at tactical and operational levels, respectively) and constructed an efficient mixed integer linear program (MILP) to strategically allocate a terminal and a time window to each of the vessels in the cycle. Moreover, results suggested that an alternative MILP formulation decreases the CPU time of a straightforward MILP formulation several orders of magnitude for real-life instances. In fact, real-life problems can be solved within a couple of hours when the alternative approach is used.

Additionally, a two-step optimization approach was introduced by means of adapting the MILP. This two-step approach was applied to perform a case study, based on representative data provided by a multi-terminal container operator in the port of Antwerp. The approach enabled us to efficiently investigate the benefit of modifying an existing allocation, i.e. the potential crane and inter-terminal transport cost savings if the existing terminal and time allocations were to be adapted. Pareto frontiers were presented to give insights in the possible reduction of quay cranes at the expense of higher inter-terminal transportation costs, and vice versa. Results suggest that a small adaptation of an existing allocation suffices to gain significant improvements: a reduction of almost 25% of the number of cranes and at the same time a reduction of more than 3% of the inter-terminal costs. Furthermore, if the current terminal allocation of all vessels is allowed to be adapted while the current time allocation is fixed, costs for inter-terminal transport can be reduced by 40%. Whether such a reallocation is practically and commercially feasible is beyond the scope of this chapter.

The strategic problem considered in this chapter is a very interesting one from a practical point of view. A terminal operator that handles a number of terminals in one port faces the problem of distributing its customers (shipping lines) among these terminals, such that i) the workload is proportionally balanced over the terminals and over time, and ii) the amount of inter-terminal transport induced by transshipment containers is minimized. This chapter proposed a method to solve this strategic problem by abstracting from actual berth positions and crane scheduling and focusing on terminal and time allocations. Once this strategic timetable is generated, several subsequent problems are to be solved. We make the following outline for these problems in the next chapters:

- Usually, a contract between a terminal operator and a shipping line stipulates that the terminal operator has to process a vessel within a certain time, provided that the vessel arrives within a certain agreed time window around the scheduled arrival time. In Chapter 4 of this dissertation, a model is developed that slightly modifies the timetables, constructed in the current chapter, to increase their robustness to the within window arrivals.

- In Chapter 5, the robust terminal timetables generated in Chapter 4 are considered to be a given. The remaining tactical problem is to find appropriate positions for the vessels at the quay and positions for the containers in the yard such that the carrier travel distance between quay and yard is minimized.

- The subproblems solved in Chapters 3 through 5 result in a reference timetable and yard lay out. At an operational level however, the considered processes, e.g. arrival of vessels and quay crane productivity, are stochastic. Due to these stochastic properties, the reference allocation has to be continuously adapted. In Chapter 6, a model predictive control (MPC) is proposed to adapt the allocation under observed disturbances.

With respect to the research performed in this chapter, the following recommendations are given:

- In the current chapter, we assumed all vessels to arrive exactly once during the cycle, implying that all vessels have the same period. However, in practice it can happen that some vessels have different cycle lengths. An extension of the model, which incorporates this phenomenon, would be an interesting study.

- In today's ports, inter-terminal transport is not only established by trucks, but also by barges. In the current approach we model the resource utilization of the barges by simply reducing the quay lengths by 200 meters and dedicating one quay crane in each terminal to barge operations. It is worth investigating the trade-off between the amount of inter-terminal transport by trucks and barges and therefore worthwhile extending the current model with the actual loading and unloading of barges.

Chapter 4

Generation of a Robust Timetable for a Container Terminal

4.1 Introduction

In Chapter 3 of this dissertation, a timetable is constructed per terminal while assuming deterministic vessel arrivals and hence ignoring disturbances on arrivals. In practice however, container vessels might arrive respectively earlier or later than their scheduled arrival time, due to all kinds of events during travel (e.g. tailwind, storms, technical problems). To cope with these disturbances, the terminal operator and each of the vessel lines agree on an *arrival window*, which is placed around the scheduled arrival time. The arrival window concept distinguishes between two kinds of arrivals: arrivals within and out of the predetermined window. If a vessel arrives within its window, the terminal operator has to process this vessel within an agreed process time. If a vessel arrives out of its window, the terminal operator is not bound to any process time. Nevertheless, he aims to serve the vessel as soon as possible, but without sacrificing the agreements for other vessels.

A plan constructed in Chapter 3 might, for particular arrival scenarios within the windows, i) not yield a feasible operational plan due to lack of quay meters or ii) require a large amount of crane capacity to fulfill the window agreements. We are interested in constructing a berth plan, which is robust to all arrival scenarios where vessels arrive within their windows. In our definition, a berth plan is robust with respect to a given set of arrival windows if a feasible solution exists for each arrival scenario where all vessels arrive within their windows. The price for achieving this robustness [3] is then the additional crane capacity reservation that is required in the worst case arrival scenario where all vessels arrive within their windows. The problem is hence to construct a window-based berth plan (denoted as WB-plan for the remainder of this chapter) that minimizes the maximally required crane capacity for all scenarios where vessels arrive within their arrival windows.

4.1.1 Related Work

To deal with stochastic disturbances in dense transportation schedules, two (complementary) approaches are gaining more and more attention as stated in [7]: i) disruption management, which is concerned with operational recovery after a disruption, and ii) pro-active robustness, which builds in buffer times and other characteristics into strategic or tactical timetables to absorb disturbances and thus to prevent delay propagation through a schedule.

So far, most studies on disruption management are conducted for airline operations. However, over the last years, this approach is gaining more and more attention in railway applications as well. An overview of disruption management approaches for airline operations and the way these approaches now enter railway applications is given in [7]. In Chapter 6, a model predictive control approach is applied to online manage the logistics activities in a container terminal.

A few studies on pro-active robustness in airline scheduling can be found. The authors in [7] and [1] address a number of robustness ideas. Of particular interest is the approach of adding slack between connected flights in [29]. Flight schedules are often that tight that in case of a small plane delay, passengers might miss their connecting flight. Adding more slack between the flights is beneficial for the passengers but reduces the productivity of the airline fleet. The authors propose an MILP in which both a flight's arrival time and the departure time of its connecting flight(s) can be scheduled somewhere within a window. Each possible arc between a time slot in the arrival window and a time slot in the departure window is called a copy. Each copy implies a connecting travel time and, as determined from historical data, induces a probability of passengers missing their connected flight (if the travel time exceeds the connecting time). Given a set of flights within a restricted amount of time, the objective is to select exactly one copy for each pair of connected flights such that the expected total number of delayed passengers is minimized.

With respect to pro-active robustness in railway applications, a few approaches can be found [5], [50]. The authors in [50] consider a stochastic optimization model for the macro-level for building in time buffers between connecting trips based on arrival and departure distributions for each train. They propose a model, which allocates a restricted amount of time supplement to a number of trips to minimize the expected total amount of delay. Experimental results suggest that applying slight modifications to an existing timetable can reduce the average passenger delay substantially. The study in [5] presents embeds robustness into the train timetable by allocating time windows for arrival and departures rather than single arrival and departure times. The model is stated as a flexible periodic event scheduling problem, which guarantees that any particular choice of event times within the computed intervals is feasible. The resulting window-based timetable is therefore robust to disturbances within these bounds. The total sum of interval lengths can be traded off against the total sum of travel times in a bi-objective optimization formulation.

In this chapter on container vessel planning, we also build in robustness by reserving quay meters and crane capacities to satisfy the agreements in each arrival scenario within the windows. Our results as well suggest that slight modifications to a representative plan yield significant reductions in the maximal amount of crane capacity reservation. A major difference with airline and rail operations however is the following: passengers can enter a plane or train by themselves, but cranes are required for discharging and loading vessels. Besides satisfying the window agreements, an additional goal for container operators is therefore to reduce the maximal amount of crane capacity ever required. Our model constructs a berth plan that minimizes the maximal crane capacity reservation and still satisfies the agreements for all arrival scenarios within the windows.

The common berth allocation problem is concerned with allocating container vessels in space and time. Different solution methods and different objectives are addressed in previous studies [8], [23], [24], [22], [17], [19], [27], [32], [38], [41], [51]. However, these studies do not take arrival uncertainty into consideration while constructing the berth plan. Although such a berth plan might be optimal with respect to a certain objective, it can be very sensitive to disturbances on arrivals. A small delay of a single vessel for instance might be propagated and even amplified through the entire schedule, making it hard or even impossible to recover on an operational level.

To the best of our knowledge, only one study [36] addresses the stochastic berth allocation problem by building in pro-active robustness. Arrival times of an existing plan are given and cannot be changed. The authors derive an expression for the expected delay for each vessel based on arrival distributions. Given these expected values and a desired berthing position for each vessel, suitable time buffers and suitable berth positions are allocated to each vessel. Conflicting objectives are to minimize total expected delay, the number of overlaps of vessels and the deviations from preferred berth locations. Once a periodic berth allocation is determined, simulations with stochastic arrivals are performed. Simulations compare the performance to a model that neglects disturbances. Results suggest that taking disturbances into consideration yields a reduction in total delay on the operational level. One of the recommendations of the authors is to incorporate crane allocations while constructing a robust berth plan.

4.1.2 Contributions

We propose a mixed integer linear program, which explicitly incorporates the within-window-arrival agreements and minimizes the maximally required crane capacity. Besides the arrival times of vessels being decision variables, the model also considers time-variant crane capacity reservations per vessel to be decision variables. The model thus incorporates two flexibilities: i) shifting the berth plan of vessels in time and ii) reserving a time-variant crane capacity for each vessel. These two flexibilities enable to better balance the workload over time and hence to minimize the maximal crane capacity reservation ever required.

We arbitrarily select one of the terminals and consider its timetable as provided by PSA HNN (based on the current policy of satisfying shipping lines' preferences) as a starting point for this chapter. Computational results for applying the MILP demonstrate that with only small modifications to the berth plan of one of the terminals, a significant reduction in the maximal crane capacity reservation can be obtained. As a comparison we use the nominal plan, which ignores the arrival window agreements and thus constructs a berth plan with minimally required crane capacity assuming zero uncertainty on the arrivals. Note that the same nominal plan is also generated by step two in Section 3.4.1. Results suggest that both approaches yield significant reductions in the required crane capacity with respect to the allocation based on policy as currently applied by PSA HNN. Furthermore, results suggest that although the WB-plan requires slightly larger crane capacity reservation than the nominal plan for narrow arrival uncertainty bounds, the WB-plan requires a significantly smaller crane capacity reservation for medium and wide arrival uncertainty bounds, which are still within the arrival window bounds.

Although it is guaranteed that the terminal length is never exceeded no matter what arrival scenario occurs, the exact vessel berth position within the terminal is still to be determined. Similarly, the operational quay crane allocation is still to be determined. In this chapter, we solve a one-dimensional capacitated packing problem under arrival disturbances within the arrival windows. In Chapter 5, a joint vessel position and container stacking problem is addressed. The objective is to minimize the carrier travel distance between vessels and yard. Experiments suggest that i) the generated berth plans always yield feasible berth position allocations and ii) significant reductions in the current carrier travel distance can be achieved.

In Chapter 6, we propose a rolling horizon planning approach to recover from i) all stochastic arrivals, i.e. arrivals within but also out of the arrival windows, ii) crane breakdowns, and iii) disturbances on vessels' load compositions. In each iteration step of the rolling horizon planning process, first berthing times and time-variant crane capacities are allocated to the vessels within the horizon. Subsequently, based on these allocations, appropriate berth positions and integer-valued crane allocations are constructed.

In this chapter, we as well aim for embedding robustness into an existing berth plan. In contrast to the study in [36], the model in this chapter strives for a more sophisticated crane capacity reservation. Additionally, our model does have the flexibility to modify the scheduled arrival times and explicitly takes the agreements for arrivals within the window into consideration. With this WB-plan tool, the arrival times are chosen such that the maximal crane capacity reservation is minimized while the within-window-arrival agreements are still satisfied. As a particular case, the model constructs a nominal plan by simply reducing the arrival window size to zero, which results in a deterministic berth planning problem (equal to step two in Section 3.4.1). It is interesting to compare i) the crane capacity required in the WB-plan and in the nominal plan, and ii) the sensitivity of both plans to different arrival uncertainty bounds, which are still within the window bounds.

The outline of this chapter is as follows: In Section 2, the problem is formally phrased. Then, an MILP is proposed to construct a WB-plan with minimally required crane capacity in the worst case scenario. In Section 3, results of a case study show that with only small modifications to an existing plan already significant improvements can be achieved. In a second experiment, the performances of the WB-plan and nominal plan are compared. We end with conclusions and future work in Section 4.

4.2 Mathematical Model

In this section, an MILP is proposed to construct a WB-plan that minimizes the maximal crane capacity reservation in the worst of arrival scenarios, where vessels arrive anywhere within their windows.

4.2.1 Problem description

For all of this chapter the following holds, unless stated differently: $v \in \{1, 2, \ldots, V\}$, the set of vessels, and $k \in \{1, 2, \ldots, K\}$, the set of discrete time slots. We consider a terminal with quay length L and a set of V container vessels, where vessel v has length M_v. Each vessel is assumed to be discharged and loaded at this terminal exactly once a week. We define C_v to be the total amount of containers that has to be discharged from and loaded onto vessel v and assume this amount to be the same each week.

Dependent on the length M_v of vessel v, a maximum number S_v of quay cranes can process vessel v simultaneously. In practice, quay cranes with different processing rates are present in the terminal. We do not take the specific allocation of quay cranes to vessels into account yet, but consider an average processing rate $\bar{\lambda} \in \mathbb{N}$ for all quay cranes. The

Parameter	Definition
V	Number of vessels in the set
K	Number of discrete time slots in the periodic plan
L	Terminal quay length [m]
M_v	Length of vessel v [m]
C_v	Nr. containers to be discharged from and loaded onto vessel v
S_v	Maximal nr. quay cranes that can process vessel v simultaneously
$\bar{\lambda}$	Mean crane rate [containers/time slot]
η_v	Crane efficiency on vessel v
W_v	Width of arrival window for vessel v
$P_v^{\min}$	Minimal process time of vessel v
$P_v^{\max}$	Maximal process time of vessel v
α_v	Fraction between $P_v^{\max}$ and $P_v^{\min}$

Table 4.1: Model parameters

efficiency $\eta_v \in [0,1]$ of cranes on vessel v depends on the length M_v of vessel v; the smaller the length, the lower the efficiency. Then the minimal handling time of vessel v in terminal t depends on i) the mean processing rate $\bar{\lambda}$ in terminal t, ii) the efficiency η_v of quay cranes operating vessel v, iii) the maximal number of quay cranes that can proces vessel v simultaneously, and iv) the total number of containers C_v to be discharged from and loaded onto vessel v. The processing time of vessel v is assumed to be inversely proportional to the first three of these items and proportional to the latter. The minimal integer number of time slots P_v^{min} required to process vessel v can thus be determined as follows:

$$P_v^{\min} = \left\lceil \frac{C_v}{\eta_v S_v \bar{\lambda}} \right\rceil. \tag{4.1}$$

According to the agreements, the terminal operator has to process a vessel within the agreed process time only if that vessel arrives within its arrival window. We assume the width W_v of the arrival window for vessel v to be equal to an integer number of time slots. In practice, the process time agreed upon by the vessel line of vessel v and the terminal operator is a factor α_v larger than the minimal proces time. This we approximate by

$$P_v^{\max} = \left\lceil \alpha_v P_v^{\min} \right\rceil, \tag{4.2}$$

where $P_v^{\max}$ is the maximal number of time slots in which vessel v has to be processed only if it arrives within its arrival window, and $\alpha_v \geq 1$. Commonly, the value of α_v is significantly larger than 1, which implies that vessel v not has to be processed with the maximal number of cranes S_v permanently while berthing. Of course a vessel may also be processed faster than the $P_v^{\max}$. Furthermore, we assume $W_v + P_v^{\max} \leq K$. The model parameters are conveniently arranged in Table 5.1. The principle of the within-window-arrival agreements for vessel v is illustrated in Figure 4.1, where l_v and r_v represent respectively the left and right end of an arrival window of width W_v. If vessel v arrives somewhere within its windows, the terminal operator has to process vessel v no more than $P_v^{\max}$ time slots later. In Figure 4.1, the upper bound on the departure time is depicted for nine arrival scenarios.

The problem is to construct a periodic berth plan that is robust to the arrival scenarios where vessels arrive anywhere within their window. While robustness of this berth plan

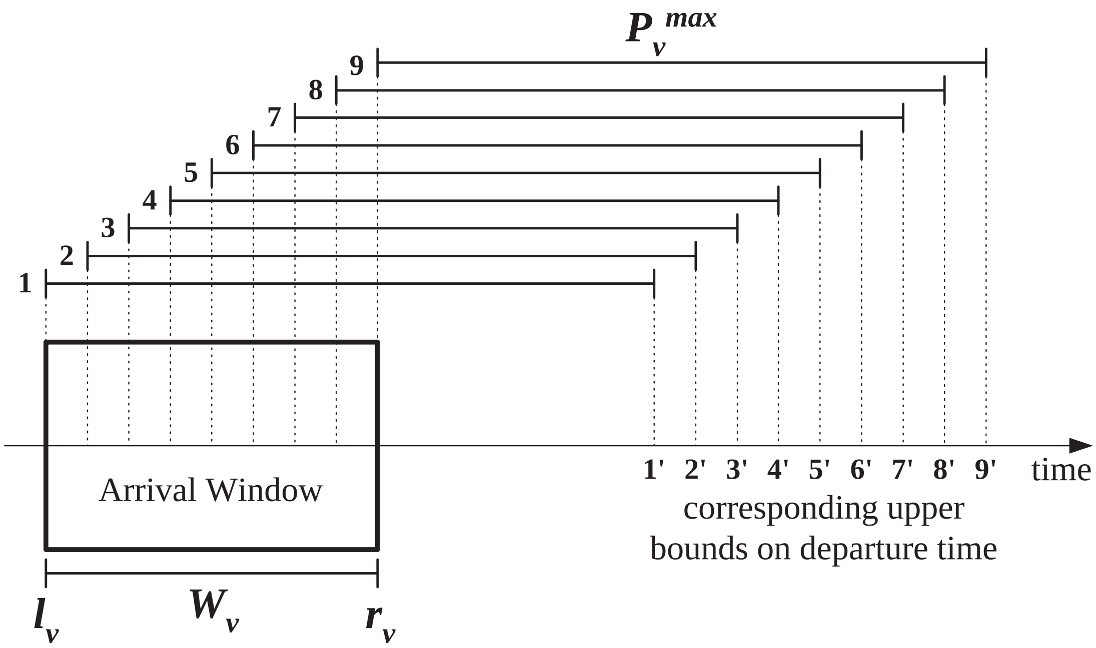


Figure 4.1: Within-arrival-window agreements.

in our definition is the property that the arrival agreements are satisfied for each of the arrival scenarios within the windows, we aim to achieve this robustness with minimum cost, that is, the crane capacity reservation required to achieve this in the worst case arrival scenario should be minimized. In the next subsection, we propose an MILP, which incorporates both these conditions.

4.2.2 MILP

Decision variables

$$
\begin{aligned}
l_v &= \text{Left end of arrival window of vessel } v \\
q_v(k) &= \text{Amount of crane capacity reserved for vessel } v \text{ during time slot } [k, k+1\rangle
\end{aligned}
$$

Auxiliary variables

$$\begin{aligned}
a_v &= \text{First time slot that quay meters and crane capacity are reserved for vessel } v \\
d_v &= \text{Last time slot that quay meters and crane capacity are reserved for vessel } v \\
b_v(k) &= \begin{cases} 1 & \text{if vessel } v \text{ can possibly berth during time slot } [k, k+1\rangle, \\ 0 & \text{otherwise.} \end{cases} \\
e_v &= \begin{cases} 1 & \text{if } a_v > d_v, \\ 0 & \text{if } a_v \leq d_v. \end{cases} \\
r_v &= \text{Right end of arrival window of vessel } v. \\
w_v(k) &= \begin{cases} 1 & \text{if time slot } [k, k+1\rangle \text{ lies within the arrival window of vessel } v, \\ 0 & \text{otherwise.} \end{cases} \\
e_v^w &= \begin{cases} 1 & \text{if } l_v > r_v, \\ 0 & \text{if } l_v \leq r_v. \end{cases} \\
m_v(k) &= \text{Amount of quay meters reserved for vessel } v \text{ during time slot } [k, k+1\rangle. \\
Q &= \text{At least the amount of crane capacity required in the worst case scenario.}
\end{aligned}$$

Constraints and objective

For vessel v the arrival window has to be positioned in time by determining the left end time position of the window l_v. From l_v the values for r_v and $w_v(k)$ follow accordingly. First of all, we have to enforce that the values of the binary variable $w_v(k)$ are equal to one if and only if time slot $[k, k+1\rangle$ lies between the left end right end of the window (see also [19]):

$$1 - l_v \leq k \cdot \big(w_v(k) - e_v^w\big) \leq r_v - 1 \qquad \forall v, k, \tag{4.3}$$

$$r_v - K \leq \big(K - k\big) \cdot \big(w_v(k) - e_v^w\big) \leq K - l_v \qquad \forall v, k, \tag{4.4}$$

$$\sum_{k=1}^{K} \big(w_v(k) - e_v^w\big) = r_v - l_v \qquad \forall v. \tag{4.5}$$

Additionally, the width of the arrival window for vessel v is fixed to W_v. Considering the discrete time model, this implies that there have to be $W_v + 1$ time slots within the arrival window of vessel v:

$$\sum_{k=1}^{K} w_v(k) = W_v + 1 \qquad \forall v. \tag{4.6}$$

For each vessel v, the earliest possible arrival and latest possible departure time (a_v and d_v, respectively) have to be decided on. Only in between its earliest possible arrival time and its latest possible departure time, a vessel might be berthing. Before its earliest possible arrival time and after its latest possible departure time, a vessel cannot berth at all. This can be formulated in a similar way as has been done for positioning the arrival window in (4.3), (4.4) and (4.5):

$$1 - a_v \leq k \cdot \big(b_v(k) - e_v\big) \leq d_v - 1 \qquad \forall v, k, \tag{4.7}$$

$$d_v - K \leq \big(K - k\big) \cdot \big(b_v(k) - e_v\big) \leq K - a_v \qquad \forall v, k, \tag{4.8}$$

$$\sum_{k=1}^{K} \big(b_v(k) - e_v\big) = d_v - a_v \qquad \forall v. \tag{4.9}$$

If time slot $[k, k+1\rangle$ is a possible berthing time slot of vessel v, M_v quay meters have to be reserved during that time slot:

$$m_v(k) = M_v \cdot b_v(k) \qquad \forall v, k. \tag{4.10}$$

The sum of lengths of all vessels possibly berthing during time slot $[k, k+1\rangle$ should never exceed the terminal length L:

$$\sum_{v=1}^{V} m_v(k) \leq L \qquad \forall k. \tag{4.11}$$

If time slot $[k, k+1\rangle$ is reserved for vessel v, the crane capacity reserved for vessel v during that time slot is limited by the number of cranes that can process vessel v simultaneously:

$$q_v(k) \leq S_v \cdot b_v(k) \qquad \forall v, k. \tag{4.12}$$

We now have to enforce that the window agreements for vessel v are satisfied. The agreements state that if vessel v arrives within its window, its process time has to be within the agreed process time $P_v^{\max}$. Hence, for each range of sequential time slots that starts from a time slot within the arrival window of vessel v and ends $P_v^{\max} - 1$ time slots later, the sum of reserved crane capacities should be sufficient to process at least C_v containers. Since the position of the window of vessel v is a decision variable on itself, we have to explicitly consider the sum of crane reservations for each possible range of sequential time slots of length $P_v^{\max}$ within the considered cycle. Only if the first time slot of such a range lies within the arrival window of vessel v, sufficient crane reservations for vessel v during these time slots are required to process at least C_v containers. To model this we make use of the value of the binary value $w_v(k)$:

$$\sum_{i=k}^{k+P_v^{\max}-1} \eta_v \bar{\lambda} \cdot q_v(i) \geq C_v \cdot w_v(k) \qquad \forall v, k. \tag{4.13}$$

The sum of reserved crane capacities of all vessels during time slot $[k, k+1\rangle$ should never exceed the maximally required crane capacity reservation:

$$\sum_{v=1}^{V} q_v(k) \leq Q \qquad \forall k. \tag{4.14}$$

The objective is to minimize the maximal crane capacity reservation:

$$\min_{l_v, q_v(k)} Q \tag{4.15}$$

4.3 Case study

The MILP proposed in the previous section determines a WB-plan with minimal crane capacity reservations in the worst case arrival scenario while within-arrival-window agreements are still met. The model thus incorporates the arrival window agreements and hence finds a plan robust to these agreements. In this section, we perform two experiments on the representative berth plan provided by PSA HNN. We consider one terminal and 15 vessels ($V = 15$), which call exactly once each week. A time slot width of one hour is chosen, so $K = 168$. We set the arrival window width of vessel v to eight hours, so $W_v = 8$, $\forall v$, which is a typical value in real-life container operations. This implies that if a vessel arrives up to four hours earlier or four hours later than its planned arrival time, the terminal operator still has to process the vessel within the agreed process time. Typical values for the other parameters in Table 5.1 are provided by PSA HNN as well. Furthermore, we assume a typical value of $\alpha_v = 1.4$, $\forall v$.

For this data set, two experiments are performed. In the first experiment, the WB-plan MILP is applied to the provided berth plan for different values of maximal modification. Results show that with small modifications, significant reductions in the maximal crane capacity reservation can already be achieved. In the second experiment, the performances of the WB-plan and the nominal plan are compared. The nominal plan can be constructed as a particular case by simply setting the arrival windows width to zero ($W_v = 0$, $\forall v$) in the MILP (or by running step two in Section3.4.1). Optimizing the MILP then results in an optimal periodic plan with a minimal crane capacity reservation. The performance of both the WB-plan and the nominal plan are compared for different arrival uncertainty bounds within the arrival windows. Results suggest that although the WB-plan requires slightly more crane capacity reservations for narrow arrival uncertainty bounds, it significantly outperforms the nominal plan for medium and wide arrival uncertainty bounds.

4.3.1 Benefit of plan modification

As mentioned before, vessels have fixed routes and a preferred arrival time in each port they call on. Negotiations have to point out whether vessel lines are willing to slightly modify their scheduled arrival times. We thus aim for a reduction in crane capacity reservation if we assume relatively small modifications to the existing plan can be made. To obtain some more insight into the improvements that can be made dependent on the extent of modification, a sequence of 4 experiments is performed. In experiment i, $i \in \{1, 2, 3, 4\}$, a vessel subset $\mathcal{V}_i$ is selected from the representative data set of PSA HNN, where $|\mathcal{V}_1| = 2$, $|\mathcal{V}_2| = 4$, $|\mathcal{V}_3| = 7$, $|\mathcal{V}_4| = V = 15$, and $\mathcal{V}_i \subset \mathcal{V}_{i+1}$, $i \in \{1, 2, 3\}$. For each of the vessels in the subset $\mathcal{V}_i$ of experiment i, we allow a maximal modification of G_v time slots with respect to the existing plan, by introducing appropriate upper and lower bounds on the arrival window position l_v. In each experiment $i \in \{1, 2, 3, 4\}$, the WB-plan MILP with $W_v = 8$, $\forall v$ is determined consecutively for $G_v \in \{0, 1, 2, ..., 8\}$, $v \in \mathcal{V}_i$.

Results are depicted in Figure 4.2. The (scaled) maximal crane capacity reservation is plotted versus the maximally allowed plan modification G_v for each v in the selected vessel subset. Each curve in this plot depicts the outcome of one experiment i. In this figure, we notice the following:

- Each of the four curves is monotonically decreasing. This makes sense, since increasing the extent of maximal modification will never yield a higher amount of maximally required crane capacity.

- Along the same line, we can explain that the curve of experiment $i+1$ never exceeds the curve of experiment i. Namely, if the plan of more vessels is allowed to be modified, the maximal crane capacity reservation will never increase.

- One interesting result is that with allowing the plan of seven vessels to be modified, the same improvements can be achieved as by allowing the plan of all fifteen vessels to be modified.
- Another interesting result is that by allowing the plan of only four vessels to be modified, at least 95% of the improvements can already be obtained as by allowing the plan of all fifteen vessels to be modified.
- By modifying the plan of four out of fifteen vessels maximally 2 hours, a reduction of about 5% in the maximal crane capacity reservation can be achieved.

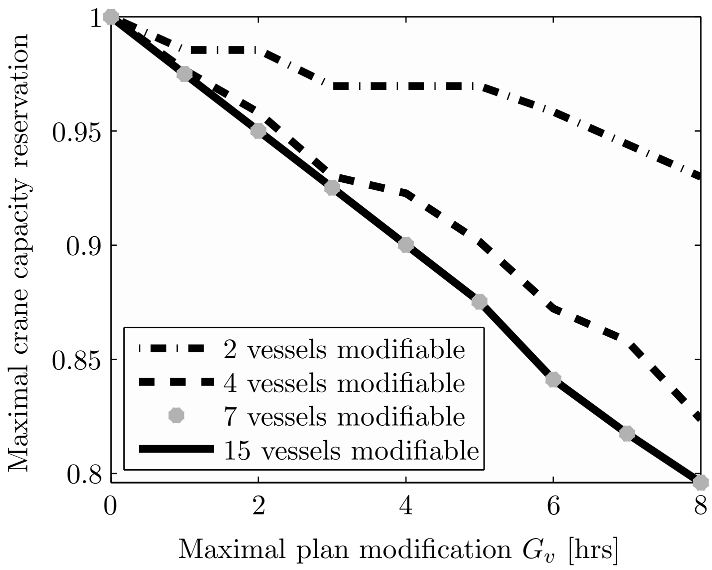


Figure 4.2: Benefit of modification.

4.3.2 WB-plan vs. nominal plan

The MILP as proposed in the previous section explicitly includes the within-arrival-window agreements for $W_v = 8$. Another approach to construct a tactical berth plan is to simply ignore the arrival window agreements $W_v = 0$ and to determine the optimal deterministic berth plan. Such a nominal plan is constructed by simply reducing the window width W_v to zero $\forall v$ and running the MILP.

We are interested in the performance of the WB-plan and the nominal plan for bounded arrivals, i.e. we assume all actual operational arrivals to be within certain bounds, which are still within the arrival window bounds. We define the time distance between the arrival uncertainty bounds to be U_v, where $0 \leq U_v \leq W_v$, $\forall v$.

Subset $\mathcal{V}_2$ of the previous experiment is chosen since for this instance $|\mathcal{V}_2| = 4$ is a reasonable guess for the number of vessels to be modifiable. The following procedures are proposed to evaluate the performance of the WB-plan and nominal plan for a particular value of U_v:

Procedure to evaluate WB-plan performance:
1. Determine a WB-plan by optimizing the MILP with a window width $W_v = 8$,

2. Record the optimal objective value Q_8^{WB},
3. Substitute the optimal values of the left end of the arrival window l_v^* into the l_v variables in the MILP and fix them,
4. Run the MILP for a window width $W_v = U_v$, $\forall v$,
5. Record the optimal objective value $Q_{U_v}^{WB}$.

Procedure to evaluate nominal plan performance:
1. Determine a nominal plan by optimizing the MILP with a window width $W_v = 0$,
2. Record the optimal objective value Q_0^{WI},
3. Substitute the optimal values of the left end of the arrival window l_v^* into the l_v variables in the MILP and fix them,
4. Run the MILP for a window width $W_v = 8$, $\forall v$,
5. Record the optimal objective value $Q_{U_v}^{WB}$.

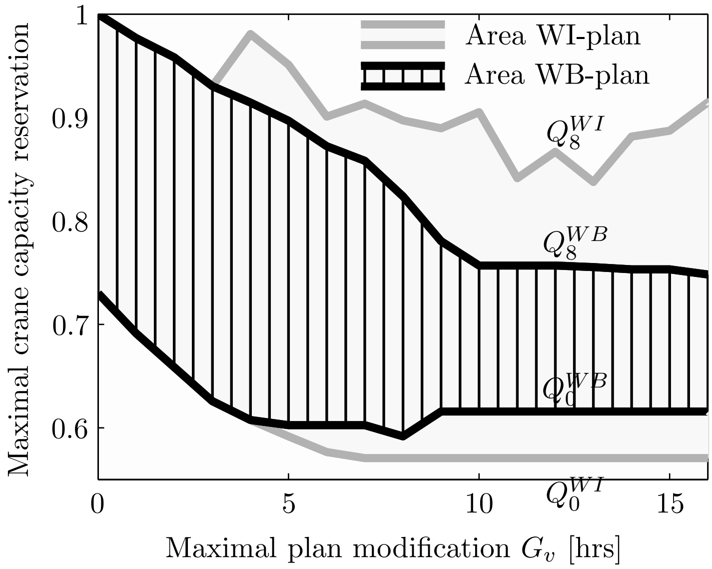


Figure 4.3: Window-Based plan ($W_v = 8$, $\forall v$) vs. Window-Ignoring plan ($W_v = 0$, $\forall v$).

For each of the values of $G_v \in \{0, 1, 2, ..., 16\}$, and $v \in \mathcal{V}_2$, these procedures are applied to determine the performance of the WB-plan and nominal plan for $U_v = 0$, $\forall v$ and $U_v = W_v$, $\forall v$. Figure 4.3 depicts Q_0^{WB}, Q_8^{WB}, Q_0^{WI}, Q_0^{WI} as a function of $G_v \in \{0, 1, 2, ..., 16\}$ $v \in \mathcal{V}_2$. In this figure we can notice the following:

- The grey and shaded area represent the limits of maximal crane capacity reservations for respectively the nominal plan and the WB-plan for values of $0 \leq U_v \leq 8$. It can be noticed that the shaded area in total lies within the grey area. Apparently, the nominal plan outperforms the WB-plan if zero disturbances on arrivals are present ($U_v = 0$), but is much more sensitive for wide arrival uncertainty bounds ($U_v = 8$).
- The curves for Q_0^{WI} and Q_0^{WB} are monotonically decreasing. This is to be expected, since these lines result from the first step optimizations (step 1 in both procedures). Hence, if the maximal plan modification increases, the maximal crane capacity reservation will never increase.
- The curves for Q_0^{WI} and Q_0^{WB} however, are not monotonically decreasing. From the procedures we notice that these curves are determined in a second optimization

(step 4) in which the values from the first optimization are already fixed. Regarding this the non-monotonic decrease makes sense, since the decisions made in the first optimization (step 1) do not necessarily have to be optimal in the second optimization (step 4), and might result in a higher crane capacity in the second optimization even when the maximal plan modification increases.

- The curve for Q_0^{WI} never exceeds the curve for Q_8^{WI}. This makes sense, since a berth plan incorporating bounded arrivals will always require at least the same amount of crane capacity as the same berth plan with deterministic arrivals. The same reasoning can be applied for the observation that Q_0^{WB} never exceeds Q_8^{WB}.

- The curve for Q_0^{WI} never exceeds the curve for Q_0^{WB}. This is to be expected since Q_0^{WI} is determined in a first optimization (step 1) and Q_0^{WB} in the second optimization (step 4). Q_0^{WI} ignores the arrival window agreements and hence is the optimal plan if no arrival disturbances are present ($U_v = 0$). Q_0^{WB} is constructed from a WB-plan, which incorporates the window agreements. This may lead to a berth plan, which is not optimal for deterministic arrivals.

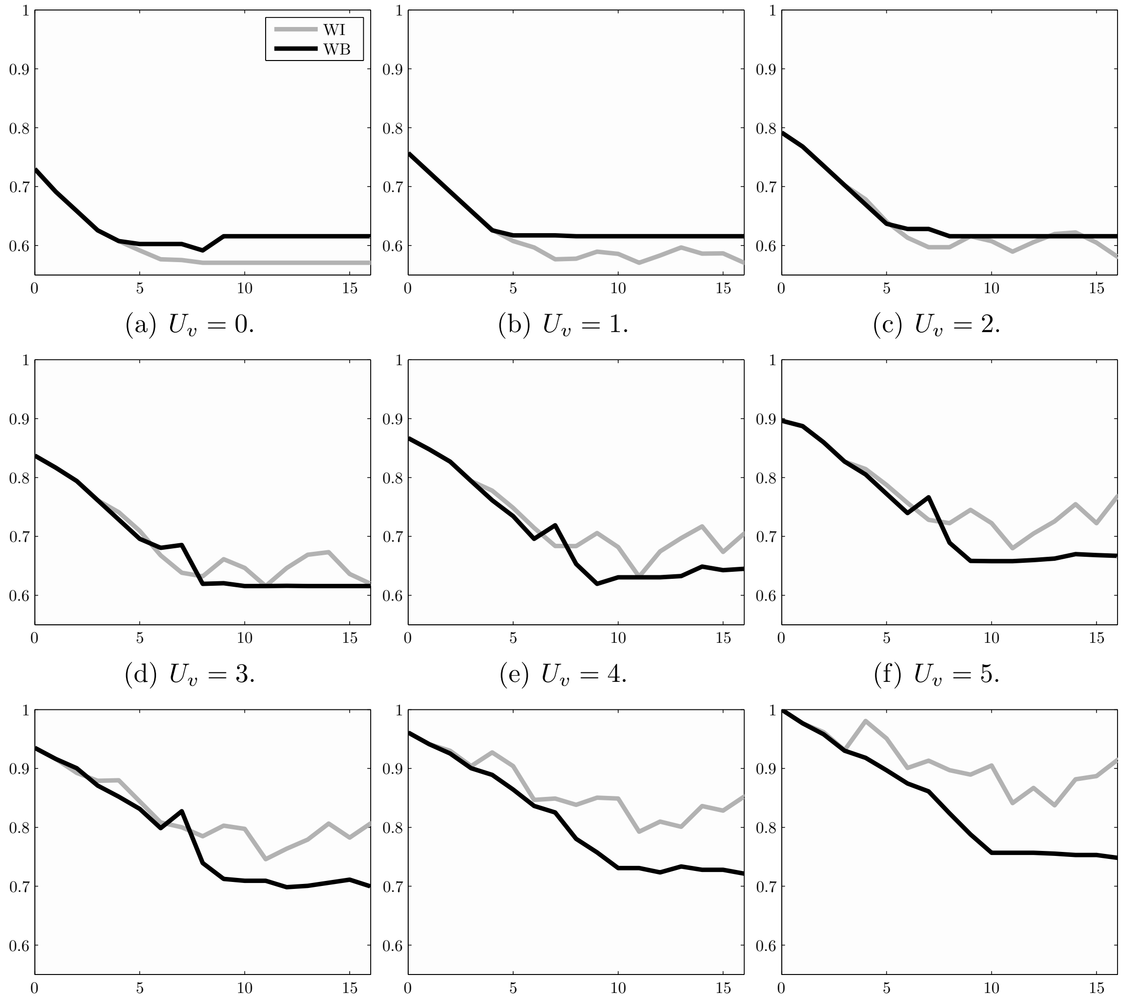
- The curve for Q_8^{WB} never exceeds the curve for Q_8^{WI}. This is to be expected since Q_8^{WB} is determined in a first optimization (step 1) and Q_8^{WI} in the second optimization (step 4). Q_8^{WB} incorporates the arrival window agreements and hence is a robust plan if arrival disturbances are present on an operational level. Q_8^{WI} is constructed from a nominal plan, which ignores the window agreements during optimization. Ignoring these agreements may lead to a plan which is not robust to arrival disturbances.

- The initial values of Q_0^{WI} and Q_0^{WB} are equal. This is to be expected since in this case $G_v = 0\ \forall v$ and hence no modifications to the berth plan can be made (l_v is fixed). An intelligent time-variant crane capacity allocation for the given plan results in the same amount of maximal crane capacity reservation when arrivals are assumed to be deterministic. The same reasoning can be applied for the observation that the initial values of Q_8^{WI} and Q_8^{WB} are equal.

- The initial values of Q_0^{WI} and Q_0^{WB} on one hand, and Q_8^{WI} and Q_8^{WB} on the other are not (necessarily) equal, although they are all based on the same berth plan (l_v is fixed to the same value in each of these cases). Still, this observation makes sense, since in the former two cases (Q_0^{WI} and Q_0^{WB}) deterministic arrivals are assumed and in the latter two cases (Q_8^{WI} and Q_8^{WB}) arrival uncertainty bounds (equal to the window bounds) are assumed. Arrival bounds will result in at least the same amount of crane capacity reservation than in the case with deterministic arrivals.

As mentioned before these curves depict only the limits on the performance for all values where $0 \leq U_v \leq 8$. We are interested in the dependency of the nominal plan and the WB-plan performance on specific values of U_v within this range. Hence, we applied additional experiments to evaluate the performance of the nominal plan and the WB-plan for $U_v \in \{1, 2, 3, 4, 5, 6, 7\}$, $\forall v$. Results for $U_v \in \{0, 1, 2, ..., 8\}\ \forall v$ are depicted in Figure 4.4. In this figure, we notice the following:

- Figures 4.4a and 4.4i again present respectively the lower and upper limits on the crane capacity reservations for stochastic arrival uncertainty bounds within the window bounds (see also Figure 4.3). As U_v increases from zero to 8, both the curves



(a) $U_v = 0$. (b) $U_v = 1$. (c) $U_v = 2$.

(d) $U_v = 3$. (e) $U_v = 4$. (f) $U_v = 5$.

(g) $U_v = 6$. (h) $U_v = 7$. (i) $U_v = 8$.

Figure 4.4: Performance WB and nominal plan for values of U_v. Horizontal axes depict the maximal plan modification G_v, $v \in \mathcal{V}_2$, vertical axes the maximal crane capacity reservation.

for the nominal plan and the WB-plan shift upwards. This makes sense, since the wider the bounds on the arrivals grow, the more crane capacity reservations are required.

- For $U_v = 0$, the WB-plan requires a slightly higher amount of crane capacity than the nominal plan. As U_v increases from zero to two, the difference between the performance decreases towards zero.
- As U_v increases from three to eight, the performance of the WB-plan with respect to the nominal plan increases significantly.
- Apparently, the WB-plan based on a window size of $W_v = 8$, is more robust to medium and wide arrival uncertainty bounds ($U_v \geq 3$) than the nominal plan.

4.4 Conclusions and Recommendations

We considered a set of container vessels that has to be discharged and loaded in a container port by a terminal operator on a periodic basis. Disturbances on travel times lead to stochastic arrivals in the port. To cope with these disturbances, the terminal operator agrees on a so-called arrival window for each vessel rather than a single arrival time. Only if a vessel arrives within its window, the terminal operator has to process this vessel within an agreed process time. If a vessel arrives out of its window, the terminal operator is not bound to any process time.

We proposed an MILP to construct a window-based periodic berth plan (WB-plan) with minimally required crane capacity in the worst case arrival scenario, i.e. an MILP that minimizes the required crane capacity while the agreements for all scenarios where vessels arrive within their windows are still satisfied. Experiments on a representative berth plan provided by the terminal operator PSA HNN in Antwerp, Belgium, suggested that with small modifications to the representative plan, already significant reductions in the maximally required crane capacity reservation can be achieved.

As a particular case, our MILP approach constructs a window-ignoring periodic berth plan (nominal plan) by setting the arrival window width to zero. This nominal plan is similar to the one generated in the previous chapter. We investigated the performance of the WB-plan and the nominal plan for deterministic arrivals and bounded arrivals within the arrival window bounds. Results suggested that although the WB-plan requires slightly more crane capacity than the nominal plan for narrow arrival uncertainty bounds, it is much more sensitive to medium and wide arrival uncertainty bounds, which are still within the windows.

In the first optimization step (see Figure 1.5 and Chapter 3), a nominal timetable is constructed for a multi-terminal container operation. In this chapter, a method is proposed that takes an arbitrary (e.g the nominal) berth plan as a starting point and makes it more robust to the within window arrival agreements. In Chapter 5, the timetables are assumed to be a given and accordingly, berth positions for the vessels and yard positions for the containers have to be allocated. The objective is to minimize the total carrier travel distance between vessels and yard. Finally in Chapter 6, the generated berth plan and yard layout are considered to be a reference planning for the considered cycle. A model predictive control approach is applied to act upon all kinds of disturbances in the daily operations.

To better quantify the robustness of a timetable, actual arrival distributions of the vessel lines are required. Having these, the expected amount of required crane capacity can be computed. Moreover, with these distributions we can compute the percentage of time during which the maximum capacity is required.

Another recommendation emerges from the observation that the proposed MILP determines an upper bound on the maximally required crane capacity. Namely, it does not explicitly evaluate each possible arrival scenario, but instead reserves sufficient crane capacity such that the window agreements are always met. This approximation significantly reduces the number of scenarios to be evaluated. A recent study investigates for small instances the deviation between our determined upper bound and the optimal value for the maximally required crane capacity.

Chapter 5

Joint Berth Allocation and Yard Design Problem

5.1 Introduction

In Chapter 3, a multi-terminal container port, run by one terminal operator, is considered and terminals and berth times are allocated to a set of vessels assuming deterministic parameters. Subsequently in Chapter 4, the constructed timetable of a terminal is slightly modified to increase its robustness to stochastic arrivals. In this chapter, we consider the (robust) timetable as given and accordingly determine appropriate berth positions for the vessels at the quay.

After unloading and before loading, containers are temporarily stacked in capacitated areas in the storage yard of a terminal. Several types of containers have designated stacks in the yard, while others can be stacked wherever space is available. The allocation of these containers in the stack is defined as the yard design problem. To guarantee consistency for the carrier drivers and yard planners, the allocation of the containers from a certain shipping line to a specific stack should not change too frequently. Decisions on the reference berth positions and yard design are therefore only reconsidered once each year or once each two years.

The decisions on the berth positions and the yard design determine the total distance that has to be covered by carriers operating between quay and yard. In this chapter, we address the joint problem of allocating i) vessels' berth positions, taking into account no overlapping of vessels, and ii) container stack positions, taking into account non-exceeding of stack capacities, such that the total carrier distance is minimized.

5.1.1 Related Work

As mentioned in Section 3.1, the berth allocation problem (BAP) has been investigated extensively over the last decades. The problem involves the allocation of container vessels in time and space in order to minimize a certain objective function. In most of the reported studies, the objective is to minimize each vessel's turnaround time [15], [35], [24], [8], [27]. A limited number of studies, considers a multi-objective problem, where besides the turnaround times, the weighted deviations from predetermined berth positions are minimized [16], [51]. The authors in these studies consider reference berth positions, which are chosen closest to given positions of containers in the yard. In this way, the distance that has to be covered by container carriers between quay and yard is tried to be reduced.

Of particular interest is the study in [36]. The authors mention that the total travel distance strongly depends on the stack positions of containers in the yard. They assume that all containers for/from a certain vessel are stacked closest to the position where the particular vessel berths and approximate the total travel distance by the travel distances for transshipment containers between connecting vessels. In practice however, two main reasons exists to contradict that all containers can be stacked close to a vessel's berth position. First of all, special container types (like reefers, IMCO's and empty containers) have designated areas in the yard and hence cannot be stacked arbitrarily. Second of all, once a stack is filled up to its capacity, containers have to be allocated to surrounding stacks, unavoidably inducing additional travel distances for carriers. In this chapter, we take both these issues into account while solving the joint problem of allocating berth positions to vessels and stack positions to containers to minimize the total carrier travel distance.

The above mentioned studies all consider the allocation of time and space to vessels within one terminal. Actually, in an increasing number of ports (e.g. Antwerp, Singapore and Rotterdam), one terminal operator is responsible for a number of terminals. Along the same line, we consider a multi-terminal container operation run by the terminal operator PSA HNN in Antwerp, Belgium. As mentioned in Section 3.1, the problem then becomes to i) allocate a vessel to a terminal (actually the loop to which a number of vessels belong), ii) allocate a time interval to a vessel for berthing, iii) allocate a vessel to a berthing position, and iv) allocate quay cranes to a vessel, taking into account the cyclic nature of vessel calls.

The BAP in such ports can no longer be considered per terminal for two main reasons. One is that it makes sense to avoid peaks and troughs in quay crane utilization and to spread vessel calls evenly over the various terminals. The other is that transshipment containers will unavoidably generate inter-terminal traffic, whose costs should be taken into account. All possible flows of containers through such a port are depicted in Figure 1.6. Since the overall problem is too complex to be solved within satisfactory time, it is cut into several subproblems. These subproblems are chosen such that they are practically interesting and at the same time can be solved within the time allowed at the considered level. We have chosen to cut the overall container operation problem into four subproblems (see Figure 1.5).

The first is a strategic problem (Chapter 3), i.e. which terminal and which time window to allocate to each (vessel in the) loop. This subproblem is addressed occasionally when contracts for new loops are negotiated or existing contracts are renegotiated. The algorithm to solve this problem optimally may run for several hours, if not days. Additionally, as a second problem, the robustness of the generated timetable per terminal is increased by taking within arrival window agreements into account. The problem of adding robustness to the timetable is addressed in Chapter 4 and can be solved within minutes.

Given these (robust) timetables per terminal, the problem left on the tactical level is where to berth the vessels along the quay. We have to emphasize that this problem is typically different from the BAP studies mentioned above for two reasons. One is that the berth interval at this level is no longer a variable of the problem, it is a given. Secondly, the objective is not to minimize the make-span of each vessel, but to optimally use the time allowed to service the vessel. In this chapter, we consider the constructed (robust) timetables to be given and address the tactical problem of where to berth vessels taking into account their lengths, and where to stack their containers taking into account the various types of containers and the storage capacity of each stack. While doing so we try to minimize the horizontal carrier travel distance between quay and yard by taking into account fixed yard positions for some special container types, and by taking into account the transshipment matrix of the vessels, i.e. which vessels carry how many containers that are to be loaded onto connecting vessels.

The generated tactical timetable, berth positions and yard design derived in Chapters 3 through 5, are considered to be the reference allocation for a longer period of time (a year or two). In the daily operation however, the system is disturbed (e.g. due to storm, tailwind, technical problems) and the terminal operator faces the problem of rescheduling the disrupted system to serve the shipping lines at minimal costs. A model predictive control approach is developed in Chapter 6 to on-the-fly allocate actual i) start and end times for processing, ii) berth positions and iii) quay cranes to vessels in a container terminal under disturbance.

5.1.2 Contributions

As has been mentioned, the timetables per terminal are considered to be a given. Furthermore, we consider four types of containers (reefer, IMCO, empty and full). Additionally, for each call we assume the expected number of each of these types to be given (both for import and export). Usually, a terminal operator uses designated areas for reefer, ICMO, and empty containers, which we assume to be given as well. The stack positions for the full containers are still to be chosen and considered as variables in this problem.

Hence, we address the joint problem of allocating berth positions to vessels at the quay and stack positions of vessels' containers in the yard to minimize the total carrier travel distance. While doing so we ensure that vessels that are berthing simultaneously do not overlap and we ensure that for each moment in time stacks are never filled up above their capacities.

We propose an appropriate mixed integer quadratic program, which turns out to be non-convex and consequently complex from a computational point of view. The constraints however are convex and separable in the two decision variables being i) the vessels' berth positions and ii) the amount of containers flowing from a particular vessel to a particular stack. This enables to construct two individual problems, an MILP and an LP, being i) the vessel position allocation problem and ii) the container stack allocation problem respectively, which are coupled in the objective function. A solution technique, that continues alternating between both problems, appears to converge to a local optimum very fast (the alternating procedure between two separable LP's, coupled in the objective, is proven to converge to a local minimum in [4]). The performance of the alternating procedure however turns out to heavily depend on the initial condition. To find a proper initial guess, we propose an MILP, which is an approximation of the earlier proposed MIQP, since it allocates fixed-sized groups of containers to stack, rather than variably-sized groups. Using this initial condition leads to a solution that outperforms results from random initial conditions.

A case study on a representative data set, provided by the terminal operator PSA HNN, learns that the proposed method is very efficient and yields a reduction of more than 20% in the total carrier travel distance compared to the distance for the provided berth plan and yard lay out. In Section 5.2, the problem is formally phrased and the parameter set is given. Additionally, the solution approach is discussed. Section 5.3 presents the case study and shows results. We end with conclusions and recommendations in Section 5.4.

5.2 Approach

5.2.1 Problem Description

The relevant quay and yard operations for this problem are depicted in Figure 5.2, which is a suitable representation of an actual container terminal in the port of Antwerp. The terminal has (quay) length Q and (yard) width B. The yard is divided into N stacks, in the figure $N = 24$ (8x3), where stack n has a capacity of C_n, $n \in \{1, ..., N\}$ containers. The position of stack n is defined by the (x, y) coordinates of its center (X_n, Y_n), where the origin O is positioned at the lower left end of the terminal. A number of V vessels berths at the quay periodically, where vessel v has length L_v and can be positioned anywhere along the quay. Since the arrival and departure times of the vessels are given, the pairs of vessels that berth simultaneously can be determined in advance. The vessel pair (i, j) is added to the mathematical set $\mathcal{S}$ if vessel i and vessel j berth simultaneously according to the given timetable. As an example, Figure 5.1 depicts the set $\mathcal{S}$ for an arbitrary timetable of an arbitrary vessel set.

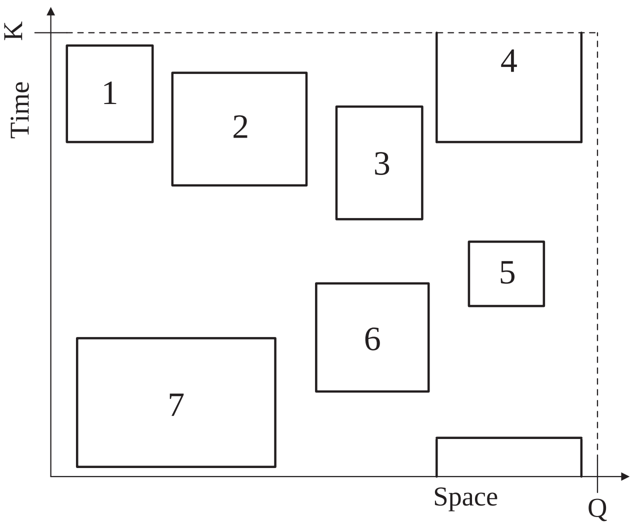


Figure 5.1: Set of pairs $\mathcal{S}$={(1,2),(1,3),(1,4),(2,3),(2,4),(3,4),(4,7),(5,6),(6,7)} for given time allocation. Note that vessel 4 arrives at the end of a cycle and departs at the beginning of the next cycle.

We consider T container types to be transported into and out of the terminal area. Each vessel imports a number of containers of each type for destinations $\{0, ..., V\}$, where destination 0 represents the "hinterland" and destination $1, ..., V$ represents vessel $1, ..., V$, respectively. Additionally, containers of different types are brought in from the hinterland with export destinations $1, ..., V$. We assume proforma load compositions and destination matrices of the weekly calls to be known. The given timetable then learns at what time, which type of containers with a specific destination arrive in and leave from the port. We assume that vessels are totally unloaded first, before the loading starts. Moreover, the unloading and loading fractions of the given berthing time are divided proportionally to the fractions of import and export containers, where the (un)loading of different container types is distributed uniformly among the total (un)loading time. Additionally, we assume all import containers to be transported out of the yard eight hours after they are stacked and all export containers to be present in the yard eight hours in advance of the corresponding vessel's arrival.

Having these assumptions and considering discrete time $k \in \{1, ..., K\}$ (where K is the cycle length), the container flows into and out of the terminal as a function of time are completely determined and captured in the following parameters: $I^t_{ij}(k)$ is the amount of containers of type t flowing into the yard from the hinterland $i = 0$ or from vessel $i \in \{1, ..., V\}$, with final destination j, $j \in \{0, ..., V\}$ during time slot $[k, k+1\rangle$. $O^t_v(k)$ is the amount of containers of type t flowing to the hinterland $v = 0$ or to vessel $v \in \{1, ..., V\}$ during time slot $[k, k+1\rangle$. The parameters are arranged in Table 5.1.

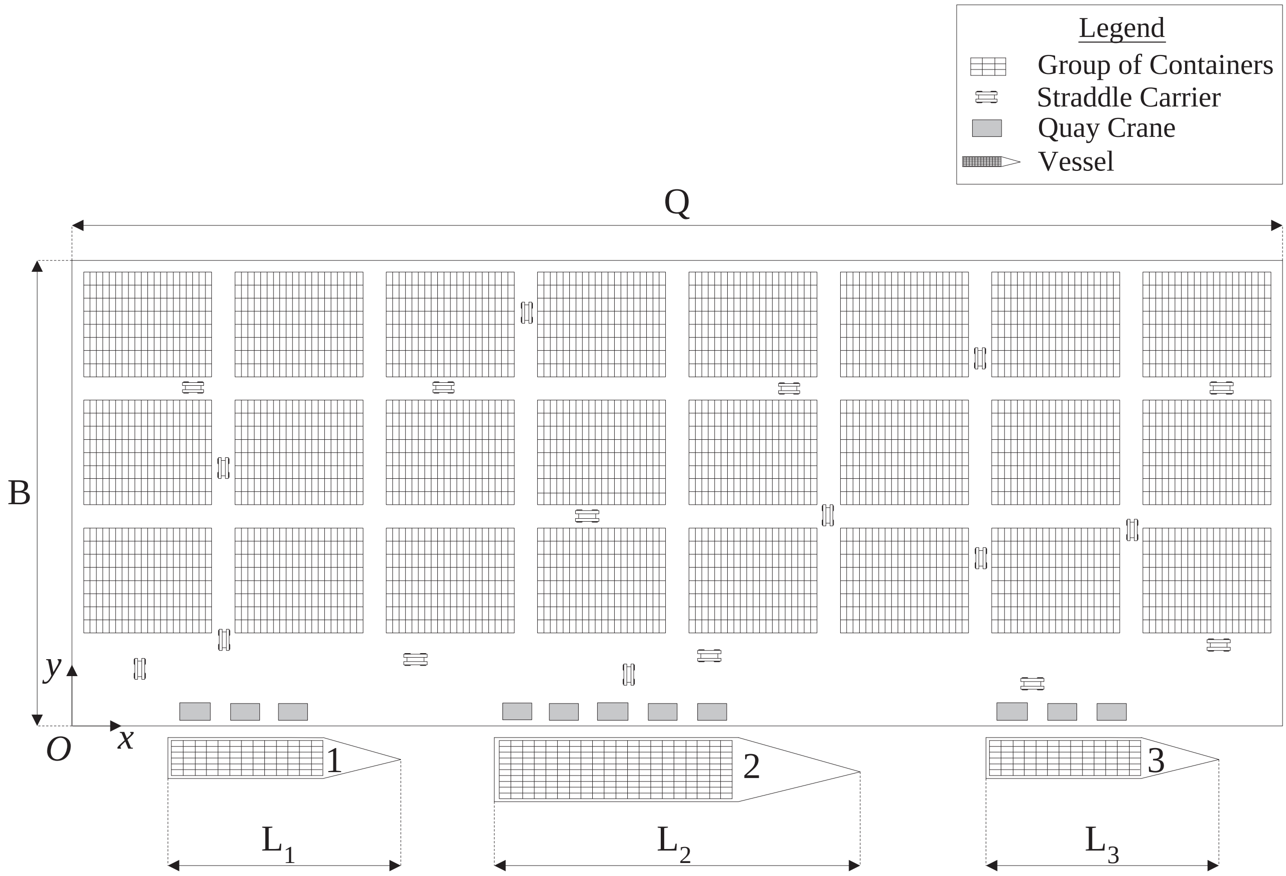


Figure 5.2: Straddle carriers operating between vessels and yard, which contains N=24 stacks.

We assume that each container is stacked somewhere in the yard in between the time it arrives in the port and the time it leaves the port again. The import and export container transport between the vessels/hinterland and the yard is performed by straddle carriers. For the distance between a vessel/hinterland and a stack we assume the principle of Manhattan distance, i.e. the distance between the vessel and the stack measured along the x and y axes. The problem is to find i) berth positions for the vessels taking into account non-overlapping, and ii) stack positions for the containers taking into account the stack capacities, such that the total carrier travel distance is minimized.

Parameter	Definition
V	Number of vessels in the set
$\mathcal{S}$	Set of pairs of vessels that berth simultaneously
N	Number of stacks in the yard
L_v	Length of vessel v [m]
K	Number of discrete time slots in the cycle
Q	Terminal length [m]
B	Terminal width [m]
X_n	x-position of stack n
Y_n	y-position of stack n
C_n	Capacity of stack n
$I^t_{ij}(k)$	# containers with source i and destination j flowing into the yard at time $[k, k+1\rangle$
$O^t_v(k)$	# containers flowing from the yard to destination v at time $[k, k+1\rangle$

Table 5.1: Model parameters

In the next section, we propose a solution technique to find a suboptimal carrier travel distance, taking into account non-overlapping of vessels and non-exceeding of stack capacities. First in Section 5.2.2, a mixed integer quadratic program (MIQP) is proposed. Since the corresponding objective is non-convex, the problem becomes complex from a computational point of view.

Next in Section 5.2.3, we split the MIQP into an MILP and an LP, which are coupled in the objective and can both be solved very fast. The problems are executed in an alternating fashion, i.e. one continues solving the problems subsequently until the objective has converged. This procedure turns out to be very fast, however experiments learn that the performance heavily depends on the chosen initial condition.

To select an initial condition in a sophisticated way (rather than randomly), an additional MILP is proposed in Section 5.2.4. Besides vessels' berth positions, this MILP allocates fix-sixed groups of containers to stacks rather than variably-sized groups, and hence generates a rough yard layout. Applying the alternating procedure suggests that starting from this initial condition, the carrier travel distance can be decreased below the best result out of an extensive number of random instances. Moreover, it reduces the total carrier travel distance in the provided allocation by more than 20%.

5.2.2 MIQP

Decision variables

$$
\begin{aligned}
p_v &= \text{Position of the center of vessel } v. \\
i^t_{ijn}(k) &= \text{\# containers of type } t \text{ with origin } i \text{ and destination } j \text{ flowing to stack } n \text{ during.} \\
&\quad\ \text{time slot } [k, k+1\rangle. \\
o^t_{vn}(k) &= \text{\# containers of type } t \text{ flowing from stack } n \text{ to destination } v \text{ during time slot } [k, k+1\rangle.
\end{aligned}
$$

Auxiliary variables

$$\begin{aligned}
e_{ij} &= \begin{cases} 1 & \text{if vessel } i \text{ is positioned left of vessel } j, \\ 0 & \text{if vessel } i \text{ is positioned right of vessel } j. \end{cases} \\
z_{vn} &= \text{Manhattan distance between the center of vessel } v \text{ and stack } n. \\
s_{vn}^t(k) &= \#\text{ containers of type } t \text{ with destination } v \text{ in stack } n \text{ during time slot } [k, k+1\rangle.
\end{aligned}$$

The objective is to find vessel positions at the quay and container positions in the stack to minimize the total carrier travel distance. The carrier travel distance of a certain amount of containers between their vessel and their stack is equal to the product of i) the amount of containers and ii) the distance between the corresponding vessel and the designated stack. Hence, we have to minimize the sum over all products of container amounts and their related distances:

$$\min_{p_v, i_{vjn}^t(k), o_{vn}^t(k)} \sum_{v=0}^{V} \sum_{n=1}^{N} \Big(\sum_{t=1}^{T} \sum_{k=1}^{K} \big(\sum_{j=0}^{V} i_{vjn}^t(k) + o_{vn}^t(k) \big) \Big) \cdot z_{vn} \tag{5.1}$$

With respect to the vessel positions, the following constraints are valid:

$$\begin{array}{rcll}
p_v & \geq & \frac{L_v}{2} & \forall v \\
p_v & \leq & Q - \frac{L_v}{2} & \forall v \\
p_i - p_j & \geq & \frac{L_i + L_j}{2} - Q \cdot e_{ij} & \forall (i,j) \in \mathcal{S} \\
p_j - p_i & \geq & \frac{L_i + L_j}{2} - Q \cdot (1 - e_{ij}) & \forall (i,j) \in \mathcal{S} \\
z_{vn} & \geq & p_v - X_n + Y_n & \forall n, v \in \{1, ..., V\} \\
z_{vn} & \geq & -p_v + X_n + Y_n & \forall n, v \in \{1, ..., V\} \\
z_{vn} & = & B - Y_n & \forall n, v = 0
\end{array} \tag{5.2}$$

The first two constraints ensure that vessel v is totally positioned within the quay. The third and fourth constraints ensure the non-overlapping of vessels that berth simultaneously. The fifth and sixth constraints define the minimal Manhattan distance between vessel v and stack n. The seventh constraint defines the minimal distance between the hinterland and stack n. Note that this distance only depends on the y coordinate of stack n, which implies that containers to and from the hinterland are respectively delivered and collected at the x coordinate of their designated stack. Note that the former four constraints can be obtained after substituting the, for this problem known, values of δ_{ij}^t, $i, j \in \{1, ..., V\}$ and δ_{ij}^p, $i, j \in \{1, ..., V\}$, into the BAP formulation of [24].

With respect to the container stacking problem, the following constraints are valid:

$$\begin{array}{rcll}
\sum_{n=1}^{N} i_{ijn}^t(k) & = & I_{ij}^t(k) & \forall t, i, j, k \\
\sum_{n=1}^{N} o_{vn}^t(k) & = & O_v^t(k) & \forall t, v, k \\
s_{vn}^t(k+1) & = & s_{vn}^t(k) + \sum_{i=0}^{V} i_{ivn}^t(k) - o_{vn}^t(k) & \forall t, v, n, k \\
\sum_{t=1}^{T} \sum_{v=0}^{V} s_{vn}^t(k) & \leq & C_n - \sum_{t=1}^{T} \sum_{v=0}^{V} \sum_{i=0}^{V} i_{ivn}^t(k) & \forall n, k
\end{array} \tag{5.3}$$

The first two constraints depict conservation laws, i.e. the total number of containers of type t with origin i and destination j flowing into the yard during time slot $[k, k+1\rangle$

should be distributed among stacks $1, ..., N$, and the number of containers of type t flowing from the (different stacks in the) yard to destination v during time slot $[k, k+1\rangle$ should sum up to the total number of containers of that type with destination v during that time slot. Note that the special types of reefers, IMCO's and empty containers ($t = 2, 3, 43$, respectively) have designated stacks and hence for these containers $i^t_{ijn}(k)$ and $o^t_{vn}(k)$ are given parameters. So, the variables left in the container stacking problem are $i^t_{ijn}(k)$ and $o^t_{vn}(k)$ for $t = 1$ (full containers).

5.2.3 Alternating Optimization

The drawback of the described MIQP is that the objective function is non-convex, which makes the high-dimensional problem hard to solve. However, the constraint set concerning the vessel position allocation and the constraint set concerning the container stack allocation are convex and separable in the variables of berth positions and container stack positions, as the distinction between equations set (5.2) and (5.3) already shows. This enables to construct an MILP and an LP, which are coupled in the objective function and can be solved alternatingly. The alternating optimization procedure is depicted and explained below.

$$
\begin{array}{rrcll}
\min\limits_{p_v} & \multicolumn{4}{l}{\sum_{v=0}^{V}\sum_{n=1}^{N}\Big(\sum_{t=1}^{T}\sum_{k=1}^{K}\big(\sum_{j=0}^{V} i_{vjn}^{t}(k)^{\star}+o_{vn}^{t}(k)^{\star}\big)\Big)\cdot z_{vn}} \\
\text{s.t.} & p_v & \geq & \frac{L_v}{2} & \forall v \\
& p_v & \geq & Q-\frac{L_v}{2} & \forall v \\
& p_i-p_j & \leq & \frac{L_i+L_j}{2}-Q\cdot e_{ij} & \forall (i,j)\in\mathcal{S} \\
& p_j-p_i & \geq & \frac{L_i+L_j}{2}-Q\cdot(1-e_{ij}) & \forall (i,j)\in\mathcal{S} \\
& z_{vn} & \geq & p_v-X_n+Y_n & \forall n, v\in\{1,...,V\} \\
& z_{vn} & \geq & -p_v+X_n+Y_n & \forall n, v\in\{1,...,V\} \\
& z_{vn} & = & B-Y_n & \forall n, v=0
\end{array}
$$

$$
z_{vn}^{\star}\ \downarrow \qquad \uparrow\ i_{vjn}^{t}(k)^{\star}, o_{vn}^{t}(k)^{\star}
$$

$$
\begin{array}{rrcll}
\min\limits_{i_{vjn}^{t}(k), o_{vn}^{t}(k)} & \multicolumn{4}{l}{\sum_{v=0}^{V}\sum_{n=1}^{N}\Big(\sum_{t=1}^{T}\sum_{k=1}^{K}\big(\sum_{j=0}^{V} i_{vjn}^{t}(k)+o_{vn}^{t}(k)\big)\Big)\cdot z_{vn}^{\star}} \\
\text{s.t.} & \sum_{n=1}^{N} i_{ijn}^{t}(k) & = & I_{ij}^{t}(k) & \forall t,i,j,k \\
& \sum_{n=1}^{N} o_{vn}^{t}(k) & = & O_{v}^{t}(k) & \forall t,v,k \\
& s_{vn}^{t}(k+1) & = & s_{vn}^{t}(k)+\sum_{i=0}^{V} i_{ivn}^{t}(k)-o_{vn}^{t}(k) & \forall t,v,n,k \\
& \sum_{t=1}^{T}\sum_{v=0}^{V} s_{vn}^{t}(k) & \leq & C_n-\sum_{t=1}^{T}\sum_{v=0}^{V}\sum_{i=0}^{V} i_{ivn}^{t}(k) & \forall n,k
\end{array}
$$

We start from the upper box, which presents the berth position problem, and select initial conditions $i_{vjn}^{t}(k)^{\star}$, the amount of container type t flowing from source v into stack n during time slot $[k, k+1\rangle$ having destination j, and $o_{vn}^{t}(k)^{\star}$, the amount of container type t flowing from stack n to destination v during time slot $[k, k+1\rangle$. For these initial values, the berth position problem is solved and the suboptimal values $z_{vn}^{\star}$, following directly from $p_v^{\star}$, are passed to the lower box problem. In this box, the container stacking problem is solved for the delivered values $z_{vn}^{\star}$. The resulting suboptimal values $i_{vjn}^{t}(k)^{\star}$ and $o_{vn}^{t}(k)^{\star}$ are passed to the upper box again. This procedure is repeated until the objective value does no longer decrease.

In Section 5.3, we show that the performance of the procedure heavily depends on the chosen initial condition. In the next subsection, we derive an additional MILP to construct a proper guess for the initial values $i_{vjn}^{t}(k)^{\star}$ and $o_{vn}^{t}(k)^{\star}$. Results suggest that the solution, generated by the alternating optimization for this initial guess, outperforms all solutions, generated by the alternating optimization for an extensive number of random initial conditions.

5.2.4 Initial Condition

In this section, an MILP is proposed to construct a proper initial condition for the alternating procedure derived in the previous section. The main difference of the MILP here and the original MIQP is that the MILP allocates fixed-sized groups of containers rather than variably-sized groups of containers. Since the fixed-sized container groups are parameters in the model proposed here, a linear objective function remains (rather than a non-convex quadratic objective function). The total amount of containers of type t with source i and destination j in the cycle is considered to be one group. The remaining problem is to allocate berth positions to vessels and stack positions to the container groups such that the total carrier travel distance is minimized. Below, the MILP formulation is discussed.

Decision variables

$$\begin{aligned}
p_v &= \text{Position of the center of vessel } v. \\
w^t_{ijn} &= \begin{cases} 1 & \text{if the group of containers of type } t \text{ from source } i \text{ to destination } j \text{ is} \\ & \text{allocated to stack } n, \\ 0 & \text{iotherwise.} \end{cases}
\end{aligned}$$

Auxiliary variables

$$\begin{aligned}
e_{ij} &= \begin{cases} 1 & \text{if vessel } i \text{ is positioned left of vessel } j, \\ 0 & \text{if vessel } i \text{ is positioned right of vessel } j. \end{cases} \\
z_{vn} &= \text{Manhattan distance between the center of vessel } v \text{ and stack } n. \\
s^t_{vn}(k) &= \#\text{ containers of type } t \text{ with destination } v \text{ in stack } n \text{ during time slot } [k, k+1\rangle. \\
a^t_{ijn} &= \begin{cases} 1 & \text{the minimal distance between source } i \text{ and stack } n \text{ if the group of containers} \\ & \text{of type } t \text{ from source } i \text{ to destination } j \text{ is allocated to stack } n \\ 0 & \text{otherwise.} \end{cases} \\
d^t_{ijn} &= \begin{cases} 1 & \text{the minimal distance between stack } n \text{ and destination } j \text{ if the group of} \\ & \text{containers of type } t \text{ from source } i \text{ to destination } j \text{ is allocated to stack } n \\ 0 & \text{otherwise.} \end{cases}
\end{aligned}$$

The objective is to find vessel positions at the quay and container group positions in the stack to minimize the total carrier travel distance. The total carrier travel distance of a certain group of containers between their vessel and their stack is equal to the product of i) the number of containers in that particular group and ii) the distance between the corresponding vessel and the designated stack. Hence, we have to minimize the sum over all these products:

$$\min_{p_v, w^t_{ijn}} \sum_{t=1}^{T} \sum_{i=0}^{V} \sum_{j=0}^{V} \sum_{k=1}^{K} \left(I^t_{ij}(k) \cdot \sum_{n=1}^{N} a^t_{ijn} + O^t_{ij}(k) \cdot \sum_{n=1}^{N} d^t_{ijn} \right) \tag{5.4}$$

First, there is a set of constraints with respect to the vessel positions. Note that this set is equal to the set of constraints in (5.2):

$$\begin{array}{rcll}
p_v & \geq & \frac{L_v}{2} & \forall v \\
p_v & \leq & Q - \frac{L_v}{2} & \forall v \\
p_i - p_j & \geq & \frac{L_i + L_j}{2} - Q \cdot e_{ij} & \forall (i,j) \in \mathcal{S} \\
p_j - p_i & \geq & \frac{L_i + L_j}{2} - Q \cdot (1 - e_{ij}) & \forall (i,j) \in \mathcal{S} \\
z_{vn} & \geq & p_v - X_n + Y_n & \forall n, v \in \{1, ..., V\} \\
z_{vn} & \geq & -p_v + X_n + Y_n & \forall n, v \in \{1, ..., V\} \\
z_{vn} & = & B - Y_n & \forall n, v = 0
\end{array}$$

Second, we have a set of constraints with respect to the group-to-stack allocation.

$$\begin{array}{rcll}
\sum_{n=1}^{N} w_{ijn}^t & = & 1 & \forall t, i, j \\
s_{vn}^t(k+1) & = & s_{vn}^t(k) + \sum_{i=1}^{V+1} I_{iv}^t(k) \cdot w_{ivn}^t - \sum_{i=1}^{V+1} O_{iv}^t(k) \cdot w_{ivn}^t & \forall t, v, n, k \\
\sum_{t=1}^{T} \sum_{v=1}^{V+1} s_{vn}^t(k) & \leq & C_n - \sum_{t=1}^{T} \sum_{v=1}^{V+1} \sum_{i=1}^{V+1} I_{iv}^t(k) \cdot w_{ivn}^t & \forall n, k
\end{array} \tag{5.5}$$

The first constraint in the upper set enforces that each group of containers is allocated to exactly one stack. The second constraint updates the number of containers of a certain type t and destination v in stack n over time k. The third constraint ensures that the capacity of stack n is never exceeded.

Finally, we have a number of dependent constraints, which link the vessel positioning to the group stacking, and count travel distances for a container group between a vessel and a stack only if that particular container group of that particular vessel is allocated to that particular stack:

$$\begin{array}{rcll}
a_{vjn}^t & \leq & (Q+B) \cdot w_{vjn}^t & \forall t, v, j, n \\
a_{vjn}^t & \geq & 0 & \forall t, v, j, n \\
a_{vjn}^t & \leq & z_{vn} & \forall t, v, j, n \\
a_{vjn}^t & \leq & z_{vn} - (Q+B) \cdot (1 - w_{vjn}^t) & \forall t, v, j, n \\
d_{ivn}^t & \leq & (Q+B) \cdot w_{ivn}^t & \forall t, i, v, n \\
d_{ivn}^t & \geq & 0 & \forall t, i, v, n \\
d_{ivn}^t & \geq & 0 & \forall t, i, v, n \\
d_{ivn}^t & \leq & z_{vn} - (Q+B) \cdot (1 - w_{ivn}^t) & \forall t, i, v, n
\end{array} \tag{5.6}$$

5.3 Case Study

The terminal operator PSA HNN provided us with a representative data set consisting of a cyclic timetable, vessels' load compositions and yard lay out. For this data set, the total carrier travel distance can be computed for the weekly cycle. In this section, we aim for reductions in this total carrier travel distance by applying the proposed alternating optimization procedure for i) quasi-randomly generated initial conditions and ii) the initial condition generated by the MILP in Section 5.2.4.

5.3.1 Random Initial Conditions

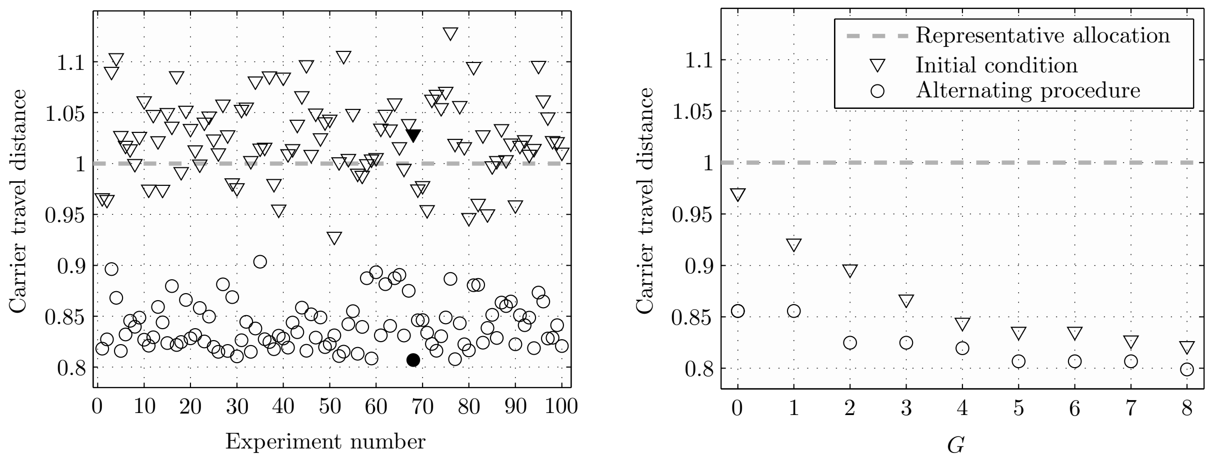
First, the performance dependency of the alternating optimization on the chosen initial condition $i_{ijn}^t(k)^\star$ and $o_{vn}^t(k)^\star$ is investigated. These initial conditions are constructed in a quasi-random manner: we use the, in Section 5.2.4 introduced, fixed-sized container groups, allocate the groups of special container types ($t \in \{2,3,4\}$) to their fixed designated stacking areas, and randomly allocate each group of full containers $t = 1$ to exactly one stack, i.e. we fix the values for w^t_{ijn}, $t \in \{2,3,4\}$ according to the provided data set, and randomly generate values for w^1_{ijn}, such that $\sum_{n=1}^{N} w^1_{ijn} = 1$, $\forall i, j$. Note that in this generation, the stack capacities are neglected and due to the random allocation, a stack might be filled above its capacity. From the generated values for w^t_{ijn} and the parameters $I^t_{ij}(k)$ and $O^t_v(k)$, values for $i^t_{ijn}(k)^\star$ and $o^t_{vn}(k)^\star$ follow directly and are fed into the the alternating procedure.

Results for one hundred of these experiments are depicted in Figure 5.3a, where a triangle represents the carrier travel distance for quasi-randomly chosen stacks and variable berth positions (objective from the first step of the alternating procedure), and the circle straight below represents the carrier travel distance after convergence of the alternating procedure for that particular initial condition. The carrier travel distances are scaled to the carrier travel distance corresponding to the provided data set (grey dotted line). From Figure 5.3a we learn the following:

- In each experiment, the alternating procedure yields significant reductions in the carrier travel distance starting from the initial condition.
- A good randomly selected initial conditions does not necessarily provide a good end solution. One of the possible reasons might be that, while allocating the groups the stacks' capacities are not taken into consideration. Hence, several groups of containers might be allocated to the same stack, inducing a small travel distance, however exceeding the actual capacity. Since the alternating optimization does take the stack capacities into account, the converged solution might end up not being that good.
- The major part (about 80%) of the quasi-randomly generated initial conditions leads to an allocation with a larger carrier travel distance than in the representative allocation.
- For each of the hundred performed experiments, the alternating procedure yields a solution that is at least about 10% better than the representative allocation.
- The best solution found (black marked circle and corresponding triangle) outperforms the current allocation by almost 20%.

5.3.2 Sophisticated Initial Conditions

In this section, we execute the alternating optimization for the initial condition generated by the MILP in Section 5.2.4. Since the MILP consists of a relatively large number of binary variables w^t_{ijn}, it takes forever to solve it to optimality. Hence, the following is proposed: the carrier travel distance for the representative allocation is computed by fixing all variables p_v, and w^t_{ijn} accordingly and evaluating the corresponding objective function (5.4). Next, the groups of full containers (type $t = 1$) are ordered according to their contributions to the total carrier travel distance. From this ordered set, we take the first G groups with the largest distance contributions, declare the corresponding values



(a) Quasi-randomly generated initial condition.

(b) Sophisticated initial condition.

Figure 5.3: Results of the alternating procedure. In each experiment (both triangles and circles), the berth positions of all vessels are variable.

for w^1_{ijn}, $\forall i,j,n$ and p_v, $\forall v$ to be variables again, and run the MILP of Section 5.2.4. Subsequently, the generated solution is fed into the alternating optimization procedure of Section 5.2.3 as an initial condition. Note that in the lower box of the alternating optimization procedure $i^1_{ijn}(k)$ is a variable $\forall i,j$ and $o^t_{vn}(k)$ is a variable $\forall v$ (and not only for the first G groups with the largest contributions). Figure 5.3b shows the results of the found initial condition and corresponding converged solution as a function of G. From the experiments shown in this Figure 5.3b we conclude the following:

- In each experiment, the alternating procedure yields significant reductions in the carrier travel distance starting from the initial condition. The reduction however decreases as G increases. A reason for this might be that for relatively large values of G, the initial guess is that good that less improvements can be obtained by the alternating procedure.

- Each generated initial condition already outperforms the allocation in the data set provided by PSA HNN.

- For $G = 0$ (zero groups can be modified) the initial condition already outperforms the allocation in the data set provided by PSA HNN. Although the container stack allocation is fixed and cannot be changed for $G = 0$ while generating this initial condition, the berth positions of the vessels are variable. Apparently, a modification of only the vessels' berth positions already yields a reduction of about 3% in carrier travel distance.

- As G increases, the found objective value for the initial conditions (triangles) decrease. This makes sense since if more container groups are variable, no larger travel distance will result from the optimization.

- A better initial condition (triangle) never yields a worse converged solution (circle). This disagrees with the observation made for Figure 5.3a, where a better initial condition not necessarily led to a better converged solution. Apparently, the reason we gave for this observation in the first place is a crucial one. Namely, the method to construct a proper initial guess does take the stacks' capacities into consideration right away.

- The modification of only the eight largest contributions (and possible all vessels' berth positions) already leads to a better solution than the best found solution for a hundred random initial conditions. The corresponding reduction in travel distance with respect to the travel distance in the representative allocation is more than 20%.

From these results we conclude that it pays off to generate a proper initial condition rather than executing an extensive number of experiments for random initial conditions.

5.4 Conclusions and Recommendations

In this chapter we considered a special form of the well-known berth allocation problem, which allocates space and time to vessels. The special form results from specific cuts chosen in this dissertation for the overall decision problems in a multi-terminal container operation. Namely, due to the generation of terminals' timetables at a higher decision level, a berthing interval is already allocated and only the space allocation problem is left at this level. This berth position problem is addressed jointly with the allocation of containers to stacks in the yard to minimize the total travel distance of carrier operating between vessels and yard.

First, an appropriate MIQP is formulated, which chooses berth positions for vessels and amounts of containers for stacks to minimize the carrier travel distance. Since the objective is non-convex and consequently real-life instances run forever, the problem is separated into an MILP, which represents the berth position allocation, and an LP, which presents the container amounts to stack allocation. These problems are coupled in the objective function and solved in an alternating fashion. The method converges to a local optimum very fast, however appears to be very sensitive to the initial condition.

Hence an MILP is proposed to find a proper initial condition by allocating berth positions to vessels and fixed-sized groups of containers to stacks. The solution found by this MILP is passed to the alternating method as an initial condition. The alternating optimization now finds a solution, which outperforms all solutions resulting from an extensive number of randomly generated initial conditions. Applying the alternating optimization on a representative data set provided by PSA HNN suggests that a reduction of more than 20% in the carrier travel distance can be obtained.

These results suggest that the same amount of work can be done with less straddle carriers or that the inter-arrival time of straddle carriers at the quay cranes can be reduced while using the same number of straddle carriers. The latter might reduce the idle time of the quay cranes and with that increase their utilization.

Chapters 3 through 5 generate a reference timetable, reference berth positions and a reference yard lay out for the considered cycle. Since in daily operations disturbances are present on e.g. vessel arrivals, call sizes and crane productivity, the daily and even hourly reference allocation can often not be met. Hence, an online optimization method is required to reschedule the disturbed system. A rolling horizon approach is developed in Chapter 6 to serve the shipping lines as good as possible for minimal operational costs.

With respect to the research in this chapter, the following recommendations are given:

- The study in this chapter considers the single objective of minimizing the total carrier travel distance. As a result, blocks closest to the berth position of a loop are typically stacked up to three containers high (in case of a straddle carrier operation) with containers from/for this loop, while other blocks remain relatively empty. In this way the total distance between vessel and containers indeed is minimized. A drawback of stacking containers on top of each other however is that it may require additional handling to retrieve a certain container. Namely, a container, which for instance is stacked at the bottom of a pile cannot be picked up directly, but first the top containers have to be shifted. Each individual container that has to be shifted to retrieve another one is called a shifter. The probability of a shifter increases as the stacking height increases.

  Since each shifter requires time and money, an interesting recommendation is to minimize the number of shifters as a second (conflicting) objective. In an ongoing case study, a relation between the stacking height and the expected number of shifters is derived and introduced in the model. Given the average time required for a shifter and the average speed of a straddle carrier, the costs of a shifter are translated into a "lost" straddle carrier distance. In this way, the two objectives can be traded off fairly.

- As mentioned in Chapter 1, one of the strategic decisions a terminal operator has to make concerns the type of equipment used for the yard operations. Generally, for this decision a terminal operator can choose from two alternatives: i) straddle carriers, or ii) rubber-tyred gantry cranes (RTG's) and trucks. In the latter case, the gantry cranes can move above the container blocks and feed/are fed by the trucks that deliver containers to/from the quay cranes. The advantage of a straddle carrier is that it can pick up and put down a container by itself. The main disadvantage is that it can only stack containers up to three high and hence the yard storage capacity is restricted.

  The other alternative requires a synchronization between a crane (quay crane or gantry crane) and a truck for loading and unloading, which may cause a crane to wait. Moreover, the container handling rate per block is very limited since maximally two gantry cranes can operate one block at a time. The main advantage of this alternative however is that containers can be stacked up to six high and therefore the yard storage capacity is double the storage capacity of a straddle carrier operation in the same terminal.

  It is interesting to compare i) the tactical performance, i.e. the straddle carrier distance and number of shifters and ii) the operational performance of both alternatives. The former of these two can be completed relatively easily by increasing doubling the storage capacities in the blocks and limiting the container handling rate per block in the approach in this chapter. The latter issue however requires a new optimization approach and/or simulation model and is therefore added to the more general recommendations in Chapter 7.

Chapter 6

Online Disruption Management in a Container Terminal: a Rolling Horizon Approach

6.1 Introduction

The solution approaches for the strategic and tactical decision making in a multi-terminal container port, addressed in Chapters 3 through 5, result in i) a timetable, ii) berth positions and iii) yard layout per terminal. This cyclic schedule is considered to be a reference for the operational planning and could be repeated over and over again if all parameters were deterministic. In practice however, a container operation is a highly stochastic system: vessels are sometimes early and often late (e.g. due to tailwind and storm, respectively), call sizes of one shipping line may heavily change from one cycle to another, and quay cranes may break down for a certain period of time. Hence, a terminal operator has to continuously adapt the reference schedule taking forecasted information on stochastic parameters into consideration and facing sudden breakdown of resources.

In this chapter, we propose a rolling horizon approach to support a terminal operator in his operational decision making. This online approach constructs operational decisions in three sequential steps taking forecasts along a limited future horizon into consideration. The former two sequential steps, i) berth time allocation and ii) berth position allocation, are similar to the ones chosen for the strategic and tactical decision making (see also Figure 1.5 and Chapters 3 through 5). The remaining step is the actual quay crane scheduling for which we propose a heuristics in this chapter.

Solving the overall operational problem in three sequential steps, rather than at once, enables to construct all considered decisions within a couple of minutes. Since in practice, these decisions are reconsidered each hour, the proposed rolling horizon approach is very suitable to be applied online. Although the found decisions are suboptimal and might theoretically even turn out to be infeasible, an extensive number of experiments suggests that found solutions are always feasible. Case studies are performed to evaluate the effect of applying different policies. For instance, taking forecasts on actual load compositions into account while finding vessels' berth positions significantly reduces the actual carrier travel distance.

6.1.1 Related Work

Many studies address the well-known berth allocation problem (BAP), which involves the joint problem of allocating time and space to vessels. As mentioned in Section 3.1.1, a main distinction is made between the static BAP and the dynamic BAP. In the static case [21], all vessels are already in the port before the actual berth allocation is determined. In the dynamic case [23], vessels arrive while work is in progress. Additional release constraints are then required to ensure that vessels only berth after their arrival. These studies however consider a deterministic problem (no disturbances on arrivals, load compositions, crane capacity or whatsoever) and do not incorporate the crane scheduling problem, which can influence the process times of vessels and with that the performance of the overall system. The authors in [41] do consider the joint problem of berth allocation and crane scheduling, however still neglect disturbances. A berth allocation (and crane scheduling) found by above mentioned studies might be optimal with respect to a certain objective, however the way of how to act in case of disturbances is not addressed.

To the best of our knowledge, only the study in [37] addresses the berth allocation problem in a stochastic environment. The authors consider a set of vessels that call at a terminal on a regular basis. Given arrival distributions and load compositions, a cyclic berth template is constructed using a simulated annealing algorithm that searches through the space of all possible sequence pairs. The objective is to minimize expected delays and connectivity costs. The connectivity costs represent the total distance that should be covered to transport containers between two connecting vessels. The performance of the constructed berth template is evaluated in a rolling horizon simulation, in which the arrivals are stochastic parameters. The objective in each iteration step of the rolling approach is to minimize delays and deviations from berth positions determined in the berth template. The study neglects the crane scheduling problem and assumes the process times of vessels to be fixed. The authors recommend to incorporate the crane allocation in the model, since it influences the process time of a vessel and probably has an impact on the system's performance.

In this chapter, we consider the berth allocation and crane scheduling problem in a stochastic environment. The robust cyclic timetable from Chapter 4 and the tactical berth position and yard lay out from Chapter 5 together are considered as the reference planning. A rolling horizon approach is proposed to online construct intelligent operational decisions taking the reference planning and disturbances on the system into account. As time evolves, updated forecasts on vessels' arrivals and load compositions along a limited future horizon become available. In each iteration step of the rolling horizon approach, the updated forecasts are taken into account while constructing the operational decisions along the limited time horizon. Only the operational decisions of the current time instance are actually executed.

The operational decisions are generated in three sequential steps: i) berth time allocation, ii) berth position allocation and iii) quay crane scheduling. The first two steps together determine the berth allocation, while the latter constructs a quay crane schedule. The considered quay crane scheduling problem is quite similar to the one considered in [41], however we develop an alternative heuristic to solve it. The authors in [41] apply a dynamic programming technique, which considers a new stage if a new event occurs being either the arrival of a vessel or a change in the number of cranes (both predetermined in the berth allocation problem). Each stage can be represented by all possible sequences of the left most crane at each vessel berthing at that stage. An optimal path through the stages is found by minimizing the total number of crane setups.

In this chapter, an appropriate MILP is constructed for the crane scheduling problem, having information on the number of cranes at each vessel over time. Since the MILP turns out to be complex from a computational point of view, it is cut into subproblems, which are solved sequentially. The end state of one subproblem is used as an initial condition for the next one. Experiments learn that the problem can then be solved much faster, while still quite accurate solutions are found.

6.1.2 Contributions

The strategic and tactical decisions generated in chapters 3 through 5 depict the reference cyclic schedule. In this chapter, the operational decision making is addressed, which takes care of adapting the schedule under disturbances. We take into account disturbances on

- arrival times,
- number and types of containers,
- quay crane productivity.

An online rolling horizon approach is proposed, which takes updated forecasts on these stochastic parameters along a limited time horizon into consideration to determine decisions for berthing and crane scheduling at the current time instance. Each iteration step of the rolling horizon approach consists of three sequential optimization steps:

1. Berth time allocation:
   All vessels that are forecasted to be (partly) within the time horizon are taken into account. Considering vessel lengths and call sizes, and considering limited quay capacity and crane capacity in the terminal, the vessels are tried to be scheduled within their reference process time. Dependent on an (expected) arrival within or out of the defined arrival window (see Chapter 4), costs are assigned for not meeting this condition.

2. Berth position allocation:
   Given the time allocation from the vessels in the previous step, the current yard layout, the (forecasted) load compositions (types and number of containers), and the (fixed) positions of vessels that are actually berthing, berth positions for all other vessels within the horizon are chosen such that non-overlapping is guaranteed and the actual carrier travel distance is minimized.

3. Quay crane scheduling:
   Given the time allocation and the berth positions, cranes are allocated to vessels, such that the average vessel delay and the additional amount of resources are minimized.

Note that the first two steps are similar to steps chosen for the strategic and tactical decision making in previous chapters. Namely, the berth time allocation and berth position allocation are also solved sequentially in chapters 3 and 4, and Chapter 5. The third step involving the quay crane scheduling typically is an operational decision problem and therefore has not yet been considered at the strategic or tactical level. For each of the three steps, an appropriate MILP is constructed. The total time for solving all three

MILP's turns out to be less than ten minutes. This enables to embed them in an online rolling horizon simulation, which is used to evaluate the system's performance.

Simulation experiments suggest that the rolling horizon approach is very suitable as an online decision support tool for real-life operational planning. Furthermore, the simulation tool enables to investigate the influence of different policies on the systems's performance:

- The rolling horizon approach takes updated information on load compositions and the actual yard lay out into account to find berth positions that minimize the actual carrier travel distance. Since the actual load composition of a call often deviates from the reference load composition, the optimal operational berth position often deviates from the reference berth position. We depict a Pareto front of the deviations from preferred berth positions versus the actual straddle carrier distance.

- A logistics provider always has to trade-off service level against resource investments. We depict a Pareto front of the vessels' delays versus the number of resources required additionally to the standard number of resources.

6.2 Approach

In this section, first the decision making faced at the operational level is formally phrased and the chosen steps to solve the problem are proposed. Next, the three sequential steps and corresponding MILP's are discussed in sections 6.2.2 through 6.2.4.

6.2.1 Problem description

We consider one terminal where a number L of shipping lines have one of their vessels calling exactly once a week according to a cyclic schedule as determined in chapters 3 and 4. Given i) this schedule and ii) the expected call sizes and compositions, suitable berth positions and container stack position(s) for each vessel line have been determined in Chapter 5. Together, this depicts a tactical timetable for the weekly cycle of L lines, where time is discrete and the considered cycle consists of P time slots.

According to this timetable, line l, $l \in \{1, ..., L\}$ has a scheduled arrival time A_l. Additionally, the expected call sizes and compositions of the lines are given by C_{ij}^t, $t \in \{1, ..., T\}$, $i, j \in \{0, ..., L\}$, the number of containers of type t from source i to destination j, where source/destination 0 represents the hinterland and source/destination $1, ..., L$ represents line $1, ..., L$, respectively.

In this chapter, we address the decision making problems during the operational execution of N cycles for a container terminal under disturbance. We define vessel $l + \alpha L$ to be the vessel of line l that is scheduled to arrive in cycle α. Accordingly, the scheduled arrival time $\bar{A}_{l+\alpha L}$ of vessel $l + \alpha L$ is given by

$$\bar{A}_{l+\alpha L} = A_l + \alpha P, \tag{6.1}$$

Moreover, the expected value of $\bar{C}^t_{i+\alpha L, j+\alpha L}$, the number of containers of type t with source $i + \alpha L$ and destination $j + \alpha L$, is assumed to be independent of the cycle:

$$\bar{C}^t_{i+\alpha L, j+\alpha L} = C^t_{ij}, \tag{6.2}$$

In this chapter, we construct operational decisions each time slot k, $k \in \{1, ..., K\}$ and $K >> P$, taking forecasts of arrivals and call sizes along a limited future horizon

$\{k, ..., k+H\}$, $H < P$ into account, and facing unexpected crane breakdowns. With respect to the forecasts, we assume $\hat{A}_{l+\alpha L}(k)$ to be the forecast at the current time k on the arrival of vessel $l+\alpha L$, and $\hat{C}^t_{i+\alpha L, j+\alpha L}(k)$ to be the forecast at current time k on the number of containers of type t with source $i+\alpha L$ and destination $j+\alpha L$. Furthermore, we assume that while a vessel approaches the port, the forecasts on its actual arrival time and load composition become more accurate and are exactly known a couple of hours in advance of the actual arrival. Due to different kind of disturbances, containers destined for a certain vessel might not yet be in the port during the time this vessel is berthing. In this case, the concerning containers are loaded onto a subsequent vessel of the same line. The values of $\hat{C}^t_{i+\alpha L, j+\alpha L}(k)$ for both vessels are adapted accordingly. Finally, we assume that a quay crane's mean availability is 98%, i.e. on average a quay crane is down for 2% of the time.

Given the forecasts along horizon $\{k, ..., k+H\}$ and given the current state of cranes in the terminal at time k, a berth and crane schedule is constructed along the horizon $\{k, ..., k+H\}$. Only the found decisions of the current time step k are actually executed. Then, the same procedure is applied for time step $k+1$, and in this way the horizon rolls forward. A rolling horizon procedure is proposed to construct operational decisions on berth allocation and quay crane scheduling at a certain time k. Figure 6.1 depicts the principle of the applied rolling horizon approach. At time k, the state of current executions and forecasts on future arrivals and load compositions are fed into the rolling horizon procedure. Based on these parameters, the berth time allocation, the berth position allocation and the crane scheduling are determined sequentially along horizon $\{k, ..., k+H\}$. Next, only the constructed decisions for time k are actually executed in the container operation. The resulting state at time $k+1$ together with updated forecasts is then again fed into the rolling horizon procedure, and in this way the horizon rolls forward. The definition of the parameters can be found in Table 6.1 and is clarified throughout Section 6.2. The following operational decisions are constructed sequentially at the current time slot k:

1. The berth interval of vessels forecasted to be (partly) within the interval $\{k, ...k+H\}$. For a vessel that is actually berthing at time k, the departure time is determined based on the remaining workload for that vessel. For a vessel that is forecasted not to be processed before the horizon end, the "artificial" departure time is set to $k+H$. Additionally, the corresponding workload is based on the part of the vessel forecasted to be within the horizon. Dependent on whether vessel v arrives within or out of its arrival window in the tactical timetable, costs are assigned if that vessel is not processed within $P^{\max}_v$.

2. The berth position of vessels forecasted to be (partly) within the horizon $\{k, ...k+H\}$. Dependent on the constructed time allocation, and the forecasted load compositions, non-overlapping berth positions are constructed. Conflicting objectives are to minimize i) the deviations from positions in the tactical plan and ii) the forecasted straddle carrier travel distance.

3. The crane schedule for all vessels that are currently berthing or have a forecasted arrival within the time interval $\{k, ...k+H\}$. Again, for vessels that are (forecasted to be) only partly within the horizon, the crane schedule is constructed for the part that is within the horizon.

From the description of the three sequential steps, it becomes clear that the procedure does not explicitly induce a convergence to the tactical timetable. Namely, although we take arrival window agreements into account, the deviations from the tactically scheduled departure time are not minimized. Hence, for zero disturbances, a time allocation might be constructed where the process time of vessel v is less than $P_v^{\max}$.

Parameter	Definition
L	Number of shipping lines in the cycle
N	Number of cycles considered for the operational planning
P	Number of time slots in the cycle
K	Number of time slots considered for the operational planning ($K = N \cdot P$)
H	Horizon length
M	Terminal length
S_v	Maximal number of quay cranes that can process vessel v simultaneously
$F^{\max}$	Total amount of quay cranes and shifts in the terminal
F^{stand}	Standard amount of quay cranes and shifts operating in the terminal
A_l	Scheduled arrival time of line l
$\bar{A}_v$	Scheduled arrival time of vessel v
$\hat{A}_v(k)$	Arrival time of vessel v forecasted at time k
$P_v^{\max}$	Agreed process time vessel v
$\hat{D}_v(k)$	Agreed departure time of vessel v forecasted at time k
C_{ij}^t	Scheduled amount of containers of type t from line i to line j
$\bar{C}_{ij}^t$	Scheduled amount of containers of type t from vessel i to vessel j
$\hat{C}_{ij}^t$	Amount of containers of type t from vessel i to vessel j forecasted at time k
W_v	Length of arrival window of vessel v
$\hat{Q}_v(k)$	Total amount of work on vessel v forecasted at time k
$Q_v^{\star}(k)$	Total amount of work on vessel v forecasted to be done within horizon at time k
L_v	Length of vessel v
G	Safety gap between quay cranes
R	Number of pieces the crane scheduling problems is cut into
$V^{\max}$	Maximal number of vessels in one piece of the crane scheduling problem
$\bar{p}_v$	Reference berth position of vessel v
c_v^t	Costs if vessel v departs after its agreed departure time
c_v	Costs if vessel v deviates from its reference berth position
c_v^e	Reward if vessel v departs before its agreed departure time
c_v^{ρ}	Costs if vessel v berths before its scheduled predecessors have left
c^f	Costs for an additional quay crane and shift

Table 6.1: Model parameters

For ease of notation, we address the operational decision making at the current time step k. At time k, we have to consider all vessels, that are forecasted to be partly or completely within the time horizon $\{k, ..., k+H\}$. We define the following mathematical sets to indicate differences in vessels' states:

- $\mathcal{V}$ is the set of vessels, which at time k are (forecasted to be) (partly) within the time horizon $\{k, ..., k+H\}$,

- $\mathcal{B}$, $\mathcal{B} \subseteq \mathcal{V}$ is the set of vessels, which are actually berthing at time k.

- $\mathcal{P}$, $\mathcal{P} \subseteq \mathcal{V}$ is the set of vessels, which at time k have a forecasted arrival time $\hat{A}_v(k)$ within the time horizon $\{k, ..., k+H\}$ and are forecasted to depart after the horizon end $k+H$,
- $\mathcal{E}$, $\mathcal{E} \subseteq \mathcal{V}$ is the set of vessels that is early, i.e. which have arrived or are forecasted to arrive before the left end of their scheduled arrival window (Chapter 4),
- $\mathcal{T}$, $\mathcal{T} \subseteq \mathcal{V}$ is the set of vessels that is in time, i.e. which have arrived or are forecasted to arrive within their scheduled arrival window.
- $\mathcal{L}$, $\mathcal{L} \subseteq \mathcal{V}$ is the set of vessels that is late, i.e. which have arrived or are forecasted to arrive after the right end of their scheduled arrival window.
- $\mathcal{E} \cup \mathcal{T} \cup \mathcal{L} = \mathcal{V}$, $\mathcal{E} \cap \mathcal{T} = \mathcal{E} \cap \mathcal{L} = \mathcal{T} \cap \mathcal{L} = \emptyset$.
- $\mathcal{W}_v$, $\mathcal{W}_v \subseteq \mathcal{V}$ is the set of vessels, which were direct predecessors of vessel v in the berth allocation at time $k-1$. In our definition, vessel i is a direct predecessor of vessel v if vessel i and v have overlapping quay positions and vessel v berths after vessel i without another vessel berthing in between.

In the next three subsections, subsequently the berth interval allocation, the berth position allocation and the crane scheduling problem at time k are addressed. A discrete event tool is used to simulate the actual container terminal under disturbances along time $k \in \{1, ..., K\}$. The below described steps are subsequently executed for each time step k.

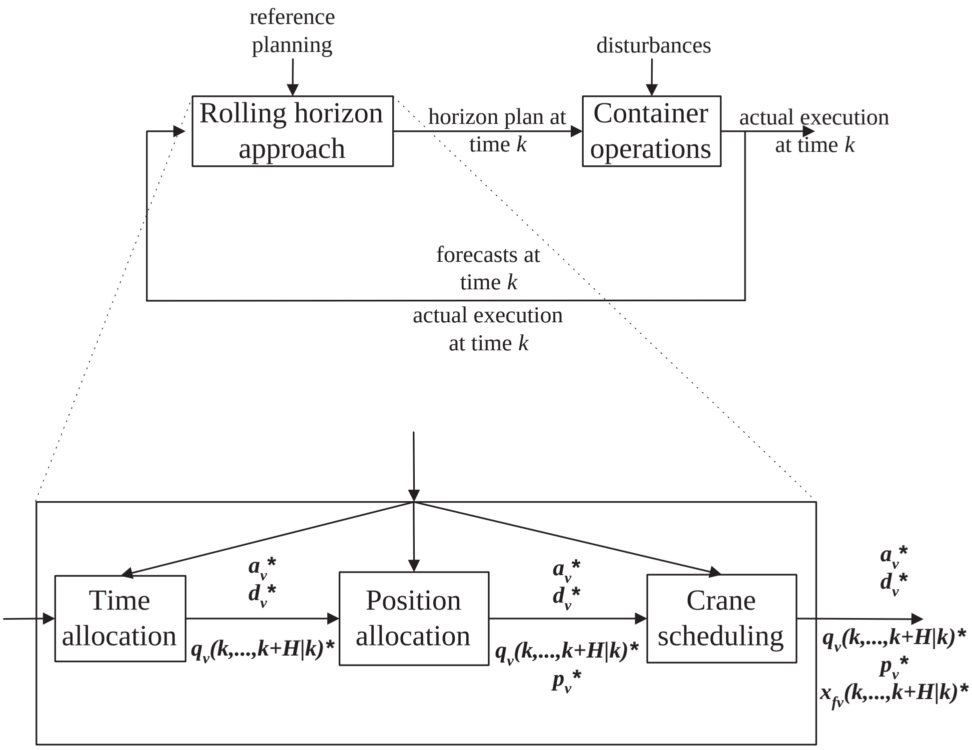


Figure 6.1: Illustration of applied approach.

6.2.2 Berth time allocation

In this subsection, the time allocation problem at time step k is addressed. First, the decision variables are defined and next an MILP is proposed and discussed.

Variables

$$\begin{aligned} a_v &= \text{First time slot that vessel } v \text{ is actually berthing} \\ d_v &= \text{Last time slot that vessel } v \text{ is actually berthing} \\ q_v(k) &= \text{Crane capacity allocated to vessel } v \text{ during time slot } [k, k+1\rangle \end{aligned}$$

MILP

Considering the previously mentioned definitions of the parameters and mathematical sets, the following MILP is proposed for the time allocation problem at time k:

$$\min_{a_v, d_v, q_v(k)} \sum_{v \in \mathcal{V}} c_v^t \Delta_v^+ + \sum_{v \in \mathcal{V}} c_v^\rho \rho_v^- + \sum_{v \in \mathcal{V}} c^f \delta^+ \tag{6.3}$$

$$a_v = k \qquad \forall v \in \mathcal{B} \tag{6.4a}$$
$$a_v \geq \hat{A}_v(k) \qquad \forall v \in \mathcal{V} \setminus \mathcal{B} \tag{6.4b}$$
$$a_v \leq k + H \qquad \forall v \in \mathcal{V} \tag{6.4c}$$
$$d_v = k + H \qquad \forall v \in \mathcal{P} \tag{6.4d}$$
$$d_v \geq k \qquad \forall v \in \mathcal{V} \tag{6.4e}$$
$$d_v \leq k + H \qquad \forall v \in \mathcal{V} \tag{6.4f}$$
$$a_v \geq d_j \qquad \forall v \in \mathcal{V}, j \in \mathcal{B} \cap \mathcal{W}_v \tag{6.4g}$$
$$\tilde{a}_v \geq d_j \qquad \forall v \in \mathcal{V}, j \in \mathcal{W}_v \setminus \mathcal{B} \tag{6.4h}$$
$$\rho_v = a_v - \tilde{a}_v \qquad \forall v \in \mathcal{V} \tag{6.4i}$$
$$\Delta_v = d_v - \hat{D}_v(k) \qquad \forall v \in \mathcal{V} \cup \mathcal{B} \tag{6.4j}$$
$$i \cdot b_v(i) \leq d_v - 1 \qquad \forall v \in \mathcal{V}, i \in \{k, ..., k + H\} \tag{6.4k}$$
$$\big(k + H - i\big) \cdot b_v(i) \leq k + H - a_v \qquad \forall v \in \mathcal{V}, i \in \{k, ..., k + H\} \tag{6.4l}$$
$$\sum_{i=k}^{k+H} b_v(i) = d_v - a_v \qquad \forall v \in \mathcal{V} \tag{6.4m}$$
$$m_v(i) = L_v \cdot b_v(i) \qquad \forall v \in \mathcal{V}, i \in \{k, ..., k + H\} \tag{6.4n}$$
$$\sum_{v \in \mathcal{V}} m_v(i) \leq M \qquad \forall i \in \{k, ..., k + H\} \tag{6.4o}$$
$$q_v(i) \geq 0 \qquad \forall v \in \mathcal{V}, i \in \{k, ..., k + H\} \tag{6.4p}$$
$$q_v(i) \leq S_v \cdot b_v(i) \qquad \forall v \in \mathcal{V}, i \in \{k, ..., k + H\} \tag{6.4q}$$
$$\sum_{i=k}^{k+H} \eta_v \bar{\lambda} \cdot q_v(i) = Q_v^*(k) \qquad \forall v \in \mathcal{V} \tag{6.4r}$$
$$\sum_{v \in \mathcal{V}} q_v(i) \leq F^{\max}(k) \qquad \forall i \in \{k, ..., k + H\} \tag{6.4s}$$
$$\delta \geq \sum_{v \in \mathcal{V}} q_v(i) - F^{\text{stand}} \qquad \forall i \in \{k, ..., k + H\} \tag{6.4t}$$

Constraint (6.4a) ensures that vessels that are already berthing, continue berthing, while constraint (6.4b) ensures that the vessels having a forecasted arrival within the window cannot start berthing before their forecasted arrival time. Constraint (6.4c) enforces that vessels, which are forecasted to arrive within the horizon start berthing before the end of the horizon. Vessels that are forecasted to arrive within the horizon, but have a forecasted departure time ($\hat{A}_v(k)+P_v^{\max}$) out of the horizon are treated as follows: an (artificial) departure time equal to the end of the horizon $k+H$ is assigned (constraint (6.4d)) and a vessel's workload is assigned based on the portion of the vessel that is forecasted to be within the horizon (see explanation of constraint (6.4r) below). Constraints (6.4e) and (6.4f) enforce that the departure time of vessel has to be within the time horizon.

Constraint (6.4g) ensures that a vessel only berths after its direct predecessors that are actually berthing have left. This constraint is introduced to avoid infeasible solutions in the next step (berth position allocation) due to sudden changes in arrival forecasts in this step. Additionally, a vessel is always tried to be placed behind its current predecessors that are not berthing yet. This is strived for to enable the allocation of a suitable berth position (Section 6.2.3). Therefore, the additional variables $\tilde{a}_v$ and ρ_v are introduced in constraints (6.4h) and (6.4i) and accordingly costs c_v^p are added to the objective function (6.3), where ρ_v^- represents $\max\{0, -\rho_v\}$.

As addressed in Chapter 4, vessel v has an arrival window of length W_v, where the left and right end of the window are given by $\bar{A}_v - \frac{W_v}{2}$ and $\bar{A}_v + \frac{W_v}{2}$, and W_v is typically 8 hours. Agreements between the terminal operator and the vessel line of vessel v state that if the actual arrival time $\hat{A}_v(k)$ of vessel v lies within the arrival window, vessel v has to be processed within $P_v^{\max}$ time. In the model, we assume the following:

- if vessel v has arrived or is forecasted to arrive before its arrival window ($\hat{A}_v(k) < \bar{A}_v - \frac{W_v}{2}$), than a high cost ($c_v^t = \beta_v$) is assigned to departing later than $\hat{D}_v(k) = \hat{A}_v(k) - \frac{W_v}{2} + P_v^{\max}$.

- if vessel v has arrived or is forecasted to arrive within its arrival window ($\bar{A}_v - \frac{W_v}{2} \leq \hat{A}_v(k) \geq \bar{A}_v + \frac{W_v}{2}$), than a high cost ($c_v^t = \beta_v$) is assigned to departing later than $\hat{D}_v(k) = \hat{A}_v(k) + P_v^{\max}$.

- if vessel v has arrived or is forecasted to arrive after its arrival window ($\hat{A}_v(k) > \bar{A}_v + \frac{W_v}{2}$), than a low cost ($c_v^t = \epsilon\beta_v$, $\epsilon << 1$) is assigned to departing later than $\hat{D}_v(k) = \hat{A}_v(k) + P_v^{\max}$.

To assign above costs and rewards appropriately, constraint (6.4j) is introduced and costs are added to the objective function (6.4g), where Δ_v^+ and Δ_v^- represent $\max\{0, \Delta_v\}$ and $\max\{0, -\Delta_v\}$, respectively. Since $c_v^p << c_v^t$, vessel v is never placed behind a direct predecessor if this would lead to a late departure time $\Delta_v > 0$.

Constraints (6.4k), (6.4l) and (6.4m) ensure that a vessel is only berthing between its start and end time of berthing. Constraint (6.4n) enforces that L_v quay meters are occupied if vessel v is berthing and constraint (6.4o) guarantees that the total quay capacity M is never exceeded.

Constraint (6.4p) and (6.4q) ensure that crane capacity can only be allocated to a vessel if this vessel is berthing. Constraint (6.4r) enforces that sufficient capacity is allocated to process the vessels in set $\mathcal{V} \cup \mathcal{B}$. Originally, the forecast at time k of the total amount of containers $\hat{Q}_v(k)$ to be unloaded from and loaded onto vessel v is given by

$$\hat{Q}_v(k) = \sum_{t=1}^{T} \sum_{i=1}^{N \cdot L} \hat{C}_{iv}^t(k) + \sum_{t=1}^{T} \sum_{j=1}^{N \cdot L} \hat{C}_{vj}^t(k) \tag{6.5}$$

We distinguish the following cases for the amount of work $Q_v^*(k)$ planned at time k to be performed on vessel v within horizon $\{k, ..., k+H\}$:

- For a vessel v in the set $\mathcal{B}$, the work $Q_v^*(k)$ is equal to the total amount of work on vessel v $\hat{Q}_v(k)$ minus the work performed on vessel v before time instance k. The work performed on vessel v before time instance k is obtained from the operations executed in previous time slots.

- For a vessel v in the set $\mathcal{V}$ that has a forecasted departure time D_v within the time horizon, the work $Q_v^*(k)$ is equal to the total amount of work $\hat{Q}_v(k)$,

- For a vessel v in the set $\mathcal{P}$ that at time k has a forecasted departure time $D_v(k)$ outside the time horizon, $Q_v^*(k)$ is equal to total amount of work minus the portion of the work that is expected to be outside of the window, i.e. $\hat{Q}_v - \frac{D_v(k)-(k+H)}{D_v(k)-\hat{A}_v(k)} Q_v(k)$.

The total crane capacity allocated at time i, $i \in \{k, ..., k+H\}$ can never exceed the maximal crane capacity $F^{\max}(k)$ at time k in the terminal (constraint (6.4s)). If a crane cannot be used due to a breakdown or maintenance, $F^{\max}(k)$ is updated accordingly. In a terminal operation, a fixed number of full time employees is working to drive the equipment. Sometimes, the workload exceeds the amount of work that can be done by this standard number of people and equipment, which we define as F^{stand}. In that case, additional shifts of people have to be hired, which induces additional costs. Therefore, we aim to minimize the amount of equipment and shifts δ^+ additional to the standard amount of equipment and shifts F^{stand}. This is achieved by including constraint (6.4t) and assigning costs c^f in the objective. The MILP can be solved within a couple of minutes for real-life instances and is hence very suitable to be applied in this online procedure.

The optimal values $a_v^\star$, $d_v^\star$, $q_v(k, ..., k+H|k)^\star$ and the positions $p_v^\star$, $\forall v \in \mathcal{B}$ ($p_v^\star$, $\forall v \in \mathcal{B}$ has been determined in the previous iteration step of the rolling horizon approach approach) are fed into the berth position allocation problem for time k. The proposed MILP is discussed in the next subsection.

6.2.3 Berth position allocation

Given the time allocation for the vessels in the considered horizon at time step k, suitable berth positions are generated in this step. Since disturbances on load compositions are taken into account, the optimal berth position of a vessel might deviate from the one determined in Chapter 5, which is defined as $\bar{p}_v$. The multi-objective then consists of minimizing i) the total straddle carrier travel distance to transport the forecasted amount of containers between vessels and stacks and ii) the weighted deviations from reference berth positions $\bar{p}_v$ as determined in Chapter 5.

Variables

$$p_v = \text{Berth position of vessel } v$$

MILP

Since the arrival and departures of the vessels are predetermined in the previous step and fixed in this step, we know the vessels that berth simultaneously. Vessel pair (i, j) is added to the set $\mathcal{S}$ if vessel i and j are in $\mathcal{V}$ and berth simultaneously (see Chapter 5 for a better explanation) according to the time allocation of the previous step. Considering the forecasted load compositions of the vessels in the set $\mathcal{V}$ and considering the set $\mathcal{S}$, berth positions p_v, $v \in \mathcal{V}$ are generated such that the carrier travel distance is minimized and non-overlapping of vessels is guaranteed. The following MILP is proposed:

$$\min_{p_v} \sum_{t=1}^{T} \sum_{i=0}^{V} \sum_{j=0}^{V} \hat{C}_{ij}^t(k) \cdot z_{ij}^t + \sum_{v=1}^{V} c_v |p_v - \bar{p}_v| \tag{6.6}$$

$$\frac{L_v}{2} \leq p_v \leq M - \frac{L_v}{2} \qquad \forall v \in \mathcal{V} \tag{6.7a}$$

$$p_v = p_v^* \qquad \forall v \in \mathcal{B} \tag{6.7b}$$

$$p_i - p_j \geq \frac{L_i + L_j}{2} - Me_{ij} \qquad \forall (i,j) \in \mathcal{S} \tag{6.7c}$$

$$p_j - p_i \geq \frac{L_i + L_j}{2} - M(1 - e_{ij}) \qquad \forall (i,j) \in \mathcal{S} \tag{6.7d}$$

$$z_{v0}^t \geq a_{v0}^{tn} \cdot p_v + b_{v0}^{tn} \qquad \forall v \in \mathcal{V} \tag{6.7e}$$

$$z_{0v}^t \geq a_{0v}^{tn} \cdot p_v + b_{0v}^{tn} \qquad \forall v \in \mathcal{V} \tag{6.7f}$$

$$z_{ij}^t \geq p_i - p_j \qquad \forall t,\ \forall i,j \in \mathcal{V} \tag{6.7g}$$

$$z_{ij}^t \geq p_j - p_i \qquad \forall t,\ \forall i,j \in \mathcal{V} \tag{6.7h}$$

Constraint (6.7a) enforces that each vessel with a forecasted arrival within the time horizon $\{k, ...k + H\}$ is totally positioned within the terminal. Constraint (6.7b) fixes the berth positions of the vessels in set $\mathcal{B}$, i.e. the vessels that are actually berthing do not change position. Constraints (6.7c) and (6.7d) guarantee that simultaneously berthing vessels do not overlap. Additional binary variables e_{ij} are required to achieve this.

In Chapter 5, the reference yard layout has been determined and hence the stack areas for the containers of a certain vessel are known. Given the position of the containers in the stack areas, the mean distance to transport an import container of type t from vessel v to stack as a function of the berth position p_v of vessel v (constraint (6.7e)) is determined. Similar functions are constructed for export containers from stack to vessel. Next, piece-wise linear functions are constructed that approximate these distance functions. Constraints (6.7e) and (6.7f) depict the lower bound on these distances, where a_{v0}^{tn} and b_{v0}^{tn} represent the parameters of the n^{th} linear piece to describe the mean travel distance of an import container of type t from vessel v to stack. In a similar way, a_{0v}^{tn} and b_{0v}^{tn} represent the parameters of the n^{th} linear piece to describe the mean travel distance of an export container of type t from stack to vessel v.

With respect to transshipment containers, the travel distance of a carrier is assumed to be equal to the absolute difference in berth position of the two connecting vessels (constraints (6.7g) and (6.7h)). The objective consists of two conflicting parts: minimizing i) the total carrier driving distance and ii) the amount of organizational changes, expressed by the weighted deviations from the reference position $\bar{p}_v$ as determined in Chapter 5. The MILP can be solved within less than a second for real-life instances and is therefore very suitable to be applied in an online optimization procedure.

The optimal values $p_v^\star$ together with the determined values $a_v^\star$, $d_v^\star$, $q_v(k, ..., k + H|k)^\star$ are fed into the next step being the crane scheduling.

6.2.4 Crane scheduling

The two previous steps construct the berth time and the berth position allocation at time k considering a future horizon of length H. In this subsection, we construct at time k a crane schedule for the horizon $\{k, ..., k + H\}$ for the found berth allocation along this horizon. First, a suitable MILP is proposed, which turns out to be rather complex from a computational point of view. Hence, a heuristics is developed, which solves the problem within satisfactory time, while finding a quite accurate solution.

Decision variables

In the first step, the arrival and departure $a_v^\star$ and $d_v^\star$, and a time-variant crane capacity $q_v(k, ..., k+H|k)^\star$ have been determined. Due to the allocation of crane capacity rather than actual cranes, the process time at the first step might turn out to be to short at this third step. Moreover, due to the breakdown of a crane, a feasible non-crossing schedule might not be found although theoretically sufficient capacity is available. Hence, in this step we introduce decision variables that describe the start and end of processing required for a non-crossing crane schedule:

$$\begin{aligned} \mathfrak{a}_v &= \text{First time slot that vessel } v \text{ is actually berthing} \\ \mathfrak{d}_v &= \text{Last time slot that vessel } v \text{ is actually berthing} \end{aligned}$$

Additional to these decision variables, the auxiliary binary variable $\mathfrak{b}_v$ is required:

$$\mathfrak{b}_v(k) = \begin{cases} 1 & \text{if vessel } v \text{ can be processed during time slot } [k, k+1\rangle, \\ 0 & \text{otherwise.} \end{cases} \tag{6.8}$$

A third decision variable is the activity of crane f, $f \in \{1, ..., F\}$ on vessel v, $v \in \mathcal{V} \cup \mathcal{B}$ at time i, $i \in \{k, ..., k+H\}$:

$$x_{fv}(k) = \begin{cases} 1 & \text{if crane } f \text{ processes vessel } v \text{ during time slot } [k, k+1\rangle, \\ 0 & \text{otherwise.} \end{cases}$$

MILP

The following MILP is proposed:

$$\min_{x_{fv}(k)} \sum_{v \in \mathcal{C}} c_v^t \cdot \Delta_v^+ + c^f \cdot \hat{\delta}^+ \tag{6.9}$$

$$\mathfrak{a}_v \geq a_v^\star \qquad \forall v \in \mathcal{V} \tag{6.10a}$$

$$\hat{\Delta}_v = \mathfrak{d}_v - d_v^\star \qquad \forall v \in \mathcal{V} \tag{6.10b}$$

$$k \cdot \mathfrak{b}_v(i) \leq \mathfrak{d}_v - 1 \qquad \forall v \in \mathcal{V}, \forall i \tag{6.10c}$$

$$(k + H - i) \cdot \mathfrak{b}_v(i) \leq k + H - \mathfrak{a}_v \qquad \forall v \in \mathcal{V}, \forall i \tag{6.10d}$$

$$\sum_{i=k}^{k+H_c} \mathfrak{b}_v(i) = \mathfrak{d}_v - \mathfrak{a}_v \qquad \forall v \in \mathcal{V} \tag{6.10e}$$

$$\sum_{f=1}^{F} x_{fv}(i) \leq S_v \cdot \mathfrak{b}_v(i) \qquad \forall v \in \mathcal{V}, \forall i \tag{6.10f}$$

$$\sum_{v \in \mathcal{V}} x_{fv}(i) \leq 1 \qquad \forall f, i \tag{6.10g}$$

$$\sum_{f=1}^{F} \sum_{i=k}^{k+H} \lambda_{fv} \cdot x_{fv}(i) \geq \sum_{i=k}^{k+H} q_v(i|k)^\star \qquad \forall v \in \mathcal{V} \tag{6.10h}$$

$$l_f(i) \geq x_{fv}(i) \cdot (p_v^\star - \frac{L_v}{2}) \qquad \forall f, i, v \in \mathcal{V} \tag{6.10i}$$

$$l_f(i) \leq x_{fv}(i) \cdot (p_v^\star + \frac{L_v}{2}) + M \cdot (1 - x_{fv}(i)) \qquad \forall f, i, v \in \mathcal{V} \tag{6.10j}$$

$$l_f(i) \leq l_f(i+1) - G \qquad \forall f, i \tag{6.10k}$$

$$\sum_{f=1}^{F} \sum_{v \in \mathcal{V}} x_{fv}(i) \leq F^{\max} \qquad \forall i \tag{6.10l}$$

$$\hat{\delta} \geq \sum_{f=1}^{F} \sum_{v \in \mathcal{V}} x_{fv}(i) - F^{\text{stand}} \qquad \forall i \tag{6.10m}$$

The start time of processing $\mathfrak{a}_v$ of vessel v has to be at least the arrival time $a_v^\star$ allocated in the first step (constraint (6.10a)). Furthermore, a costs is assigned if the end time of processing $\mathfrak{d}_v$ is larger than the generated departure time $d_v^\star$ (constraint (6.10b)). Only between the start and end time of processing, a number of cranes can operate the vessel with a minimum of zero and a maximum of S_v (constraints (6.10c) through (6.10f)). Constraint (6.10g) enforces that at time i, a crane can be operating at one vessel at most. Within the considered horizon $\{k, ..., k + H\}$, at least a total amount of work $\sum_{i=k}^{k+H} q_v(i|k)^\star$ forecasted at time k has to be performed on vessel v (constraint (6.10h)).

Constraints (6.10i) and (6.10j) ensure that at time i, crane f is positioned within the vessel bounds of vessel v if at time i crane f is allocated to vessel v. Furthermore, the non-crossing of quay cranes is modeled by constraint (6.10k), which enforces that crane f is always left of crane $f + 1$, $f \in \{1, ..., F - 1\}$, where G is the safety gap between two neighboring cranes. The total amount of cranes allocated at time i cannot exceed the maximal number $F^{\max}(k)$ of cranes that are not broken or in maintenance at time k (constraint (6.10k)). Finally, costs are assigned for each crane and shift, which is required additional to the standard number of cranes and shifts.

If the found start and end time $\mathfrak{a}_v^\star$ and $\mathfrak{d}_v^\star$ of vessel v deviate from the generated berth interval $a_v^\star$ and $d_v^\star$ of vessel v, the values of $\mathfrak{a}_v^\star$ and $\mathfrak{d}_v^\star$ are set as respectively upper and lower bounds in the first step of the next rolling horizon approach iteration. To be more precise, the following constraints are added for vessel v to the time allocation problem at time $k+1$:

$$a_v \leq \mathfrak{a}_v^\star \tag{6.11a}$$
$$d_v \geq \mathfrak{d}_v^\star \tag{6.11b}$$

The MILP cannot always be solved within a time that is required for an online application. Hence, we propose a heuristics, which can be solved within satisfactory time and still finds quite accurate solutions.

Heuristics

In each of the Sections 6.2.2, 6.2.3 and 6.2.4, a problem is solved considering a limited time horizon. A vessel that is partly within the horizon is cut off and the amount of work to be done for the part within the window is assigned proportionally. The principle of the heuristics for the crane scheduling problem at time k is very similar:

1. The values $p_v^\star$, $a_v^\star$, $d_v^\star$ and $q_v(k,...,k+H|k)^\star$ from the generated berth allocation are considered along the horizon $\{k,...,k+H\}$.
2. The considered berth allocation is cut into R time pieces, where $\mathcal{V}^r$ is the set of (parts of) vessels in piece r, $r \in \{1,...,R\}$ such that $|\mathcal{V}^r| \leq V^{\max}$ $\forall r \in \{1,...,R\}$.
3. The proposed MILP is applied to sequentially construct crane schedules for pieces $r \in \{1,...,R\}$. The solution values at the end time interval of piece r is used as an initial condition for the crane scheduling of piece $r+1$.

The time required to solve the heuristics significantly decreases as $V^{\max}$ decreases, at the expense of a less accurate solution. An extensive number of experiments suggest that for $V^{\max}=6$ a solution is found within 5% of the global solution while the runtime is satisfactory. Since the heuristics can be solved within a couple of minutes, it can be applied in an online setting.

Experiments on real-life cases suggest that one rolling horizon approach iteration step (involving the time allocation, position allocation and crane scheduling) requires less than ten minutes. Since in practice, the operational decisions on berth allocation and crane scheduling are updated each hour, this method is very suitable to be implemented into a practical online container terminal application. It can serve as an online decision support tool, where it is up to the terminal operator to which extent he adopts the generated decisions.

6.3 Results

The above described rolling horizon approach method is applied in a simulation environment (see also Figure 6.1). A discrete event tool is used to mimic the real-life container operation system under disturbances, by generating stochastic arrivals, load compositions and crane breakdowns. These parameters together with the reference planning are input for the rolling horizon approach at time k, $k \in \{1,...,K\}$. The rolling horizon approach executes the three sequential steps (sections 6.2.2, 6.2.3 and 6.2.4) and delivers the values into the container operation simulation. The constructed decisions for time k are actually simulated. Next, the state of the system under disturbance after these executed decisions is again fed into the rolling horizon approach and subsequently the same procedure is performed for time $k+1$.

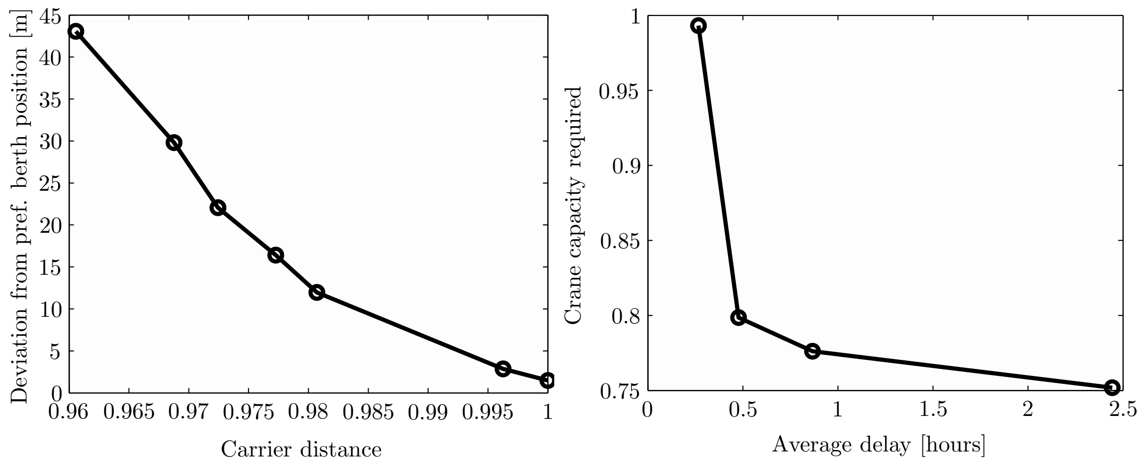
The constructed simulation tool is applied to investigate the system's performance for different policies. In the presented experiments we consider discrete time k, $k \in \{1, ..., K\}$, where each time slot has a width of one hour. Furthermore $N = 10$ and $P = 168$, and thus $K = 1680$, i.e. we consider the operational decision making along ten weeks (ten cycles). Furthermore, at each time k a future horizon of three days length $H = 72$ is taken into consideration. Forecasts on arrivals are updated three times and become more accurate as a vessel approaches the port according to the following procedure: If the right end of the rolling horizon approaches the scheduled arrival time of vessel v, the first forecast on the actual arrival of vessel v $\hat{A}_v(k)$ is drawn from a truncated normal distribution $N(\bar{A}_v, \sigma_l^2)$. For the next time steps, this first forecast is considered to be the actual arrival time of vessel v. However, if the center of rolling horizon approaches this forecast, a second sample from a truncated normal distribution $N(\hat{A}_v(k), \epsilon^1 \sigma_l^2)$, $\epsilon^1 < 1$ is drawn. This sample becomes the updated forecast of the actual arrival time of vessel v. Finally, if the left end of the rolling horizon is only eight hours away from the updated forecast, a third sample is drawn from a truncated normal distribution $N(\hat{A}_v(k), \epsilon^2 \sigma_l^2)$, $\epsilon^2 < \epsilon^1$. This latter forecast is assumed to be the actual arrival time of vessel v. The forecasts for the load composition of vessel v are constructed according to a similar procedure. Means and variances of the stochastic parameters are representative for real-life container operations, however left out for confidentiality reasons.

6.3.1 Experiment 1

The berth positions for the various loops in the tactical plan are constructed based on the expected load compositions of the loops (Chapter 5), i.e. we assume that each of the vessels in one loop import and export exactly the same number and types of containers each cycle. In practice however, the load composition of the vessels of the same loop can differ from one cycle to another. We are interested in a reduction in carrier travel distance if operational berth positioned are generated by taking forecasts on load compositions into account, rather than trying to satisfy the berth positions in the tactical plan. This can be investigated by running experiments for different values of c_v in (6.6) while the other parameter values remain the same. To obtain a statistically confident result, this is repeated nine times for each value of c_v. Although this number seems quite low, a Lilliefors test (see [31]) suggests that the output data is normally distributed. Results are shown in Figure 6.2a.

From this figure we learn that the carrier distance can be reduced by 4% by taking the forecasts on actual load compositions into account while finding vessels' berth positions. In this case, on average a vessel berths about 45 meters away from its scheduled berth position (Note that a terminal length is commonly larger than one kilometer). We think it is worth to construct berth positions based on forecasts on load compositions at the expense of relatively small deviations from the tactical berth positions.

Note that the average deviation from preferred (or tactical) berth position is never zero. A reason for this is that delayed vessels block the preferred berth positions of other subsequent vessels.



(a) Deviation from preference berth position vs. Carrier distance.

(b) Average delay vs. Additional cranes and shifts.

Figure 6.2: System performance for different policies.

6.3.2 Experiment 2

In a container terminal, commonly a standard number F^{stand} of cranes and employees are working on a regular basis. In busy periods, an additional number of people or actually shifts is hired to guarantee a desired service level (of vessels departing in time).

We are interested in the average vessel delay as a function of the additional cranes and shifts. This can be investigated by running experiments for different values of $\frac{c_v^t}{c^f}$ in (6.3) and (6.9) while the values of all other parameters (forecasts included) remain the same. Again a Liliefors test suggests that the output data (of nine experiments) is normally distributed. Results are shown in Figure 6.2b.

This figure suggests that on average 5% of additional cranes and shifts are required to reduce the average vessel delay below half an hour. We have to remark that in the simulation all vessels that have arrived before or within their arrival window were processed within their maximum process time. Hence, the average delays as depicted in the figure totally result from vessels that arrived late.

6.4 Conclusions and Recommendations

The strategic and tactical decisions constructed in Chapters 3 through 5, result in a cyclic reference schedule for the berth allocation and yard layout. If a container operation was a deterministic system, this reference schedule could be executed cycle after cycle. A container operation however is a highly stochastic system and hence the operational schedule will often deviate from the reference.

In this chapter, we addressed the operational decisions of finding berth times, berth positions and crane schedules for a container terminal under disturbances. We applied a rolling horizon approach that takes forecasts on arrivals and load compositions along a limited future horizon and sudden crane breakdowns into account while making the operational decisions at a certain time step. Each time step, updated information becomes available, and decisions are adapted accordingly. This results in a rolling horizon procedure, which can be used in an online setting.

Each step of the rolling horizon approach consists of three sequential steps, being i) berth time allocation, ii) berth position allocation and iii) crane scheduling. For each step, an MILP is proposed, which can be solved very efficiently. Solving the three steps sequentially rather than solving the joint problem leads to a small computation time, which is required to implement the approach in an online setting. Since the total computation time of the three sequential steps is less than ten minutes it is very suitable to be applied in an online container operation. Namely, in practice the considered operational decisions are only updated each hour of the day.

The rolling horizon approach approach was successfully embedded in a simulation environment, where a discrete event tool is used to mimic the real-life container operation under disturbances. Forecasts on arrivals and load compositions are generated by the discrete event tool and fed into the rolling horizon approach. Given the forecasts and the reference schedule, the rolling horizon approach constructs operational decisions along the future time horizon. Only the decisions at the current time instance are actually executed by the discrete event tool. The resulting state of the system and updates on arrivals and load compositions are passed to the rolling horizon approach again.

The simulation tool enables to investigate the performance of the system for different policies. A first experiment learned that the straddle carrier travel distance can be reduced by 4% if the forecasts on load compositions are taken into account and deviations from scheduled berth positions are allowed. In a second experiment, we depicted the average vessel delay as a function of the additionally required amount of cranes and shifts. Results suggested that on average one additional crane and shift are required to reduce the average vessel delay below one hour.

The developed rolling horizon approach is very suitable to be applied in an online container operation. Forecasts on stochastic parameters and resource breakdowns are taken into account while constructing intelligent operational decisions.

For the conducted experiments with representative stochastic behavior, a horizon length of three days appeared to be sufficient to yield a stable system. However, if due to a confluence of events a large number of vessels would arrive at the port at the same time, the question is whether could all be fit within this horizon. A first solution would be to adapt the window length in case such a situation occurs. An interesting theoretical study would be to determine the minimal horizon length required to have a rolling horizon approach that is proven to be stable for given bounds on the stochastic parameters.

Chapter 7

Conclusions and Recommendations

The last decades, the number of people and goods that have to be transported from one place to another has made an exponential growth. For the transport resources are required. A set of these resources linked together is called a logistics network. Such a logistics network is usually run by a logistics service provider, who aims to deliver the right amount of people and/or goods at the right place at the right time. To fulfill these objectives, a provider has to make decisions on different levels, classified according to their extent of impact on the network performance and according to the order of time scale of their actual execution.

We distinguish strategic, tactical and operational decision making levels on the large, medium and short term time scale, respectively. Due to the tremendous sizes of today's logistics networks, intelligent decision support tools are required to make the combined decisions so as to run the network efficiently. Since the combined decisions at the various levels or even at one specific level are often too complex to be solved at once, the overall decision making problem can be cut into several subproblems, which then are solved sequentially or alternatingly.

This dissertation addresses the overall decision making problems in two specific logistics networks, being i) a distribution network run by a logistics service provider and ii) a multi-terminal container operation in a sea port. For both these logistics, the following conceptual approach is applied: firstly, we reason about how to cut the overall decision making problem into subproblems. This is done in such a way that each subproblem can be solved within the time allowed (dependent on the decision level) and is still a practically interesting problem in its own right. Secondly, for each of the subproblems, a mathematical model is built to optimize the decisions concerned. Finally, case studies are performed to compare the performance of the currently applied decisions to the performance of the decisions found by our approach.

7.1 Conclusions

Distribution network

In Chapter 2, the decision making problem in a distribution network from the viewpoint of a logistics service provider (LSP) is considered. The tasks of an LSP in such a distribution network is to ship different product from suppliers to retailers, possibly using intermediate warehouses for i) storage to buffer the variability in supply and demand and

ii) consolidation of the various product flows so as to leverage on the economy of scales principle.

In this study, the strategic decisions of locating warehouses in between suppliers and retailers are assumed to be given, and only the combined tactical and operational decisions are addressed: the LSP faces the tactical problem of constructing an efficient and consistent network topology (i.e. where to establish so-called line hauls, which enable shipments between two facilities), that still performs satisfactorily on the operational level. An efficient topology contains a small number of line hauls so as to leverage on the economy of scales principle. A consistent topology has fixed line hauls to provide constant and with that familiar schedules and routings for personnel. We therefore aim to construct a network topology with a small number of fixed line hauls that still performs well on the operational level, where costs are assigned per truck, per stored item and for early and late delivery.

Since solving the combined tactical and operational decisions at once requires too much computational time, a procedure is proposed that alternates between the tactical and operational decision making. This procedure iteratively drops least used line hauls (starting from a fully connected network) at the higher tactical level, after constructing the operational decisions for a medium scale time period (where each day forecasts on supply and demand are available along a limited time horizon).

Experiments suggest that this procedure finds approximately similar topologies if linear transportation costs are used rather than stepwise (economy of scale) transportation costs. The linear model however can be solved much faster and enables to construct topologies for real-life network instances within minutes. An extensive number of experiments suggests that the procedure always finds a topology with a very small number of links that still performs satisfactorily at the operational level. Furthermore, the found topology turns out to be robust to changes in second and higher order moments of supply and demand distributions, however appears to be sensitive to changes in the first moments.

Multi-terminal container operations

A container terminal operator provides vessel lines with his services of unloading and loading, transporting and storing containers. Dependent on his turnover, an operator might consider the expansion of the number of terminals he is operating. Once a terminal becomes available somewhere, an operator can take place in a competitive bidding procedure to operate the concerning terminal in the future. Further strategic decisions concern the type of yard operation (either a straddle carrier operation or an operation with rubber-tyred gantry cranes and trucks), and the number of quay cranes that have to be placed at the quay to unload and load the vessels.

Each vessel line owns a fleet of vessels to realize a number of loops around the world. Commonly, the number and phasing of the vessels in one loop are such that one vessel calls at a terminal at the same time each cycle (typically once a week). On the tactical level, a terminal operator faces the problem of constructing a tactical timetable for the loops that call on his terminal. Since the actual arrivals often deviate from the tactical timetable due to bad weather conditions and breakdowns, a terminal operator and each of the vessel lines agree on two types of arrivals: within and out of a so-called arrival window. If a vessel arrives within its arrival window, the terminal operator has to process the vessel within a predetermined process time. If not, the operator is not bound to any process time. Of course he tries to operate these vessels as soon as possible, however without sacrificing the within window agreements for other vessels. Hence, a terminal operator faces the problem of constructing a tactical timetable, which is robust to all scenarios where vessels arrive anywhere within their arrival windows.

A second tactical problem is to allocate a berth position to a loop at the quay and the stack positions of its containers in the yard. Goal is to minimize the distance that has to be covered to transport containers between quay and yard. To provide operational consistency for its employees, decisions on berth positions and yard lay-out are reconsidered at the medium term time scale.

The strategic and tactical decisions result in a periodic plan, that can theoretically be repeated over and over again. However, in practice a container operation is a highly stochastic system: arrivals, call sizes and compositions often deviate from the tactical plan and resources may break down for unknown time. Hence, a terminal operator constantly has to reschedule the operations to service the vessel lines as good as possible.

In this dissertation, we consider the terminal operator PSA HNN, which is responsible for a number of terminals in the port of Antwerp, Belgium. The current policy of PSA HNN is to satisfy the preferences of the vessels lines with respect to the terminal and time of berthing. Although the strategic decisions of the number of terminals, the kind of operations (straddle carriers), and the number of quay cranes are already given, we question whether the same amount of vessel lines could be serviced with less quay cranes if small modifications to the lines' preferences were allowed. If so, the spare crane capacity could be used to service additional vessel lines. Additionally, we are interested in a reduction in the costs for inter-terminal transport induced by connecting vessels that berth at different terminals (in the same port).

In Chapter 3, we address the combined strategic and tactical decision making problem of determining a i) terminal and ii) time for berthing such that the required quay crane capacity and the costs for inter-terminal transport are minimized. A mathematical model is proposed to allocate a terminal and time for berthing to a number of loops that calls periodically on the port of Antwerp. Conflicting objectives are spreading the workload evenly over the terminals and minimizing the costs for inter-terminal transport. Although sufficient quay and crane capacities are reserved, the actual berth position of vessels and quay cranes operating them are left to be allocated in a subsequent problem. The model is applied to a representative allocation provided by PSA HNN. Results suggest that with small modifications to the representative allocation, the same number of vessel lines can be operated with only 75% of currently used capacity while at the same time a reduction of 3% in the costs for inter terminal transport can be achieved.

The decisions made in Chapter 3 result in a (periodic) timetable per terminal. In Chapter 4, this timetable is slightly modified to increase its robustness to the within arrival window agreements. In our definition, a berth plan is robust with respect to a given set of arrival windows if a feasible solution exists for each arrival scenario where all vessels arrive within their windows. The price for achieving this robustness is then the additional crane capacity reservation that is required in the worst case arrival scenario where all vessels arrive within their windows. The problem is hence to construct a window-based berth plan that minimizes the maximally required crane capacity for all scenarios where vessels arrive within their arrival windows.

An appropriate mathematical model is proposed that allocates a time for berthing and reserves crane capacities for each vessel such that the within window arrival agreements are satisfied. Objective is to minimize the maximally required crane capacity reservation. Although sufficient quay and crane capacities are reserved, the actual berth position of vessels and quay cranes operating them are left to be allocated in a subsequent problem.

As a special case, the model finds the nominal berth plan, i.e. the optimal deterministic plan if disturbances were neglected. Results suggest that the window-based plan requires slightly more crane capacity than the nominal plan if arrivals are only slightly disturbed. However, the window-based plan is much more robust for relatively large disruptions that are still within the arrival window bounds.

The allocations in chapters 3 and 4 result in a robust tactical timetable per terminal. The exact positions of the vessels within the terminal however, are still to be determined. In Chapter 5, the combined problem of berth position allocation and yard layout design is addressed. Given the arrival and departure times per loop and their call sizes and call compositions, the objective is to find berth positions at the quay and container positions in the yard such that the total carrier travel distance is minimized.

A mathematical model for the combined problem is proposed, however cannot be solved within reasonable time. Hence the problem is cut into the berth position problem and the yard design problem. Since both problems can be decoupled completely, but share the same objective function they are very suitable to be solved alternatingly. The alternating optimization converges to a local minimum very fast, however heavily depends on the initial condition. Therefore, an additional mathematical model is proposed that finds a proper initial condition. Experiments suggest that starting from this initial condition leads to a solution that outperforms the best of all solutions resulting from an extensive number of random initial conditions. Experiments suggest that this method finds an allocation that reduces the current carrier travel distance by over 20%.

The strategic and tactical decisions constructed in chapters 3,4 and 5 depict the periodic schedule and allocation per terminal. In Chapter 6, an online decision tool is proposed to react upon disrupted vessel arrivals, call sizes and crane rates. A rolling horizon approach is used that takes forecasts of these stochastic parameters along a limited time horizon into account while constructing the operational decisions for the current time instance. The operational decisions are subdivided into three sequential steps. For each of the steps a mathematical model is proposed: firstly, a time for berthing is allocated taking the within window arrival agreements, and quay and crane capacities into account. Secondly, given the time allocation, an appropriate berth position is found for each vessel such that the carrier travel distance is minimized. Finally, given time and position allocation, a feasible quay crane schedule is constructed.

Experiments for a representative data set suggest that the first and third steps can be solved within a couple of minutes, while the second step can be solved within less than a second. Since in practice these operational decisions are only reconsidered each hour, the proposed procedure is very suitable to be applied in an online setting. Simulations suggest that using this procedure, significant reductions in the carrier travel distance can be achieved compared to the distance following from the current policy. Namely, the procedure takes forecasts on actual call compositions into account while finding an appropriate berth position of a vessel, which may deviate from the tactical berth position. The current policy however tries to satisfy these tactical berth positions as good as possible.

As mentioned before, the combined decisions in the considered logistics networks are too complex to be solved within reasonable time and we choose to cut them into several subproblems, which are then solved subsequently. It turns out that with the proposed method each of the subproblems can be solved within the time allowed at the concerning decision level. Although some managerial decisions or maybe even some physical constraints might not be incorporated, the proposed methods at least enable to quantify the resulting additional costs, which can be very valuable for a logistics provider. The solution methods derived in this dissertation are therefore very suitable to provide insights in the costs of certain modifications. Moreover, the methods can be applied as decision tools, which support a provider in his decision making. Next, it is up to the provider to which extent he actually follows the decisions generated by the proposed methods. Below we give a short summary of the main contributions and refer to the corresponding section(s) where these contributions can be found.

Contributions

- Identification of the combined decision problems at strategic, tactical and operational levels in two logistics networks: a distribution network and a multi-terminal container operation. This is addressed in Section 1.2.1 for the distribution network and in Section 1.2.2 for the multi-terminal container operation.

- Division of the combined problems into subproblems that i) can be solved within the time allowed and ii) are practically interesting in their own right. For the distribution network this is discussed in Section 1.4.1, for the multi-terminal container operation in Section 1.4.2.

- Proposition and construction of approaches to solve the individual subproblems. In Section 2.2, the bi-level approach for the combined decision making at the tactical and operational levels in a distribution network is developed. Section 3.2 proposes two mathematical formulations (mixed integer linear program) for a strategic decision problem in a multi-terminal container operation. A mathematical problem (mixed integer linear program) for the construction of a robust container terminal timetable is derived in Section 4.2. An alternating optimization method is developed in Section 5.2 to find tactical berth positions and a corresponding yard lay-out. In Section 6.2, a rolling horizon approach is proposed and constructed for the online disruption management of a container terminal.

- Validation of the performance and suitability of the methods. Section 2.3 suggests that the proposed bi-level optimization method for the distribution network can be solved very efficiently and hence can be very helpful to a logistics service provider. In Section 3.3, a performance analysis of the two proposed models for the strategic decision making problem in a multi-terminal container port learns that the alternative formulation can be solved much faster than the straightforward formulation. The alternative method enable to solve real-life instances. The suitability of the rolling horizon approach for an online application is shown in Section 6.3.

- Application of the methods in case studies with representative data provided by logistics providers. A case study on the potential reductions in the number of quay cranes and inter-terminal transport is conducted in Section 3.4. The case study in Section 4.3 aims for robustness improvements on a representative terminal timetable. In Section 5.3, a case study is performed on to reduce the total carrier travel distance of a representative data set.

- Quantification of the improvements that can be achieved with respect to solutions as currently applied by logistics providers. Although we are aware of the fact that our models do not take managerial decisions or even physical constraints into account, the performed case studies give a proper quantification of the resulting additional costs. The case study in 3.4 suggests that with relatively small modifications to an existing terminal allocation, the required crane capacity can be reduced by about 25%, while at the same time the costs for inter-terminal transport can be reduced by 3%. Furthermore, results suggest that if all vessel lines would allow a change in the terminal they call on, the costs for inter-terminal transport could be reduced by about 40%. The results of the case study in Section 4.3 suggest that improvements can be made on the robustness of a timetable. For a proper quantification, arrival distributions are required. The case study in Section 5.3 suggests that the total carrier driver distance can be reduced by over 20%.

7.2 Recommendations and Ongoing Work

At the end of each chapter, recommendations are given for studies that can be performed relatively fast by slightly adapting or expanding the proposed models or methods for that particular chapter. In this section, first recommendations for improvements on the total chosen multi-step approach are given. Next, we phrase ongoing or subsequent studies that require relatively large modifications or even totally new models. Finally, we discuss the potential of the proposed models and methods for different applications.

Criticism

One drawback of the concept of solving the subproblems subsequently is that an allocation constructed in one subproblem might be infeasible in a subsequent step. Although not one of an extensive number of representative experiments revealed this problem, a recommendation is to build in feedback loops that in case of an infeasibility return to a previous subproblem and generates additional constraints to fix it. For instance, in the first optimization step of the container operations, a set of vessel loops is allocated to a certain terminal and a corresponding timetable is constructed. Although sufficient quay capacity is reserved at this first optimization step, an actual feasible berth position (third optimization step) might not exist. A possible solution may be to track the time instance(s) that the quay is overloaded and to decrease the total quay capacity on these instances in the first optimization step and run it again.

Another drawback of the concept of solving subsequent subproblems is that we are no longer guaranteed to find a global optimum. In this dissertation however little attention is paid to determine the (expected) deviation between the found solutions and a global optimum. The focus is more on the improvements that can be gained with respect to the solutions as currently applied in real-life logistics networks. A recommendation is to quantify the deviations between the solutions found be the proposed methods and a global optimum. In Chapter 2, results from the heuristics are compared to a global optimum for small instances. The same can be done for the combined decision problems in the multi-terminal container operations. For instance, the performance of the initial model that incorporates both the berth position allocation and the yard design problem (Chapter 5) can be compared to the developed alternating procedure (Chapter 5) for small instances.

Additionally, the performance of the chosen subsequent procedure can be compared to the performance of other heuristics (e.g. genetic algorithms).

Subsequent studies

In Chapter 2, it is assumed that the strategic decisions on the location of the warehouses in the distribution network are given. A study that addresses the strategic problem of finding proper locations and capacities of (new) warehouses is very interesting. As a problem definition we think of the following: given the locations of suppliers, retailers and existing warehouses and given the expected supplies and demands, find the location(s) of new warehouses such that the expected driving distance is minimized. The alternating optimization procedure as proposed in Chapter 5, would be very suitable to solve this problem. Namely, the warehouses to be placed can be seen as vessels and the existing facilities as the stacks. Variables are i) the (two-dimensional) position of the new warehouses and ii) the number of trucks along a certain line haul between two facilities. The number trucks along a line haul can be seen as the number of containers between a vessel and a block. The goal is then to minimize the sum over all products of the number of trucks along a line haul and the graphical length of this line haul.

In Chapter 3, a timetable for a multi-terminal container operation is constructed under the assumption that each calling loop has a cycle length of one week. In practice however, a limited number of loops have a deviating cycle length, by calling three times in four weeks or once in two weeks. In this case, a cyclic timetable of four weeks instead of one is required. The existing model only needs to be adapted slightly to incorporate these multiple cycle lengths. For each loop four, three or two (dependent on the cycle length) variables are introduced, which each represent subsequent calls of the same loop. The time allocations of the calls however are dependent since their inter-arrival time (cycle length) is known. Moreover, containers can now be distributed among the (four, three or two) vessels of the destined loop and hence an additional constraint is required to model this. By minimizing the number of stored containers over time, stacked containers are induced to leave with the next vessel. Although the number of variables will grow by about a factor four, the additional variables are all auxiliary. Hence, we expect the complexity of the model not to increase much.

Along the same line, the method in Chapter 5 can be expanded to manage multiple cycle lengths. In an ongoing case study for a terminal operated by PSA HNN, we have already successfully adapted the model to incorporate multiple cycle length. The case consists of a number of vessel line that calls once each week and one vessel line that calls three times in four weeks. Hence, a cyclic plan for four weeks had to be constructed. Variables were introduced for the berth positions of the four, and in the one case three, calls of one loop. Since the arrivals, departures and load compositions of the calls are given for this problem, we could predetermine which containers had to go with which call. The four week plan resulting from the alternating optimization has recently been adopted by PSA HNN.

The online operational decision making has been addressed in Chapter 6 of this dissertation. The latter of the three steps approach assigns quay cranes to vessels for a certain time. However, the exact sequence of unloading and loading the bays of a vessel is still to be determined. In an ongoing study, we aim to construct an unloading and loading sequence for quay cranes such that the number of opposite (loading versus unloading) movements is maximized. The underlying thought of this objective is to increase the number of double moves by straddle carriers. Commonly, a number of straddle carriers is assigned to a vessel. If the vessel is totally unloaded first, the concerning straddle carriers bring containers from vessels to stack and return empty (single move). For the loading this process is performed exactly the other way around. However, if part of the loading and unloading processes of a vessel would be done simultaneously, straddle carriers could transport import containers from vessel to stack and bring export containers on their way back. We think this will significantly increase the efficiency of the straddle carriers. A mixed integer linear program has been constructed that maximizes the opposite movements given the unload and load times for each bay and guaranteeing non-crossing of quay cranes. Since the model is quite slow, we have to find a heuristics to solve real-life instances.

In Chapter 5, we shortly introduced the principle of a shifter, defined as the handling of shifting a container to retrieve another one underneath. As has been mentioned, the number of shifters increases with the stacking height and can thus be reduced by spreading containers evenly over the blocks. However, another way to reduce the number of shifters is by intelligent stacking. Containers that are expected to leave the stack shortly should be stacked on top of containers that are expected to remain in the stack for a longer period of time. With respect to the minimization of shifters one can consider i) the tactical problem, i.e. construct a three-dimensional yard layout that is robust (with respect to shifters) to stochastic behavior of vessel arrivals and load compositions, and ii) the operational problem of how to stack containers in an online setting.

Potential applications

As discussed above, the method proposed in Chapter 5 of finding i) node (vessel) locations and ii) flows between nodes to minimize the total expected travel distance can be applied in the strategic distribution network design problem. Since the minimization of the total (expected) travel distance is one of the core objectives in strategic network design, we think this alternating method is suitable for the strategic design of many other applications (railway, air and road networks) as well.

The same is valid for the tactical decisions of allocating vessels to the different terminals in the same port. Railway and air applications for instance deal with similar problems. One main difference is that in these networks the minimization of the inter-terminal traffic (of people) is of major importance rather than spreading the workloads evenly. For an airway network the workload for unloading and loading should still be spread evenly over time, but is of minor importance. In public transportation, no effort for loading and unloading is required at all.

In Chapter 6, the online disruption management tool is constructed for one single terminal. This can be justified by the fact that in practice container vessels of one loop have a fixed terminal (according to the tactical timetable) and never deviate from it. In public transport and air applications, vehicles are often assigned a different terminal just before arrival. The online tool should therefore be expanded to consider multiple terminals simultaneously. Another difference is that the timescale in air and railway applications is much more refined and decisions have to be reconsidered each minute or even each second. The question is whether the proposed rolling horizon approach can still be solved in satisfactory time. Probably, additional heuristics are required to update decisions much faster.

Bibliography

[1] M. Ball, C. Barnhart, G. Nemhauser, and A. Odoni. *Handbooks in Operations Research and Management Science: Transportation*, chapter 1: Air Transportation: Irregular Operations and Control, pages 1–67. Elsevier B.V., 2007.

[2] L. Bertazzi, G. Paletta, and M.G. Speranza. Deterministic order-up-to level in an inventory routing problem. *Transport Sci*, 36(1):119–132, 2002.

[3] D. Bertsimas and M. Sim. The price of robustness. *Operations Research*, 52(1):35–53, 2004.

[4] J.C. Bezdek and R.J. Hathaway. *Lecture Notes in Computer Science*, volume 2275/2002, chapter Some Notes on Alternating Optimization, pages 187–195. Springer Berlin / Heidelberg, 2002.

[5] G. Caimi, M. Fuchsberger, M. Laumanns, and K.r Schüpbach. Periodic railway timetabling with event flexibility. In C. Liebchen, R. Ahuja, and J. Mesa, editors, *ATMOS - 7th Workshop on Algorithmic Approaches for Transportation Modeling, Optimization, and Systems*, pages 124–141. Internationales Begegnungs- und Forschungszentrum für Informatik (IBFI), Schloss Dagstuhl, Germany, 2007.

[6] R.K.M. Cheung and W.B. Powell. Models and algorithms for distributions problems with uncertain demands. *Transport Sci*, 30(1):43–59, 1996.

[7] J. Clausen. Disruption management in passenger transportation — from air to tracks. In C. Liebchen, R. Ahuja, and J. Mesa, editors, *ATMOS - 7th Workshop on Algorithmic Approaches for Transportation Modeling, Optimization, and Systems*, pages 30–47. Internationales Begegnungs- und Forschungszentrum für Informatik (IBFI), Schloss Dagstuhl, Germany, 2007.

[8] Jean-François Cordeau, Gilbert Laporte, Pasquele Legato, and Luigi Motta. Models and tabu search heuristics for the berth-allocation problem. *Transportation Science*, 39:526–538, 2005.

[9] J.F. Cordeau, G. Laporte, and F. Pasin. An iterated local search heuristic for the logistics network design problem with single assignment. *Int J Prod Econ*, 2006.

[10] Jim Dai, Wuqin Lin, Rajeeva Moorthy, and Chung-Piaw Teo. Berth allocation planning optimization in container terminals. Working Paper, 2004.

[11] M.S. Daskin. *Networks and Discrete Location: Models, Algorithms, and Applications.* John Wiley and Sons, Inc., 1995.

[12] R. de Borst, S.J. Hulshoff, S. Lenz, E.A. Munts, H. van Brummelen, and W. A. Wall. Multiscale methods in computational fluid and solid mechanics. In P. Wesseling, E. O nate, and J. Périaux, editors, *ECCOMAS CFD 2006*, 2006.

[13] C.E. Garcia, D.M. Prett, and M. Morari. Model predictive control: theory and practice. *Automatica*, 25(3):335–348, 1989.

[14] Michael R. Garey and D.S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. Freeman, New York, 1979.

[15] Y. Guan and R.K. Cheung. The berth allocation problem: Models and solutions methods. *OR Spectrum*, 26:75–92, 2004.

[16] P. Hansen, C. Ǒguz, and N. Mladenović. Variable neighborhood search for minimum cost berth allocation. *European Journal of Operational Research*, 191(3):636–649, 2008.

[17] Pierre Hansen, Ceyda Oğuz, and Nenad Mladenović. Variable neighborhood search for minimum cost berth allocation. Article Press, Eur. J. Oper. Res, 2007, 2007.

[18] A.C. Hax and D. Candea. *Production and Inventory Management*. Prentice Hall, Inc., 1984.

[19] Maarten Hendriks, Dieter Armbruster, Marco Laumanns, Erjen Lefeber, and Jan Tijmen Udding. Strategic alloction of cyclically arriving container vessels to inter-related terminals. *Submitted*, 2007.

[20] A. Imai, H.C. Chen, E. Nishimura, and S. Papadimitriou. The simultaneous berth and quay crane allocation problem. *Transportation Research Part E*, 44:900920, 2008.

[21] A. Imai, K. Nagaima, and W.T. Chan. Efficient planning of berth allocation for container terminals in asia. *Journal of Advanced Transportation*, 31:75–94, 1997.

[22] A. Imai, E. Nishimura, Masahiro Hattori, and S. Papadimitriou. Berth allocation at indented berths for mega-containerships. *Eur. J. Oper. Res*, 179:579–593, 2007.

[23] A. Imai, E. Nishimura, and S. Papadimitriou. The dynamic berth allocation problem for a container port. *Transportation Research Part B*, 35:401–417, 2001.

[24] A. Imai, E. Nishimura, and S. Papadimitriou. Berth allocation in a container port: using a continuous location space approach. *Transportation Research Part B*, 39:199–221, 2005.

[25] P. Jaillet, G. Song, and G. Yu. Airline network design and hub location problems. *Locat Sci*, 4(3):195–212, 1996.

[26] W.C. Jordan and S.C. Graves. Principles on the benefits of manufacturing process flexibility. *Manage Sci*, 41(4):577–594, 1995.

[27] Kap Wan Kim and Kyung Chan Moon. Berth scheduling by simulated annealing. *Transportation Research Part B*, 37:541–560, 2003.

[28] S. Kwon, K. Park, C. Lee, S.S. Kim, H.J. Kim, and Z. Liang. Supply chain network design and transshipment hub location for third party logistics providers. In *Lecture notes in computer science*, volume 3982, pages 928–933. Springer Berlin / Heidelberg, 2006.

[29] S. Lan, J.P. Clarke, and C. Barnhart. Planning for robust airline operations: Optimizing aircraft routings and flight departure times to minimize passenger disruptions. *Transportation Science*, 40(1):15–28, 2006.

[30] J.M.Y. Leung, T.L. Magnanti, and V. Singhal. Routing in point-to-point delivery systems. *Transport Sci*, 24:245–261, 1990.

[31] H. Lilliefors. On the kolmogorov-smirnov test for normality with mean and variance unknown. *Journal of the American Statistical Association*, 62:399–402, 1967.

[32] Andrew Lim. The berth planning problem. *Operations Research Letters*, 22:105–110, 1998.

[33] C.I. Liu, H. Jula, K. Vukadinovic, and P. Ioannou. Automated guided vehicle system for two container yard layouts. *Transportaion Research Part C*, 12:349368, 2004.

[34] Y. Meepetchdee and N. Shah. Logistical network design with robustness and complexity considerations. *Int J Phys Distrib*, 37(3):201–222, 2007.

[35] M.F. Monaco and M. Sammarra. The berth allocation problem: A strong formulation solved by a lagrangean approach. *Transportation Science*, 41(2):265–280, May 2007.

[36] R. Moorthy and C.P. Teo. Berth management in container terminal: the template design problem. *OR Spectrum*, 28(4):495–518, October 2006.

[37] Rajeeva Moorthy and Chung-Piaw Teo. Berth management in container terminal: the template design problem. *OR Spectrum*, 28:495–518, 2006.

[38] E. Nishimura, A. Imai, and S. Papadimitriou. Berth allocation planning in the public berth system by genetic algorithms. *Eur. J. Oper. Res*, 131:282–292, 2001.

[39] Jaap A. Ottjes, Hans P.M. Veeke, Mark B. Duinkerken, Joan C. Rijsenbrij, and Gabriel Lodewijks. Simulation of a multiterminal system for container handling. *OR Spectrum*, 28:447–468, 2006.

[40] Christos H. Papadimitriou and Kenneth Steiglitz. *Combinatorial optimization: algorithms and complexity.* Englewood Cliffs: Prentice-Hall, 1982.

[41] Young-Man Park and Kap Hwan Kim. A scheduling method for berth and quay cranes. *OR Spectrum*, 25:1–23, 2003.

[42] Michael Pinedo. *Scheduling Theory, Algorithms, and Systems.* Englewood Cliffs: Prentice-Hall, 1995.

[43] Hassan Rashidi and Edward P.K. Tsang. Container terminals: Scheduling decisions, their formulation and solutions. Submitted to the Journal of Scheduling, 2006.

[44] D. Simchi-Levi, X. Chen, and J. Bramel. *The Logic of Logistics.* Springer Science+Business Media, Inc., 2nd edition, 2005.

[45] R. Stahlbock and S. Voß. Operations research at container terminals: a literature update. *OR Spectrum*, 30:1–52, 2008.

[46] Dirk Steenken, Stefan Voß, and Robert Stahlbock. Container terminal operation and operations research-a classification and literature review. *OR Spectrum*, 26:3–49, 2004.

[47] H. Topaloglu. An approximate dynamic programming approach for a product distribution problem. *IIE Trans*, 37:697–710, 2005.

[48] P. Tsiakis, N. Shah, and C.C. Pantelides. Design of multi-echelon supply chain networks under demand uncertainty. *Ind Eng Chem Res*, 40:3585–3604, 2001.

[49] Iris F.A. Vis and René de Koster. Transshipment of containers at a container terminal: An overview. *European Journal of Opertional Research*, 147:1–16, 2003.

[50] M.J.C.M. Vromans, R. Dekker, and L. Kroon. Cyclic railway timetabling: A stochastic optimization approach. *Lecture Notes in Computer Science*, 4359:41–66, 2007.

[51] Fan Wang and Andrew Lim. A stochastic beam search for the berth allocation problem. *Decision Support Systems*, 42:2186–2196, 2007.

[52] M. Wasner and G. Zapfel. An integrated multi-depot hub-location vehicle routing model for network planning of parcel service. *Int J Prod Eco*, 90:403–419, 2004.

[53] G. Zapfel and M. Wasner. Planning and optimization of hub-and-spoke transportation networks of cooperative third-party logistics providers. *Int J Prod Econ*, 78:207–220, 2002.