Associatiefaculteit Nautische Wetenschappen

Essays on terminal optimization

Proefschrift voorgelegd tot het behalen van de
graad van doctor in de Nautische Wetenschappen
Academiejaar 2011-2012

Van Schaeren Rowan
Promotor Universiteit Antwerpen: Prof. Dr. Dullaert W.
Promotor Hogere Zeevaartschool Antwerpen: Prof. Dr. Bueken P.

Print: Silhouet, Maldegem

2012 Uitgeverij UPA University Press Antwerp
UPA is an imprint of ASP nv (Academic and Scientific Publishers nv)
Ravensteingalerij 28
B-1000 Brussels
Tel. + 32 (0)2 289 26 50
Fax + 32 (0)2 289 26 59
E-mail: info@aspeditions.be
www.upa-editions.be

ISBN 978 90 7028 934 8
NUR 969 / 977 / 982
Legal deposit D/2012/11.161/030

Examining-board

The examining-board evaluating this dissertation was composed as follows:

Prof. Dr. Peter Bueken (Antwerp Maritime Academy) - Chairman, Promoter Antwerp Maritime Academy

Prof. Dr. Wout Dullaert (University of Antwerp) - Promoter University of Antwerp

Prof. Dr. Imai Akio (Japan, Kobe University)

Prof. Dr. Theo Notteboom (University of Antwerp)

Prof. Dr. Ir. Birger Raa (University of Ghent)

Prof. Dr. Marc Vervoort (Antwerp Maritime Academy)

The hardest part

Now that this work lies behind me comes the hardest part: saying thank you. It is not difficult to do so as such but with so many people deserving of thanks it is hard to do it correctly. It is only when endeavouring to understand something thoroughly that you realise there is always more to learn. Problems or situations that might at first seem insurmountable become suddenly interesting when people with different approaches and qualities join the debate. And when these people share their knowledge, those problems become at the very least manageable if not solveable. For having shared their experiences and resources with me, I would like to thank:

First of all the Antwerp Maritime Academy that offered me the opportunity to do this research, assisting in every way possible and making the doctorate a reality. One of the many people involved once said: "I wasnt of much help". But it is exactly all that (often underestimated) input that made it happen!

To my promotor Wout Dullaert for putting up with me and keeping me on track during my sometimes wild investigations.. He kept my feet on the ground. With him came Birger Raa. Both helped me take my first (and hopefully not my last) steps in the academic world of research.

Right from the beginning it became obvious that without a close link to the industry my work would be much harder. Terminal operators, shipping lines and software companies such as PSAHNN, Stolt-Nielsen, MSC and Dynadec assisted me in keeping the research questions operationally interesting. For all the time spent debating detailed processes, program lines or the meaning of the word "handling time", I would like to say thank you to them.

And last, but certainly not least, a huge thanks to my parents, children and close friends. They sometimes had to suffer my moods or be on the receiving end of my frustration when things didn't go as planned. But they kept me going as only family and friends can. Without these, my "safety valves", disaster would certainly have struck.

I may have lost some friends during the process of this research but certainly have gained many more. Thank you ALL!

Bijdragen over terminaloptimalisaties

Internationale handel kende de laatste jaren een sterk dynamisch karakter, wat blijkt uit de sterke schommelingen in het totale import en export volume: +6.3% in 2007, +2.5% in 2008, -13.3% in 2009 en terug een stijging van 13.8% in 2010 [1]. De impact hiervan op de scheepsmarkten maar ook de terminals waar scheepsvracht behandeld wordt, is dan ook groot. De sterke marktfluctuaties, de vaak beperkte uitbreidingsmogelijkheden in havens en de concurrentiële druk dwingen terminaloperatoren om de behandelingsprocessen zo efficiënt mogelijk te organiseren.

Hoewel elke terminal zijn eigenheid heeft kan men toch een aantal processen identificeren die terug te vinden zijn op elke terminal: het toewijzen van ligplaatsen en kranen aan schepen die de terminal aandoen, het plannen van de los- en laadsequenties, het beheren van de oplsagruimtes aan de kaden alsook de interactie met de transportmodi die de verbindingen met het hinterland verzorgen (trein,lichter en vrachtwagen). Uiteraard is niet elk process even belangrijk voor elke terminal. Het zijn namelijk de knelpunten in het hele proces die de capaciteit en productiviteit van de terminal bepalen. Die kunnen van terminal tot terminal verschillen.

Containervervoer is het sterkst groeiende marktsegment binnen de internationale handel en ervaart ook sterke fluctuaties wanneer de economische activiteit terugloopt of opnieuw aantrekt. Containerterminals vormen dan ook een belangrijk toepassingsgebied voor terminal optimalisatie. De verschillende planningsprocessen worden geïllustreerd in figuur 1:

1- Pleinplanning: het plein of opslaggebied voor de containers is de draaischijf waar de containers van alle transportmodi samenkomen. Een goede planning zorgt er voor dat elke container die toekomt onmiddellijk een plaats krijgt op het plein en dat elke container die afgehaald wordt onmiddellijk beschikbaar is. Bij de pleinplanning hoort ook de planning van de *prime movers* die de containers vervoeren op de terminal zelf.

2- 3- Kraan- en ligplaatsplanning: voor een container terminal zijn de kranen de duurste investering. Een optimale benutting van deze infrastructuur is dan ook een prioriteit maar ze moet worden afgestemd op de andere processen op de

[1]UNCTAD, Trade and development report, 2011

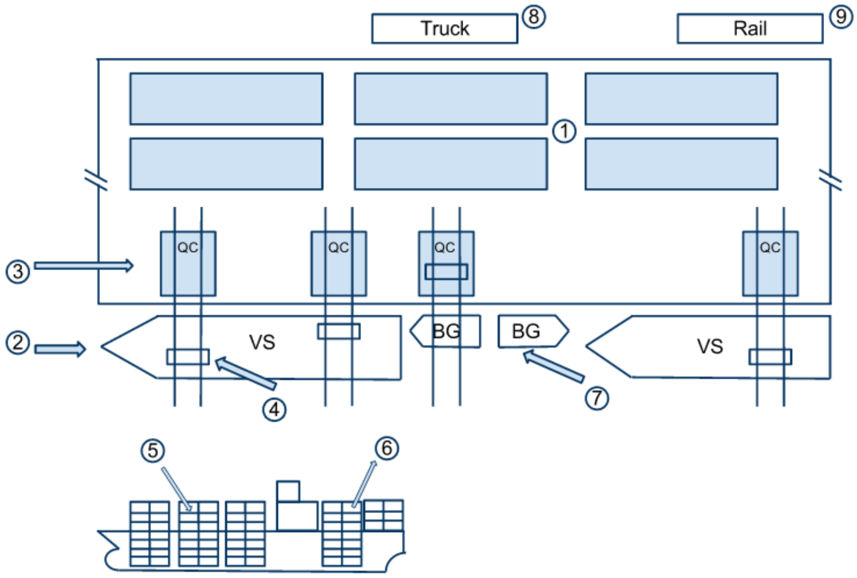


Figure 1: Overzicht van de planningsprocessen op een container terminal

terminal. Ook het toewijzen van ligplaatsen aan schepen heeft een grote invloed op de uitbatingkosten van de terminal. Indien schepen niet op een optimale plaats liggen langsheen de kade, nemen de afstanden die de *prime movers* moeten afleggen om de containers aan- en af te voeren toe. Daar de toewijzing van kranen aan een schip rechtstreeks de tijd beïnvloedt dat dit schip tegen de kade blijft liggen om geladen/gelost te worden, is het logisch dat deze beide planningsaspecten samen aangepakt dienen te worden. In dit proefschrift worden twee optimalisatiemodellen ontwikkeld om deze problematiek efficiënt aan te pakken. Praktijkgegevens worden gebruikt om de praktische haalbaarheid van deze modellen na te gaan.

4- Kraansequentie-planning: nadat kranen werden toegewezen aan schepen moet ook nog bepaald worden in welke sequentie de containers zullen gelost/geladen worden. Deze sequentie heeft een rechtstreekse invloed op de productiviteit van de kranen: ze mogen elkaar niet hinderen en het totale los- en laadpakket van een schip moet goed verdeeld worden over de beschikbare kranen.

5- 6- 7- Scheeps- en lichter los/laad -planning: het toewijzen van unieke posities aan boord van het schip voor de te laden containers en het toewijzen van unieke posities op het plein voor de te lossen containers moet geoptimaliseerd worden. Best wordt bijvoorbeeld voor het laden begonnen met de containers die bovenaan staan op het plein. Ook dienen structurele beperkingen van schepen in acht genomen te worden, bijvoorbeeld voor de gewichtsverdeling. De te lossen containers dienen een plaats op het plein te krijgen in functie van hun volgende transport:

getracht wordt om de containers slechts twee maal te verplaatsen: éénmaal van de losplaats naar een positie op het plein en vervolgens van deze positie naar het volgende transportmiddel. Stabiliteitscriteria zijn essentieel in de opmaak van de laad- en losplanning van schepen. Vooral bij het laden van chemicaliën stellen zich bijzondere nevenvoorwaarden door de aard van de cargo en van de tankers. Daarom bevat dit proefschrift een optimalisatiemodel dat voor het eerst alle stabiliteitscriteria in rekening neemt bij het laden van chemicaliëntankers. Het raamwerk biedt aanknopingsmogelijkheden voor het laden en lossen van andere scheepstypes.

8- 9- Hinterlandvervoer: door middel van weg, spoor of binnenvaart wordt de terminal met het hinterland verbonden. De *prime movers* die de containers op het plein beheren of speciaal daarvoor voorziene kranen, plaatsen de containers op de vrachtwagens of nemen deze containers af bij levering. Zowel de pleinplanning als de organisatie van het *gate-in* process heeft een grote impact op de wijze waarop deze vrachtwagens behandeld worden. Alvorens een vrachtwagen met containers tot een terminal kan toegelaten worden, dient deze zich met de container eerst bij de *gate* aan te melden voor nazicht van alle betrokken documenten (betalingen, douane, ...). Een voorgereden container dient ook eerst fysiek nagekeken te worden op gebreken en verzegelingen alvorens de terminal deze in ontvangst zal nemen. De fysieke overdracht van een container houdt namelijk ook de overdracht van verantwoordelijkheid ervoor in. In dit proefschrift wordt dit hele *gate-in* gebeuren gesimuleerd om een aantal productiviteitsverbeteringen te evalueren en het beheer van dit kritische proces te ondersteunen. Net zoals de planning voor de andere transportmiddelen spelen bij de spoorplanning spoortoewijzing en los/laad planningen een rol.

Contents

Chapter 1

Introduction

International trade has shown a very dynamic trend over the last years. This is reflected in import and export volumes worldwide: +6.3% (2007), +2.5% (2008), -13.3% (2009) and a rise again of +13.8% in 2010 [21]. This fluctuation in trade volumes has a great impact on both shipping lines and terminals servicing vessels. Faced with these combined market effects and a fierce competition, terminal operators are often limited with regard to expansion possibilities in ports, forcing them to look for system redesigns and make their processes more efficient (Daganzo [8], Legato and Mazza [14]).

****

ZIN IN SRUKKEN KAPPEN

****

The essays (Chapters 4,5, 6 and 7) presented in this work aim to contribute in this exploration by offering more operational insights concerning already known problems in the existing literature and by using a constraint programming (CP) approach in chapters 6 and 7 that, to the best of our knowledge, is used for the first time in order to solve the problems presented here. Chapter 2 gives an introduction on CP.

Although each terminal is unique, it is possible to highlight specific processes that can be found in each of them: assigning berths and cranes to vessels, planning the loading and unloading of these vessels, managing the storage locations and the interaction with the transport modes that link the terminal with the hinterland (train, barge and/or truck). It is acknowledged that not every process is equally important for terminals as it is the bottlenecks in the whole process that define the capacity and productivity of each individual terminal.

The container is the fastest growing commodity transported by sea. Container transport is closely linked with the evolutions of international trade. Notice also the drop in 2009 when considering Figure 1.1. For these reasons container terminals form a good reference when considering optimizations for terminal processes. Chapter 3 gives an overview of processes encountered at a container terminal.

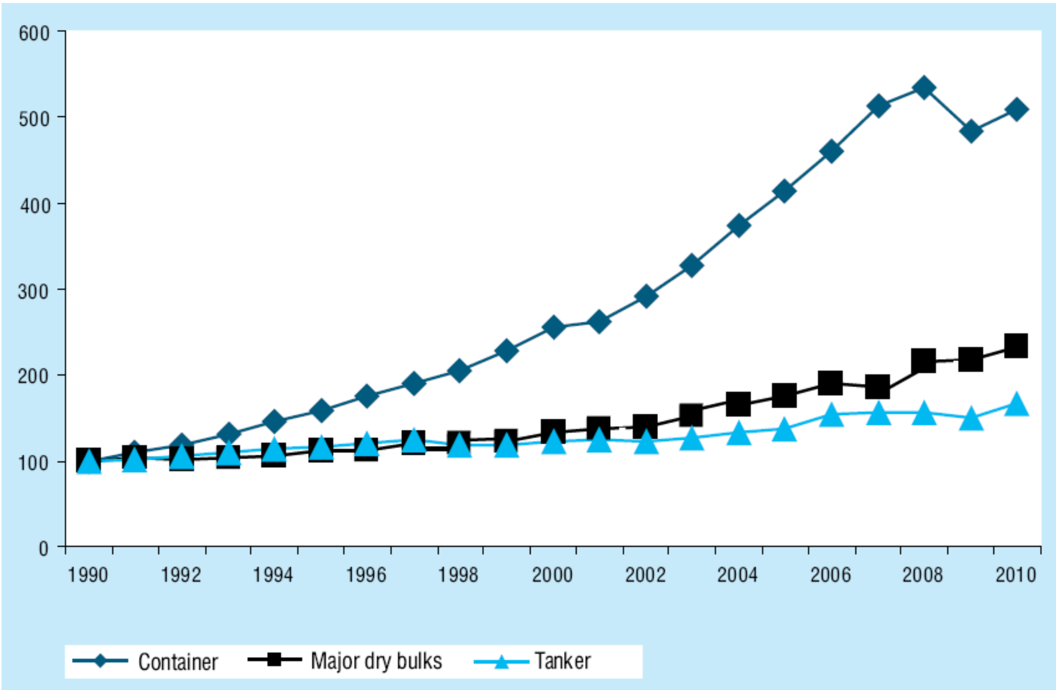


Figure 1.1: Indices for global container, tanker and major dry bulks volumes, 1990 till 2010, source UNCTAD [20]

Chapter 2

Constraint programming introduction

Constraint programming (CP) was first developed in the mid eighties as a computer science technique (Lustig et al. [5]). Since then, CP has evolved into new architectures that make it easier to combine, understand and apply (Wallace [6]). Examples of these new architectures are also given by Barth et al. [2] using PROLOG and by Lustig et al. [5] using ILOG.

A constraint program is not a statement of a problem as in mathematical programming, but rather a computer program that indicates a method for solving a particular problem (Lustig et al. [5]). Constraint programming consists of two levels: the first being the constraints that apply to the variables and the second being the description of how the variables must be adapted in order to meet the requirements of the constraints. We could view this as a constraint level and a search level. In traditional CP, the user must define an algorithm for the search level. By the 1990s, constraint programming features were introduced in general-purpose programming languages together with strong default search strategies. These search strategies can also be modified or tailored by the users.

One of the important features of CP is declarative problem modelling [6]. As the tank allocation problem is a complex operational problem, it is easier to work with understandable declarative models. Another important feature of CP is the propagation of the effects of decisions. This also proves to be very helpful for developing a TAP model as small changes in constraints readily translate in different results without compromising the complexity. This aspect is also useful for debugging the model as many variables and different inputs can easily lead to mistakes. An interesting quote in Wallace [6] hints at the possibility to solve intricate problems as the TAP using CP: "The applications are similar to those addressed by mathematical programming with the difference that mathematical programmers seek a clean model of the problem or often a simplified abstraction of the problem whilst constraint programmers revel in the messy details of practical problems!" With respect to this last feature, it is important to point out that two branches of constraint programming exist: constraint satisfaction and constraint solving (Bartk [1]). Constraint satisfaction deals with 95% of all industrial constraint applications. It uses finite domains. Constraint solving deals with solving constraints over infinite or very complex domains. Mathematicians use this method for proving whether certain constraints are satisfiable. Using CP it is possible to find just one solution, all solutions, an optimal or at least a good one. Again, this applies to the cargo scheduling problem: there are often many different solutions and the goal is to find "good" ones. Caprara et al. [4] already reported that CP has been used for solving hard combinatorial optimization problems such as scheduling, planning, sequencing and assignment problems.

Wallace [6] stated that CP in combination with LP is a powerful tool: side constraints can be used to describe and bound the problem after which a linear programming algorithm can produce an optimal solution. This combined use of CP and LP could prove very useful for the chemical tanker problem as cargo scheduling solutions are ideally optimized with regard to the stability constraints.

Barth et al. [2] point out an important advantage of CP: various types of constraints are well supported. In addition to numerical constraints, other constraint types can be used like symbolic constraints (e.g. alldifferent), global constraints (e.g. Global constraint catalog by Beldiceanu et al. [3]) or meta-constraints (e.g. a constraint reigning over other constraints).

These different possibilities of constraint formulating make it more intuitive to formulate operational constraints that are often not easily structured in pure numerical constraints.

Bibliography

[1] Bartk, R. *Constraint Programming: In Pursuit of the Holy Grail.* Proceedings of the Week of Doctoral Students WDS99. Part IV,MatFyzPress, Prague (1999) p555..564 `http://www.cs.sfu.ca/CourseCentral/417/havens/papers/CPintro.WDS99.pdf` (accessed on 08 May 2009)

[2] Barth P., Bockmayr A. *Modeling discrete optimization problems in constraint logic programming.* Annals of Operations Research 81 (1998) p467..495

[3] Beldiceanu N., Carlsson M., Rampon J.-X. *Global Constraint Catalog 2008.* `http://www.emn.fr/x-info/sdemasse/gccat/doc/catalog.pdf` (accessed on 31 July 2009)

[4] Caprara A., Focacci F., Lamma E., Mello P., Milano M., Coth P., Vigo D. *Integrating Constraint Logic Programming and Operations Research Techniques for the Crew Rostering Problem.* Software practice and experience 28(1) (1998) p49..76

[5] Lustig I. J., Puget J-F. *A Program Does Not Equal Program: Constraint Programming and its relationship to Mathematical Programming.* Interfaces 31:6 (2001) p29..53

[6] Wallace M. *Practical Applications of Constraint Programming Constraints.* An International Journal 1 (1996) p139..168

Chapter 3

Container terminal processes

Figure 3.1 and Table 3.1 represent the container flows at a container terminal with their primary equipment used.

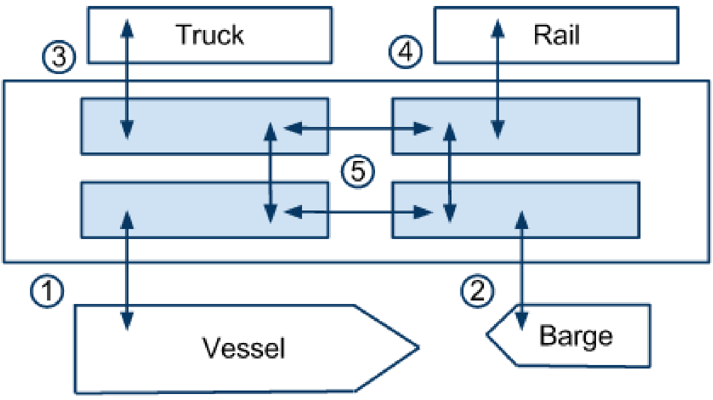


Figure 3.1: Container flows on a container terminal

We will consider physical container flows on the terminal as it will permit us to link these planning processes to operational decision making. Figure 3.2 illustrates

ID	Transport and storage	Primary equipment used
1	Vessel	Berths and quay cranes
2	Barge	Berths and quay cranes
3	Truck	Trucks, parkings, prime movers
4	Rail	Rail tracks, gantry cranes
5	Yard	Yard surface, prime movers, stacking cranes

Table 3.1: Overview of container terminal aspects

the different planning processes.

The following list gives a summary of the planning processes involved (the num-

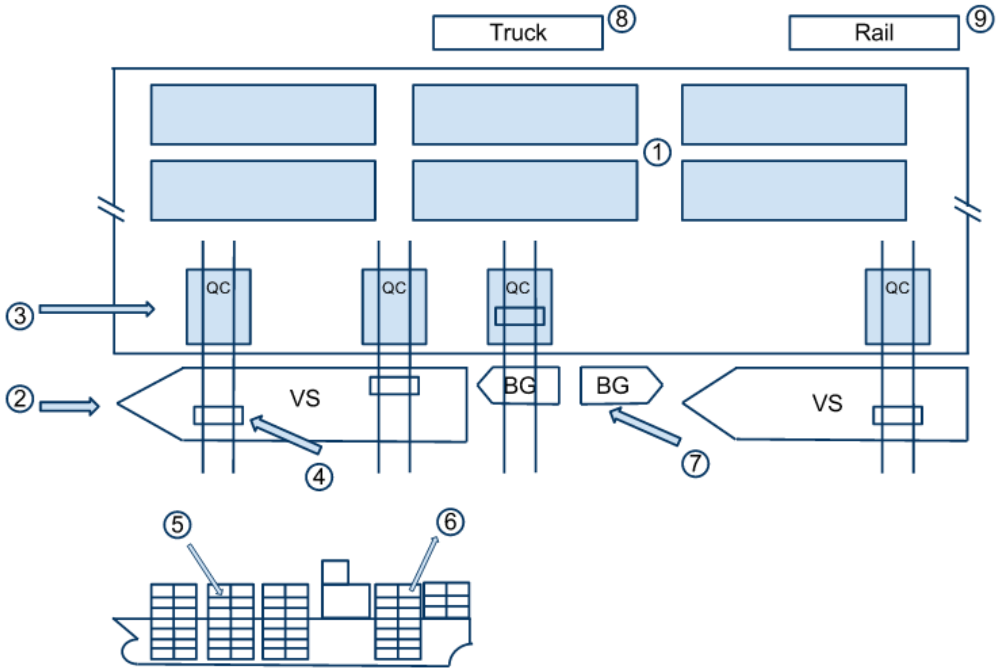


Figure 3.2: Overview of the planning processes on a container terminal

bers correspond with those represented in Figure 3.2):

1. Yard planning: assigning parts of the yard to specific containers like import, export, dangerous goods, over-sized, reefers ... Yard planning is necessary for keeping control of the available yard. It is the place where containers from all transport modes meet. Keeping the yard running smoothly has a direct impact on the cost function of the container terminal because terminal operators only get a fixed fee for each container handled. It is hard for the terminal operator to charge their clients additionally for re-handling containers on the yard itself (also called shifting or housekeeping: from a yard position to another yard position). It is therefore important that containers are stacked as perfectly as possible on the yard : the total distance traveled on the yard for each container should be minimized: position of discharge (from truck/barge/vessel/rail) $\leftrightarrow$ yard $\leftrightarrow$ position of loading (on truck/barge/vessel/rail). A recent paper on this subject is given by Rodriguez-Molins et

al. [17] in which they minimize the number of reshuffles of containers in a complete yard.

2. Berth allocation: assigning vessels to specific positions alongside the available quay length. Assigning vessels to berths can be a complex matter. When a quay length is not congested, the best position of a vessel would be in the middle of the available quay length as it is easier for all quay cranes to reach the vessel and the distance from the containers on the yard is shortest on average. When more vessels are alongside the same quay at any given time more parameters can come into play: reserved positions for the concerned containers on the yard (preferred berth problem), height and reach of the container cranes, water depths available alongside the quay, height of the superstructure of each vessel, interrelation of the vessels (for transshipment containers) ... The berth allocation process is important for a terminal operator as the berthing position of each vessel is related to the distance each prime mover must drive in order to bring the containers from the yard to the vessel or the other way round. A recent paper published on the berth allocation problem is written by Xu et al. [22] taking into account the water depth and tidal condition.

3. Quay crane assignment: assigning quay cranes to vessels in order to handle them. Quay crane planning is closely intertwined with the berth allocation process as it will define how long a certain berth will be occupied. Generally, the more cranes can be assigned to a vessel, the faster the vessel will be handled and the sooner the berth will be available for other vessels. The duration alongside the quay length will be defined in advance in a contract between the vessel operator and the terminal operator. Any deviations from these predefined handling times will generally be penalized. For the terminal operator it will therefore be a matter of adhering to the predefined duration of stay of each vessel. A recent paper concerning the quay crane assignment problem is given by Hu [11] proposing an optimal crane assignment solution taking crane shifting into consideration. An integrated approach for tackling the berth and quay crane assignment simultaneously is given in chapters 5 and 6.

4. Quay crane scheduling: the crane scheduling process defines how the assigned cranes (by the quay crane assignment process) will actually discharge and load the vessel on container level. For optimizing this process a lot of operational information is required: structure of the vessel, container positions on board of the vessel in order to define crane interferences, crane productivity for each individual crane, feeding capabilities of the cranes by prime movers like straddle carriers or other types of container transport, superstructure information of the vessel, layout of the pontoons on deck of the vessels, distances between the bays on board of the vessel, minimal distances

between the quay cranes . . . A recent paper on this subject is given by Chen et al. [7] considering the feature of quay crane scheduling at an indented berth.

5. Vessel loading planning (also called stowage planning): assigning containers to be loaded to specific positions on board a vessel. Linking container numbers with unique positions on board of vessels is generally a responsibility of the terminal operator although the final approval of the loading plan will lay with the chief officer or master on board the vessel. It is his task to check whether the proposed loading plan offers a good stability for the vessel and whether the structural restrictions of the vessel are respected. Loading plans also need to be approved by the vessel's line manager as he attunes the loading plans of all the ports the vessel calls. This supervision is important as a good repartition of the different destination ports on board of the vessel allows container terminals to assign more quay cranes to the vessel simultaneously. Generally it is the line planner of the shipping line that will tell the terminal operator in which bays the containers for a certain destination need to go. It is this *destination plan* that the terminal operator uses as a starting point for linking unique containers to unique positions on board. When assigning unique containers to positions on board of a vessel several aspects need to be taken into account: sequence of discharging, maximal stack weights both under and on deck, dangerous goods segregation regulations, marine pollutant containers, heat sources, reefer connections, oversized containers, . . . Together with the requirements concerning the vessel, the planner must also take into account the positions of the containers to be loaded on the yard: when loading containers in a predefined sequence one must first choose the containers that are readily available in order to avoid shifting/housekeeping of containers on the yard thus causing additional costs for the terminal operator. It is also necessary to check whether all the operational quay cranes are not requiring containers from the same part of the yard in order to avoid congestion of the straddle carriers or other types of prime movers in the same area. An example of a stowage problem is given in chapter 7, presenting the challenging load planning of a chemical tanker. It is the first paper that takes all stability constraints in consideration when repartitioning liquid chemicals in bulk over the cargo tanks of a chemical tanker.

6. Vessel discharge planning: assigning containers to be unloaded to specific positions on the yard. The discharge planning works best when the forecast information of the containers to be discharged is known: what is the next mode of transport by which the container will leave the terminal, when will the container leave the terminal again . . . This planning is for obvious reasons closely linked with the vessel loading planning as the vessel first needs to be discharged before it can be loaded again. A recent paper on the subject

is written by Goodchild and Daganzo [10] offering double-cycling strategies for quay cranes when loading and unloading.

7. Barge planning: assigning barges to berths along the quay side is often neglected when assigning berths and quay cranes to visiting vessels. The reason may be that a terminal does not handle barges or that barges are not treated with the same software as the vessels. At a first glance this might seem correct to do but they also take up berths and also require quay cranes for handling. It is noted that on some terminals barges are handled at dedicated berths and cranes. When handled at the same quay length as vessels they should also be considered as they use the same resources as vessels. On some terminals the number of barges is considerable, therefore they should be formally scheduled. A recent paper on container barges is given by Caris et al. [6] in which she compares freight bundling strategies for container barge transport by means of discrete event simulation.

8. Truck planning: loading and unloading of trucks. Trucks bring and pick up containers at a container terminal where they are serviced by cranes or prime movers. Together with the assignment of prime movers or cranes, gate-in and -out are crucial parts in a truck's visit. The details of managing a gate-in process are studied in chapter 4.

9. Rail planning: the allocation of trains to tracks and the allocation of cranes to handle the trains. Like vessels, trains need to be loaded taking into account several operational criteria. A recent paper on train load planning in seaport container terminals is given by Ambrosino et al. [1] suggesting a heuristic approach for the train load planning problem of import containers.

An extended literature review on operations research at container terminals can be found in Stahlbock and Voß[18].

Chapter 4

Gate-in simulation

This section proposes a detailed simulation of the gate-in process. The question asked is whether the encountered waiting times at the gate-in could be reduced. This quest for making the gate-in process more efficient is not only driven by the container terminals but also by the transporting companies (trucks) that call the container terminal on a daily basis for whom any decrease in waiting time could mean more net driving time. Any non-efficiency encountered at the gate-in may also create a congestion problem on the public roads at which the container terminal is located.

When trying to make container terminal processes more efficient two approaches are used in literature: simulation and optimization. Arena et al. [3] and De Mol et al. [9] both compared the utilization of simulation and optimization on a specific topic. Although it might be possible to bring simulation and optimization on the same level, differences will remain. Optimization is better used for decisions on strategic levels while simulation is preferred when the network structure is known or when only a small number of possibilities are taken into account. It is for these reasons that the gate-in process will be approached from a simulation point of view as it proved useful for evaluating the impact of different resource allocations while the other chapters in this work use an optimization approach. As also described by Lai and Leung [13] simulation can be a valuable tool for evaluating new approaches for the gate-in process. They studied three expansion policies for coping with an increased amount of traffic based on an evaluation of four identified critical factors: number of parking lanes, number of computer terminals, average time of "gate create" and number of parking spaces. For a recent overview of the available academic papers using simulation for container terminal management we refer to Angeloudis and Bell [2]. Our proposed simulation could be defined as dynamic, microscopic and focused in the classification of Angeloudis and Bell [2].

Recent literature advocates an integrated container terminal simulation approach (e.g. Bielli et al. [4], Sun et al. [19]). Nevertheless, we propose an in-depth description of the truck gate-in process of a container terminal. To the best of our knowledge this is the first paper that presents this level of operational detail concerning the gate-in process.

The gate-in process consists of two main components:

- administrative check (A-Check): during this check the truck visit is registered or confirmed when the terminal works with pre-announcements for truck visits to get access to the terminal. In this phase all the documents (commercial and customs-related) are validated.
- physical check (P-Check): before allowing a new container on the terminal and taking responsibility for it, the container needs to be checked e.g. for damage, seal number(s) ... This transfer of responsibility is an important aspect because as soon as the container is positioned on the container yard, the terminal operator takes liability for the container. In the case when a truck only needs to pick up a container or multiple containers, this phase is kept to a minimum: the truck activates its visit to the terminal by passing through a gate with its unique truck visit identification created during the A-Check.

It is only after these two main phases have been properly executed and validated that a truck will be allowed on the terminal. The detailed description of the proposed iGrafx [12] simulation together with all the required data is given in the following section. In section 4.2 the results are explained and two additional scenarios are presented. Section 4.3 presents our conclusions and suggestions for further research.

4.1 Simulation details

The structure of this section corresponds with the movements of a truck throughout the gate-in process as depicted in Figure 4.1:

1. arrival
2. driving to the A-Check area
3. A-Check building
4. driving to the P-Check area
5. P-Check area

After the P-check area the truck will proceed to the actual container terminal and thus leaving this simulation. Care should be taken that the gate-in operation details may vary due to the use of automation or local practices. All details described in this section give an example of the practices observed at terminals in the port of Antwerp, but the framework can be easily extended to account for the particulars of any container terminal in the world. Figure 4.1 gives an overview of the complete process.

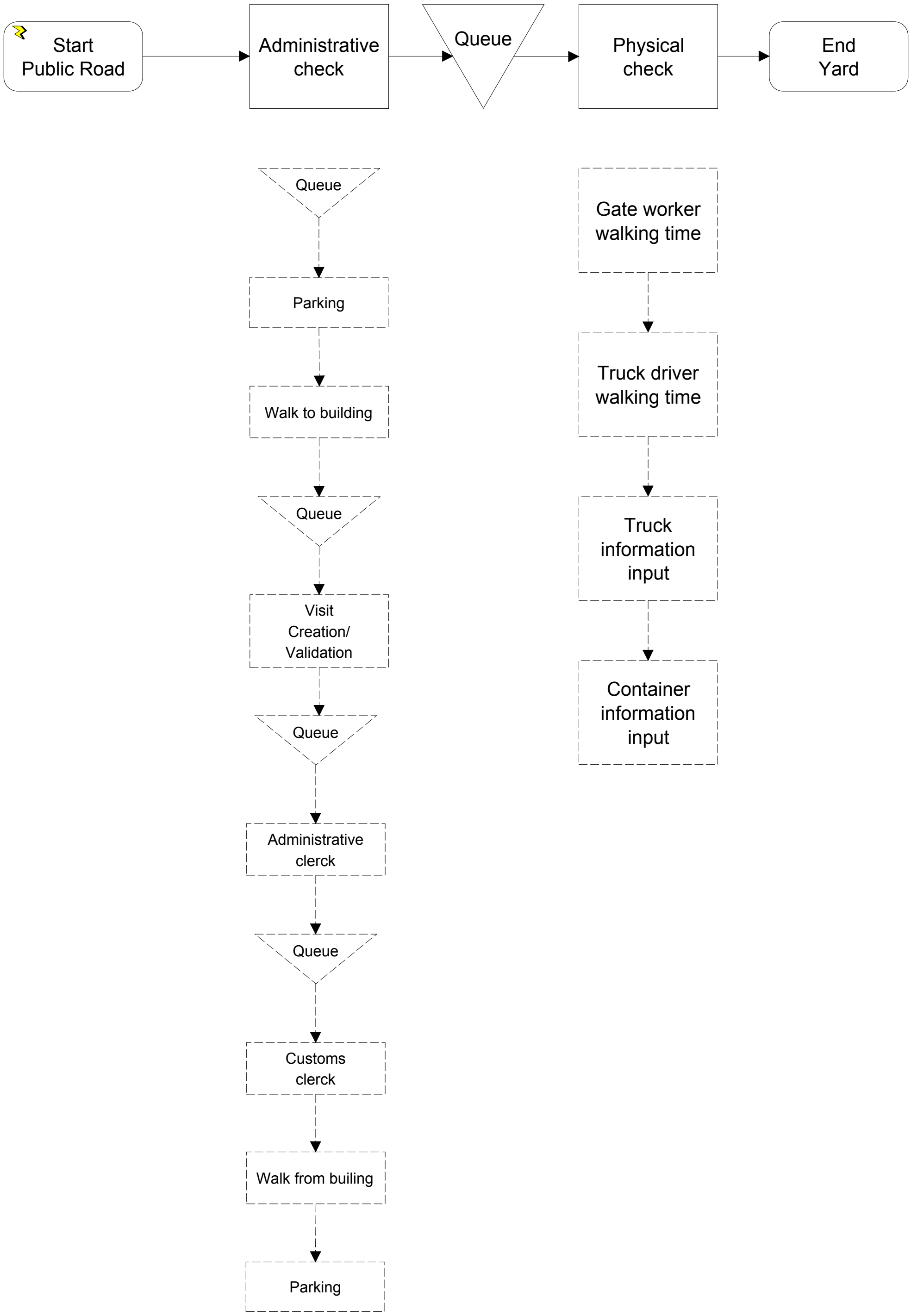


Figure 4.1: Schematic overview of the gate-in process

4.1.1 Arrival

All trucks calling at the terminal approach from the public road. The arrival pattern of the trucks in our simulation study is derived from observations at several major container terminals in the port of Antwerp. Most of the arrivals at the Antwerp container terminals occur between six in the morning and ten in the evening. As stated by Legato and Mazza [14] it is hard to find large amounts of terminal data over sufficiently long periods. For validating the proposed simulation we encountered the same concerns. Based on several observations and measurements though, we were able to construct Table 4.1. The table represents the distribution of arrivals over a period of sixteen hours. For confidentiality reasons we can not report the actual arrival rate per hour.

Gate operating hours	Percentile distribution
00-01	4.38
01-02	6.79
02-03	5.72
03-04	5.62
04-05	2.88
05-06	9.14
06-07	8.51
07-08	4.66
08-09	9.82
09-10	10.50
10-11	10.28
11-12	8.99
12-13	2.39
13-14	6.71
14-15	3.20
15-16	0.42

Table 4.1: Percentile arrival pattern of trucks per hour over a period of sixteen hours

The arrivals within each hour are considered to be exponentially distributed.

The container types (normal, special or empty) are important to take into consideration as the time needed to register the container at the A-check varies with the details that need to be registered. The container types are:

- normal (N): a normal container
- special (S): a reefer container or a container containing dangerous goods. Reefer temperatures as well as dangerous goods labels need to be registered

in addition to the general information of for a normal container (container number, ISO number, damage codes . . .).

- empty (E): an empty container. An empty container needs to be inspected visually to ascertain that the container is in fact empty and clean on the inside.

Because no details are registered concerning container types at the terminals under consideration, assumptions had to be made that were validated by personnel on the work-floor afterwards. A summary of the assumptions made concerning the types of containers is detailed in Table 4.2. The percentile figures represent the distribution of the types of containers at the terminal.

<table>
<tr><td colspan="11">All Gate Trucks</td></tr>
<tr><td colspan="11">100%</td></tr>
<tr><td colspan="9">Delivery</td><td colspan="2">PickUp</td></tr>
<tr><td colspan="9">75%</td><td colspan="2">25%</td></tr>
<tr><td colspan="3">1 container</td><td colspan="6">2 containers</td><td>1 PickUp</td><td>2 PickUps</td></tr>
<tr><td colspan="3">67,50%</td><td colspan="6">7,50%</td><td>17,50%</td><td>7,50%</td></tr>
<tr><td colspan="2">Full</td><td>Empty</td><td colspan="3">2 Full</td><td colspan="2">1 Full 1 Empty</td><td>2 Empty</td><td></td><td></td></tr>
<tr><td colspan="2">40,50%</td><td>27,00%</td><td colspan="3">4,05%</td><td colspan="2">1,73%</td><td>1,73%</td><td></td><td></td></tr>
<tr><td>N</td><td>S</td><td></td><td>NN</td><td>NS</td><td>SS</td><td>NE</td><td>SE</td><td></td><td></td><td></td></tr>
<tr><td>27,14%</td><td>13,37%</td><td></td><td>2,03%</td><td>1,26%</td><td>0,77%</td><td>0,98%</td><td>0,74%</td><td></td><td></td><td></td></tr>
</table>

<table>
<tr><td colspan="3">After delivery also PickUp?</td></tr>
<tr><td colspan="3">75%</td></tr>
<tr><td colspan="2">Yes</td><td>No</td></tr>
<tr><td colspan="2">37,50%</td><td>37,50%</td></tr>
<tr><td>1 PickUp</td><td>2 PickUps</td><td></td></tr>
<tr><td>26,25%</td><td>11,25%</td><td></td></tr>
</table>

Table 4.2: Types of containers handled

The table is read as follows:

- Empty (27%): out of all the trucks calling the container terminal in one day, 27% of the trucks brings one empty container.

- NS (1.26%): out of all the trucks calling the container terminal in one day, 1.26% of the trucks brings two containers: one normal and one special container.

- 1 pickUp after delivery (26.25%): out of all the trucks calling the container terminal in one day, 26.25% of the trucks also pick up a container on the terminal after delivering one or more containers.

- PickUp (25%): out of all the trucks calling the container terminal in one day, 25% of the trucks only need to pick up one or two containers without having to deliver one or two containers.

4.1.2 Driving to the A-Check area

For our simulation we used a fixed thirty seconds to drive to the parking area of the administrative building. The number of parking places is restricted to sixty. When the truck arrives at the parking area, one parking place will be reserved for it until the truck leaves the parking area again (acquire and release of a parking resource). If all the available parking spaces are occupied upon arrival, the truck will wait until one becomes available. The parking resource will only be released when the truck leaves the parking area in order to drive to the P-Check area (section 4.1.4). From here the truck driver needs to get out of his truck and walk inside the building (taking him sixty seconds). All the durations of the tasks mentioned in this section are averages of observations conducted by two independent persons. As these durations are considered deterministic for the simulation, we understand that this causes the possible waiting times to be lower than they actually are.

4.1.3 A-Check building

After entering the administrative building the truck driver will register using one of the fifteen available computers. The computer resource is acquired until the truck driver has a printout of the validation of his visit (see also further in this section). Identification happens by means of an electronic identification card used in the port are of Antwerp called Alfapass (`http://www.alfapass.be`). A small percentage (2%) of the truck drivers visit the port area for the first time and therefore need to obtain an Alfapass on site. The process of obtaining this electronic identification card and creating the truck visit takes two minutes. The registration and validation of a container on a computer requires sixty eight seconds. If two containers need to be registered an additional thirty seconds is counted. After the visit has been validated and the concerned containers have been registered, the driving instruction and validation papers are printed (five seconds). If a full container is involved a customs officer needs to confirm the visit (sixty seconds). In a small percentage of the visits (2%) there will be a problem with either the visit or the container(s) requiring manual intervention (one minute) by a clerk. In total three customs officers and three clerks are available throughout the opening hours of the gate.

4.1.4 Driving to the P-Check area

After releasing the parking resource at the A-Check the truck needs to drive to the P-Check area (taking him two minutes). In case of a congestion at the P-Check area, ample parking places are available.

	Gate workers available	Breaks
06:00-14:00	2	10:00-10:30
10:00-18:00	1	14:00-14:30
14:00-22:00	2	18:00-18:30

Table 4.3: Availability of gate workers at the P-Check

4.1.5 P-Check area

The physical check is being done manually for trucks delivering a container and automatically for trucks only retrieving one or multiple containers. When only picking up containers, the truck uses an automatic gate system where the driver needs to punch in his visit reference obtained at the A-Check. After validation the truck is admitted to the terminal. This process takes in total eleven seconds.
When delivering a container a gate worker has to do the following operations (fifty three seconds in total):

- input truck license plate: six seconds
- input transport company: ten seconds
- walk to, from and around the truck: nine seconds
- input container number: eight seconds
- input container ISO number: six seconds
- input seal type and number: eight seconds
- input container damage codes: six seconds

When a second container needs to be registered only the last five items from the previous list need to be repeated. If a special or empty container is involved the gate worker also needs to check the following items:

- inspect the cleanliness of the empty container: six seconds
- input the reefer or dangerous goods information: eight seconds

Before and after this process the truck driver also needs to get out of his cabin in order to assist the gate worker (twenty seconds). Table 4.3 represents when and how many gate workers are available during the opening hours of the gate-in.

Average	18.4309	
Median	18.4250	
Standard deviation	0.3631	
99.5% Confidence interval	18.3930	18.4688
99% Confidenceinterval	18.3963	18.4655
95% Confidence interval	18.4047	18.4571

Table 4.4: Statistical information for the average waiting times of the trucks (in minutes) for the baseline simulation scenario

4.2 Simulation model and results

For comparing the simulation scenarios to the current operational practice a baseline scenario is created containing two thousand trucks over a period of sixteen hours. The baseline scenario is a representation of the current practices.

Later in this section (4.2.2) also the results of two alternative scenarios will be discussed. All inputs used for the baseline scenario results were detailed in section 4.1. Unless stated otherwise the same inputs are used for the alternative scenarios.

4.2.1 Baseline scenario

For evaluating the results of a simulation the inputs and outputs can be compared to operational results. As stated earlier in section 4.1.1 it is sometimes hard to find sufficient empirical data to properly validate a simulation. As we face the same problem, we had to fall back on experiences from the users to acquire all the inputs and to validate the outputs. What was validated objectively was the average waiting time a truck experienced between the A-Check and the end of the P-Check. These times were registered by a major container terminal over a period of six months. The baseline simulation almost exactly matches the observed data (difference less than one minute). Table 4.4 represents the statistical information concerning the output for the average waiting time by the trucks for the whole simulation using the baseline.

4.2.2 Simulation scenarios

For the scenarios we examined the work delivered by the gate-in workers at the P-Check. Two major reasons triggered this approach:

- Yates analysis: an analysis of the used resources during the simulation proved that the P-Check had the largest influence on the average waiting times of the truck. As changes in infrastructure often are hard to achieve on a busy container terminal, only the resources that can be modified at

	Levels	
	Low	High
Gate worker	2	3
Parking spaces	60	70
Administrative clercks	3	5
Custom clercks	3	5
Computers	15	20
Lanes for retrievals at P-Check	2	3

Table 4.5: Factors and levels used as input for the Yates analysis

an operational level are analyzed using a Yates analysis. For further information on the Yates analysis we would like to refer to the website of the Information Technology Laboratory (ITL) at the National Institute of Standards and Technology (NIST) (`http://itl.nist.gov/div898/handbook/eda/section3/eda35i.htm`). Table 4.5 details the so called factors considered with their levels. Each combination was run ten times using different random numbers. Figure 4.2 represents the results in the form of a Pareto chart with *Alpha* representing the limit of statistical significance and *Lenth's PSE* Lenth's pseudo standard error. Figure 4.3 represents the main effect plot with on the vertical axis the waiting times expressed in minutes and on the horizontal axis the levels of the factors considered.

- terminal responsibility: a container terminal during the gate-in process has only an impact during the P-Check as the trucks are still able to leave the terminal between the A-Check and the P-Check. This means that concerning waiting times the container terminal can only influence the waiting times of a gate-in process during the P-Check.

It would be interesting to know therefore if some small adjustments could be useful for reducing these waiting times during the P-Check. The following scenario is therefore examined: what would happen to the waiting times if the input time by the gate-workers were to be reduced?

For scenario one we assume that inputting times done by the gate-worker can be halved. We consider this to be a realistic scenario because it does not change the process too much and is technically feasible. This assumption is also supported due to the fact that more and more electronic pre-announcements are used containing more container details. Table 4.6 represents the original and the new inputting times. The times for all other processes remain the same.

Scenario two goes one step further than scenario one: instead of having a gate worker from ten AM till six PM, we only keep two gate workers in the morning and two gate workers in the afternoon. Table 4.7 compares the statistical values

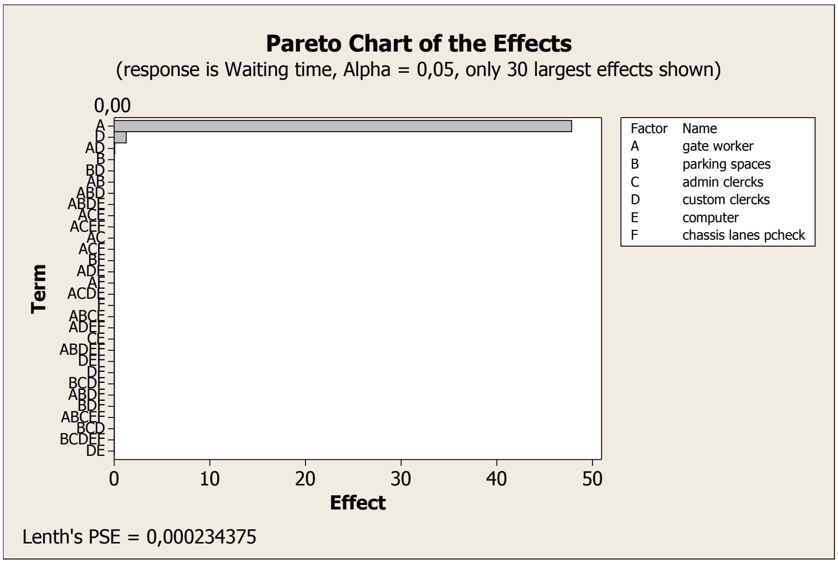


Figure 4.2: Yates analysis for the resources used during the simulation represented using a Pareto chart

for the different simulations run for the average waiting time of the trucks over the whole gate-in process.

In order to be able to compare the results we first need to ascertain whether the obtained results are not due to chance (triggered by random effects).This test can be done by stating a "Null hypothesis" and challenging that hypothesis by means of a T test. The "Null hypothesis" here is the statement that results are due to random and that the results can not be used. To be able to do this T test an assumption needs to be made: the standard deviations of both the datasets need to be identical. This assumption is justified by the facts that (i) both the histograms of the datasets are normally distributed (see Figure 4.4) and (ii) the average waiting times is a summation of different small waiting times that are independent of each other. With a 198 degrees of freedom and a T value of 192.208 we find that the "Null hypothesis" can be rejected with a probability higher than 99.9%. Meaning that the observed differences are statistically significant.

From the scenarios we can deduce that a small technical adjustment might cause waiting times to drop significantly. Care should be taken not to dismiss the gate-worker during the day shift too easily (scenario two). It would cause

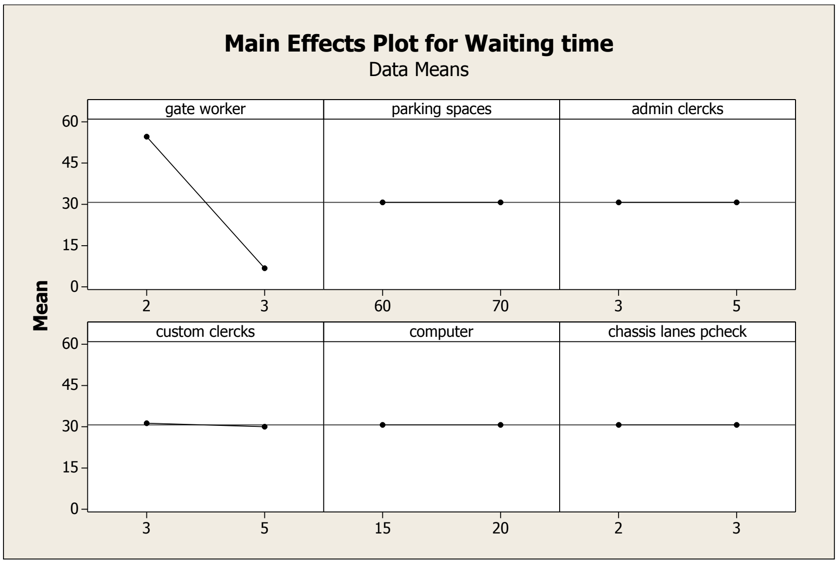


Figure 4.3: Main effects plot for the waiting time

the waiting times to increase by an average of two minutes per truck. Although this increase might seem insignificant, two minutes might be just enough to cause other reactions like stress increase and unhappiness for the transport companies thus causing other negative side effects.

4.3 Conclusion

This paper presents a simulation of the gate-in process in full operational detail as can be encountered in the port of Antwerp. Only a few papers provide this level of detail. Simulation results are statistically validated. This paper confirms that simulation on a very detailed operational level can also contribute to a better understanding of processes and allows stakeholders to gain an idea of the impact when changes to processes are considered. It also proves that small changes in service times can have a major impact on waiting times (-62%) and resource management. As service times of the processes are considered deterministic in this simulation it is, however, highly probable that waiting times are more significant than experienced. This simulation tool can be used to help improve the overall efficiency at gate-in's for container terminals. A cost benefit analysis including

Input field	Baseline	Case study 1
License plate	6	3
Transport company	10	5
Container number	8	4
Container ISO number	6	3
Seal type and number	8	4

Table 4.6: Scenario consequences on the input times expressed in minutes

	Baseline	Scenario 1	Scenario 2
Average	18.4309	7.0095	20.9188
Median	18.4250	6.9700	20.9150
Standard deviation	0.3631	0.7578	0.40422
99% Confidence interval for the average	18.3374 18.5244	6.8143 7.2047	20.8147 21.0229

Table 4.7: Statistical information for the average waiting times of the trucks for the different simulations expressed in minutes

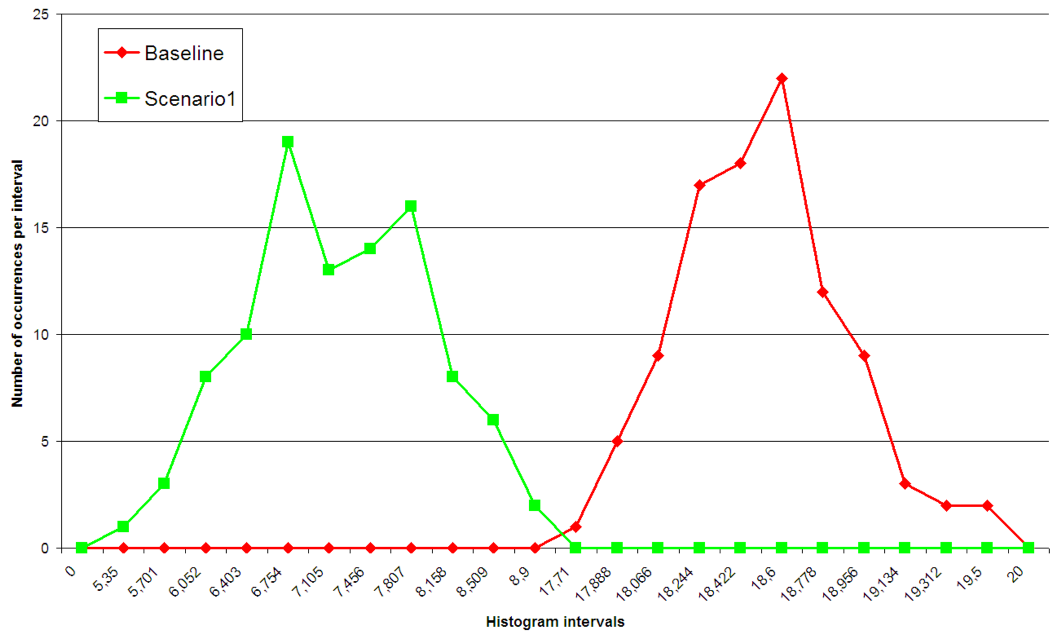


Figure 4.4: Histograms for the Baseline and scenario one datasets concerning the average waiting times

external costs can be used to examine whether the proposed approach can be implemented or not.

Bibliography

[1] Ambrosino D., Bramardi A., Pucciano M., Sacone S., Siri S. *Modeling and solving the train load planning problem in seaport container terminals.* International Conference on Automation Science and Engineering (2011) p208..213

[2] Angeloudis P., Bell M.GH. *A review of container terminal simulation models.* Maritime Policy and Management vol. 38 no. 5 (2011) p523..540

[3] Arena C., Fortunato A., Mazzola M.R. *Simulation versus optimization in the assessment of the resource opportunity cost in complex water resources systems the case of Agri-Sinni in southern Italy.* 18th World IMACS / MODSIM Congress, Cairns, Australia 13-17 July (2009) p3237..3244 `http://www.mssanz.org.au/modsim09/I3/arena.pdf`

[4] Bielli M., Boulmakoul A., Rida M. *Object oriented model for container terminal distributed simulation.* European journal of operational research vol. 175 no. 3 (2006) p1731..1751

[5] Bierwirth C., Meisel F. *A survey of berth allocation and quay crane scheduling problems in container terminals.* European Journal of Operational Research 202 (2010) p615..627

[6] Caris A., Macharis C., Janssens G.K. *Network analysis of container barge transport in the port of Antwerp by means of simulation.* Journal of Transport Geography 19 (2011) p125..133

[7] Chen J.H., Lee D.-H., Cao J.X. *Heuristics for quay crane scheduling at indented berth* Transportation Research Part E 47 (2011) p1005..1020

[8] Daganzo C.F. *The productivity of multipurpose seaport terminals.* Transportation Science vol. 24 no. 3 (1990) p205..216

[9] De Mol R.M., Jogems M.A.H., Van Beek P., Gigler J.K. *Simulation and optimization of the logistics of biomass fuel collection.* Netherlands Journal of Agricultural Science 45 (1997) p219..228 `http://www.biomassandbioenergy.nl/filesdwnld/NJAS96.pdf`

[10] Goodchild A.V., Daganzo C.F. *Double-Cycling Strategies for Container Ships and Their Effect on Ship Loading and Unloading Operations.* Transportation Science vol. 40 no. 4 (2006) p473..483

[11] Hu Z.-H. *A quay crane assignment approach for berth allocation problem in container terminal.* Proceedings of the Third International Symposium on Intelligent Information Technology and Security Informatics (2010) p17..21

[12] iGrafx `http://www.igrafx.com/`

[13] Lai K.K., Leung J. *Analysis of gate house operations in a container terminal.* International Journal of Modelling and Simulation vol. 20 no. 1 (2000) p89..94

[14] Legato P., Mazza R. *Berth planning and resources optimization at a container terminal via discrete event simulation.* European Journal of Operational Research 133 (2001) p537..547

[15] Legato P., Mazza R. *Simulation-based optimization for discharge/loading operations at a maritime container terminal.* Operations Research Spectrum 32 (2010) p543..567

[16] Raa B., Dullaert W., Van Schaeren R. *An enriched model for the integrated berth allocation and quay crane assignment problem.* Expert Systems with Applications vol. 38 no. 11 (2011) p14136..14147

[17] Rodriguez-Molins M., Salido M.A.,Barber F. *Intelligent planning for allocating containers in maritime terminals.* Expert Systems with Applications 39 (2012) p978..989

[18] Stahlbock R., Vo S. *Operations research at container terminals: a literature update.* Operations Research Spectrum 30 (2008) p1..52

[19] Sun Z., Lee L.H., Chew E.P., Tan K.C. *MicroPort: A general simulation platform for seaport container terminals.* Advanced Engineering Informatics, in press (2011) doi:10.1016/j.aei.2011.08.010

[20] UNCTAD *Review of maritime transport 2010.* `http://www.unctad.org/en/docs/rmt2010_en.pdf`

[21] UNCTAD *Trade and development report, 2011.* `http://www.unctad.org/en/docs/tdr2011_en.pdf`

[22] Xu D., Li C.-L., Leung J. Y.-T. *Berth allocation with time-dependent physical limitations on vessels.* European Journal of Operational Research 216 (2012) p47..56

Chapter 5

An enriched model for the integrated berth allocation and quay crane assignment problem

Authors:
Birger Raa, *Ghent University*, birger.raa@ugent.be
Wout Dullaert, *University of Antwerp, ITMMA, Antwerp Maritime Academy*, wout.dullaert@ua.ac.be
Rowan Van Schaeren, *Antwerp Maritime Academy*, rowan.van.schaeren@hzs.be

5.1 Abstract

Given the increasing pressure to improve the efficiency of container terminals, a lot of research efforts have been devoted to optimizing container terminal operations. Most papers deal with either the Berth Allocation Problem (BAP) or the (Quay) Crane Assignment Problem (CAP). In the literature on the BAP, handling times are often simplified to be berth dependent or proportional to vessel size, so the CAP can be ignored when scheduling vessels. This is unsatisfactory for real-life applications because the handling time primarily depends on the number of containers to be handled and the number of cranes deployed. Only a limited number of papers deal with the combination of berth allocation and crane assignment. In these papers however, authors often have resorted to algorithmic simplifications that limit the practical use of the models. This paper presents a MILP model for the integrated BAP-CAP taking into account vessel priorities, preferred berthing locations and handling time considerations. The model is used in a hybrid heuristic solution procedure that is validated on real-life data illustrating the potential to support operational and tactical decision-making.

Keywords: container terminal; berth allocation; quay cranes; mathematical modeling

5.2 Introduction

Since the 80s, the annual growth rate of seaborne trade has been 3.7 percent on average (Grossmann et al., [8]). Container growth rates, however, have been significantly higher. According to leading maritime analyst Drewry Shipping Consultants [6][7], the number of full TEUs shipped on worldwide trade routes more than doubled from 69.6 million TEU in 2000 to 141.2 million teu in 2007, representing an average annual growth rate of no less than 10.6%. This growth rate is expected to continue in the short-term future: by the year 2012 Drewry forecasts a worldwide container traffic of 223.7 million full TEUs, i.e. an increase of nearly 60% compared to the 2007 figure. It is currently unknown how these forecasts should be adjusted in terms of the recent economic crisis.

Additional container handling is generated by the hub-and-spoke strategy, in which larger ports (hubs) serve as ports of call and smaller ports (spokes) offer additional cargo via feeder lines. Figures on total throughput handled by the world's ports are therefore more suited to illustrate the increasing demand for container handling capacity. For 2007, the total volume handled at the world's ports is estimated at 493.2 million teu (including empties and transshipment), a figure expected to increase by some 57% up to 773.7 teu in 2012 (Drewry Shipping Consultants [6][7]).

As argued in Vernimmen et al. [31], many shipping lines have anticipated the increased demand for container transport by ordering additional and larger vessels. According to AXS-Alphaliner [1], the total cellular containership fleet at 01/01/2008 consisted of 4320 vessels for a combined capacity of 10.92 million teu slots. Based on the shipping lines' order books as at 01/04/2008, these figures are expected to increase to 5813 vessels and 17.69 million teu, respectively, by 01/01/2012. Hence, the total slot capacity provided by the world cellular fleet will increase by more than 60% in four years time, or nearly 13% per year.

In contrast, many planned investments in additional container terminal infrastructure in Northern European ports (such as Le Havre, Antwerp, Rotterdam, Wilhelmshaven, Flushing and ports in the UK) have been delayed for several years or even cancelled altogether. If all these proposed projects had been realized in accordance with their original time schedule, an extra capacity of no less than 11.4 million teu (nearly one third of the capacity available in 2004) would have been available in North European ports in 2005 (Vernimmen et al. [31]).

Increasing container handling capacity by expansion projects appears to be difficult for environmental, financial, technical and legal reasons. In many cases there is even no land available to build additional infrastructure. Optimizing the processes of existing infrastructure is therefore often a better - if the not only - way to increase handling capacity.

The productivity of a container terminal is determined by the interaction of a number of processes. Based on the academic literature devoted to them, the best-known processes are probably berth planning (which allocates vessels at the available quays) and quay crane planning (which assigns the available cranes to the vessels alongside the quays). Other important, but less studied processes are yard planning (for allocating all the containers handled by the terminal on a yard), vessel planning (positioning of the containers on board of vessels) and labor planning (assigning people to all the jobs to be carried out). An OR literature overview of container terminal operations is given by Stahlbock and Voß[28]. A dedicated literature review on berth allocation and quay crane scheduling problems is presented in Bierwirth and Meisel [2].

This paper focuses on the berth planning and quay crane planning processes, the most studied container terminal processes from the academic literature. Section 5.3 presents a focused literature review on the Berth Allocation Problem (BAP) and the Crane Allocation Problem (CAP). In Section 5.4, we propose an extended model for the combined BAP and CAP, accommodating some of the shortcomings of the existing models in the literature. This model is relaxed to obtain an approximation method in Section 5.5 that is validated using real-life data. Section 5.6 concludes and offers directions for further research.

5.3 Literature review and insights

In the literature, different names are being used to denote the time that vessels stay alongside the quay: e.g. processing time (Guan and Cheung [9]), handling time (Imai et al. [12] [16]) and duration of operation (Wang and Lim, 2007). Not only the terminology is sometimes different, also its meaning tends to differ. For example, Nishimura et al. [25] and Imai et al. [15] assume handling time to be berth dependent, whereas Guan and Cheung [9] assume handling times to be proportional to vessel size. Because the number of containers to be handled (call size or workload) does not have to be proportional to vessel size, we do not want to make this assumption. We therefore define the workload of a vessel as the number of containers to be handled and handling time as the total time needed for handling. Handling time is influenced by several aspects such as: workload, interruptions during the loading/unloading process, the number and types of cranes available/used for loading and unloading, ability of the crane driver using the crane, number of prime movers feeding the cranes. In particular situations the assumption of the handling time being proportional to the vessels length can be justified as described by Tang et al. [29]. They propose two mathematical models for dynamically scheduling vessels to multiple continuous berth spaces in an iron and steel complex.

As many container berths are privately operated and because of the impact of terminal operations on a terminals productivity and competitive position, several papers have already been published on optimizing container terminal operations.

Most of these papers deal with BAP and CAP workload optimization.

Lai and Shih [18] propose a heuristic algorithm for berth allocation with a first-come-first-serve (FCFS) queuing discipline for vessels. Brown et al. [3][4] present an integer-programming model for vessel berthing in naval ports. They consider a quay with a finite set of berths at which berth shifting of moored vessels is allowed, a practice uncommon in commercial ports. Imai et al. [11] develop a BAP for commercial ports questioning the FCFS principle. Instead, they suggest a heuristic to find solutions maximizing berth performances whilst minimizing changes in the vessels order of service. They assume a static BAP (SBAP), implying that all vessels to be served are present in the port before starting the planning of the berth allocation. To improve the practical relevance of the model, they extend it to a dynamic one (DBAP) (Imai et al. [12] [14]) also including different water depths at the berths (Nishimura et al., [25]) and vessel priorities (Imai et al. [13]).

Already in 2001, Legato and Mazza [19] rightfully pointed out that vessels can have different priorities for receiving service at a container terminal. They consider a case in which there are two sets of berths available. Primary vessels are handled upon arrival and have dedicated (or reserved) berths. This is enforced by assigning them a high priority. Secondary vessels are served according to a FCFS rule. If only one quay is available, both types of vessels have to compete for the same berths. It is our belief that reserving berths for vessels is not the most efficient approach for guaranteeing high service levels and optimizing the BAP. Not only the service of a subset of the vessels needs to be optimized, the berth allocation of all vessels should be optimized simultaneously. Moreover, it may also be possible that the reserved berth is unavailable, e.g. when a vessel with a higher priority is still alongside the quay or when maintenance work is taking place (crane maintenance, dredging). Focus should then be on servicing high priority vessels as close to their preferred berthing position as possible. We therefore prefer not to formulate a direct relationship between priority and berth position. Instead, we suggest only taking priorities into account when balancing the costs of two vessels competing for the same berth. The reasons for prioritization can be numerous: operational, commercial, number of containers to be handled, emergencies, tide restrictions, transshipment aspect between vessels etc. Priorities like in Legato and Mazza [19] should be used for making sure that vessels are served as soon as possible, not for reserving their actual berthing location. Our view on priorities is also found in Imai et al. [13], in which the authors state that any kind of weight/priority can be attached to individual vessels: After all, this formulation has the advantage that any kind of weight can be attached to individual vessels. For instance, when a ship must be handled quickly for a certain reason such as an emergency, high priority may be realized in the resulting solution by adding a high value to it in the formulation. In Imai et al. [16] very large containervessels are given priority to guarantee that they will be served upon arrival at the right type of berth. The importance of the preferred berth aspect is also illustrated by

Moorthy and Teo [24]: the preferred berth is used as a key input to yard storage, personnel and equipment deployment planning and a framework is proposed to address the preferred berth design problem. Lokuge and Alahakoon [23] force the waiting time to zero for high-priority vessels, guaranteeing immediate service. Priorities are clearly an important aspect of the handling time, but not the only one: e.g. also berthing places and the number of cranes assigned to service the vessel influence its handling time.

Berthing places are assigned sections of the quay. In literature, the quay is modeled in different ways. Many models (e.g. Imai et al. [12]) assume the quay to consist of a discrete set of berthing locations. These so-called Discrete Berth Allocation problems (BAPD) often result in underutilized berthing capacity because the berth lengths do not correspond exactly to vessel lengths. The Continuous Berth Allocation Problem (BAPC) models the quay as a continuous line segment (Lim [21]). Li et al. [20] formulate a BAPC solution approach with and without fixed vessel positions. In both Park and Kim [26] and Kim and Moon [17], the BAPC is extended with handling priorities and preferred berths for vessels. Priorities are imposed by adding penalty costs for violating a vessels arrival and departure times to the original objective function of minimizing container handling cost. Guan and Cheung [9] propose a BAPC with handling times proportional to vessel size. It is our belief that this assumption is not justified because handling time is more related to the number of containers to be handled than the vessels overall capacity. Operational experiences show that small vessels sometimes have more containers to be handled than larger ones. Imai et al. [15] develop a heuristic for a BAPC with the handling time being dependent of the berthing position. They assume that the handling time is defined by the vessels quay location and its container storage location on the yard. If a sufficient number of prime movers (e.g. straddle carriers) are employed to haul containers between the vessel and the storage location on the yard, there should not be an interruption or delay of the quay crane cycle. We therefore consider the handling time to be dependent on the productivity of the quay cranes (which depends on several factors of which the number of straddle carriers is one) and the number of quay cranes used to service a vessel.

Only a limited number of papers have been published on the Crane Allocation Problem (CAP) and its combination with the BAP. Both Daganzo [5] and Peterkofsky [27] propose quay crane assignments using vessel sections (bays). They consider a static CAP and minimize total weighted completion times. Park and Kim [26] suggest a two-phase solution procedure: the first phase determines the berthing position and berthing time of each vessel as well as the number of cranes assigned to each vessel at each time segment while a detailed schedule for each crane is constructed in the second phase. They develop an integer programming model assuming discrete berths and time intervals. Liu et al. [22] assume the BAP to be solved upfront and use a two-phase approach for the quay crane scheduling. In the first phase, they minimize each vessel's processing time for various possible numbers of quay cranes (QC) taking into account QC interaction constraints. During the second phase, QCs are assigned to vessels to minimize the tardiness of all vessels. Imai et al. [16] introduce a formulation for minimizing total service time in the simultaneous berth and crane allocation problem by extending the model of Imai et al. [12] with a decision variable for vessel-berth-order assignment and with additional CAP constraints solving it with a genetic algorithm.

As mentioned by Park and Kim [26], the integrated BAP-CAP can be split up in two phases: the first phase, for which they use a sub-gradient optimization method, is the assignment of quay cranes to vessels and of vessels to berths. The second phase consists of a detailed schedule for each quay crane based on the solution from the first phase. Park and Kim propose a dynamic programming technique for this second phase. We consider the second phase described by Park and Kim [26] to be an operational decision that is taken on another level. This assumption is based on the fact that there is more to be considered than crane movements alone (e.g. hatch types, workload per bay of a vessel) in determining the detailed schedule of a quay crane. Indeed, to make the operational planning of detailed crane assignments the loading plans first need to be finalized as well. As these plans are only finalized a day before arrival at the most (due to the lateness of availability of the loading figures), it is generally impossible to generate detailed crane assignments days beforehand. We therefore focus on the first phase only. A genetic algorithm is proposed by Tavakkoli-Moghaddam et al. [30] for this type of quay crane scheduling problem.

Liu et al. [22] solve the CAP for a given optimized solution of the BAP. The CAP determines the number of cranes deployed to a vessel and hence the vessels handling time and overall berth occupancy. The latter is a crucial input to the BAP. As such, we feel that the BAP and CAP should be jointly optimized. On the other hand, the detailed workload assignment of each crane can safely be uncoupled from the BAP because this decision is taken on another level and at a different moment in time.

Imai et al. [10] remark that quay cranes mounted on common rails limit the transferring capabilities of the cranes from vessel to vessel: cranes cannot be transferred from an origin berth to a destination berth during loading and discharging. We want to relax this constraint because the moving cranes can take over each others workload when moving from one berth to another, so cranes can always be shifted between vessels at any time. To avoid time loss due to the inoperability of the cranes whilst moving from one berth to another (thus improving crane productivity), they are ideally shifted during scheduled breaks. This crane movement during the breaks can then be carried out e.g. by the electricians. Further, Imai et al. [10] assume that work on a vessel can only be started if all cranes needed to handle the workload are available. Again, we believe that this assumption is too restrictive, so we relax it in our model, making it possible to start servicing a vessel as soon as a crane is available.

5.4 A rich model for the BAP and CAP

This section presents a richer model for the integrated BAP-CAP, incorporating additional real-life features as identified above. In the development of the model, we assume that all vessels approaching the berth need to be scheduled at minimum cost. We assume moreover that draft restrictions and quay crane restrictions (reach and height) are not an issue since all vessels have been assigned to a compatible preferred berthing area on a terminal with movable quay cranes. A final assumption is that once a vessel is moored at the quay, it will stay in that position until the end of its service.

The best place to moor a vessel is as close as possible to the dedicated storage location of the containers to be loaded and unloaded on the yard. If the vessel is moored too far from its preferred berthing location, the prime movers (e.g. straddle carriers) that bring the containers from the yard to the quay cranes or vice versa have to cover too much distance. Deviations from the preferred berthing location are therefore to be minimized.

To be able to formulate the problem as a mixed integer linear program, the time horizon of the model is discretized into intervals for the assignment of the quay cranes. The length of these intervals is a user-defined parameter as illustrated in the computational experiments of Section 5.5.

The following notations are used for the parameters in the mathematical model:

S	the set of vessels to be planned (indexed by i)
$pbl(i)$	preferred berthing location of vessel i along the quay
$l(i)$	length of vessel i, including space in front of and behind the vessel for safe mooring
$wl(i)$	workload of vessel i (number of containers)
$maxCr(i)$	the maximum number of cranes that can be assigned to vessel i at any time
T	number of periods in the time horizon (indexed by t)
$arr(i)$	period in which vessel i arrives
$mst(i)$	the minimum service time of vessel i
$epd(i)$	the earliest possible departure of vessel i, i.e. $arr(i) + mst(i)$.
$ddl(i)$	the deadline of vessel i, i.e. the period during which its service must be finalized
$S(i)$	the subset of vessels that can be along the quay together with vessel i, i.e. vessels whose arrival is before i but whose deadline is not, and vessels that arrive after i but before its deadline
$\pi(i)$	handling priority of vessel i
$\alpha(i)$	positioning priority of vessel i
ε	penalty cost for moving cranes
L	total available quay length
P	crane productivity (cranes loaded/unloaded per period)
$totalCranes$	the total number of cranes available at the terminal

In the model we assume that, even when the terminal is highly congested, every vessel i can still be feasibly and fully serviced within two times its minimum service time $mst(i)$, or within 24 hours if the minimum service time is less than 12 hours. In other words, for each vessel i, we define its deadline as follows: $ddl(i) = arr(i) + max(24h; 2 * mst(i)) - 1$. Period $ddl(i)$ is then the last period in which vessel i can receive service. If it turns out that offering a service time window of two times the minimum service time, with a minimum of 24 hours, is not enough for serving all vessels, i.e. if there are in-feasibilities when solving the model, this deadline can be extended and the model resolved until a feasible solution is found. However, these in-feasibilities also serve as a warning signal to the container terminal operator, because they indicate that at certain moments, the terminal is really overloaded, and that the assignment of vessels to that terminal, or the schedule of shipping lines visiting the terminal has to be revised in order to spread the load more evenly over time.

In the mixed integer linear programming (MILP) model presented in Figure 5.1, the following decision variables are used:

$busy_{it}$	binary variable stating whether vessel i is being served in period $t = arr(i)..ddl(i)$
$nrCr_{it}$	integer variable representing the number of cranes assigned to vessel i in period $t = arr(i)..ddl(i)$
$change_{it}$	number of additional cranes assigned to vessel i in period t compared to period $t-1$: $(t = arr(i) + 1..ddl(i))$
rw_it	remaining workload of vessel $i$in period $t = arr(i)..ddl(i)$
pl_i	planned position of vessel i
dl_i	deviation to the left of the preferred berthing location for vessel i
dr_i	deviation to the right of the preferred berthing location for vessel i
y_{ij}	binary variable stating whether vessel i is positioned entirely in front of vessel j

The objective function is a minimization of three components. The first component is related to the handling time of the vessels and includes a penalty term for vessel handling delays. When a vessel arrives and is being handled immediately by the maximum possible number of cranes, we can calculate how many containers will be handled (discharged and/or loaded) during the consecutive time intervals that the vessel remains alongside the quay until it is completely serviced. The workload that remains at the beginning of each of the consecutive time intervals when assigning the maximum number of allowable cranes is the ideal remaining workload, $iwl(i,t)$, and the minimum service time $mst(i)$ is the number of periods after which $iwl(i,t)$ drops to zero. The actual assignment of cranes results in a remaining workload for each consecutive time interval, $rwit$, that is either equal to the ideal workload $iwl(i,t)$ or above it (deviations from the ideal remaining workload are indicated in black in Figure 5.2). At period $epd(i) = arr(i) + mst(i)$, when the ideal workload drops to zero, the actual workload may still be nonzero, because of limited quay length and quay crane availability. This indicates that the

$$\text{Minimize} \sum_{i \in S}\left(\left(\pi(i)\cdot \sum_{t=epd(i)-1}^{ddl(i)} rw_{it}\right)+\alpha(i)\cdot wl(i)\cdot(dl_i+dr_i)+\sum_{t=arr(i)+1}^{ddl(i)} \varepsilon\cdot change_{it}\right)$$

s.t.

$$\begin{array}{lll}
busy_{it} \geq \dfrac{rw_{it}}{wl(i)} + busy_{i,t-1} - 1 & \forall i \in S, \forall t = arr(i)+1..ddl(i) & (1)\\
rw_{i,arr(i)} = wl(i) & \forall i \in S & (2)\\
rw_{it} \geq rw_{i,t-1} - nrCr_{i,t-1}\cdot P & \forall i \in S, \forall t = arr(i)+1..ddl(i) & (3)\\
nrCr_{i,ddl(i)} \geq rw_{i,ddl(i)}/P & \forall i \in S & (4)\\
nrCr_{it} \leq maxCr(i)\cdot busy_{it} & \forall i \in S, \forall t = arr(i)..ddl(i) & (5)\\
\sum_{i\in S: arr(i)\leq t \leq ddl(i)} nrCr_{it} \leq totalCranes & \forall t = 1..T & (6)\\
change_{it} \geq nrCr_{it} - nrCr_{i,t-1} & \forall i \in S, \forall t = arr(i)+1..ddl(i) & (7)\\
pl_i = pbl(i) - dl_i + dr_i & \forall i \in S & (8)\\
pl_i + l(i) \leq L & \forall i \in S & (9)\\
pl_i + l(i) - L(1-y_{ij}) \leq pl_j & \forall i \in S, \forall j \in S(i) & (10)\\
busy_{it} + busy_{jt} \leq 1 + y_{ij} + y_{ji} & \begin{cases}\forall i \in S, \forall j \in S(i),\\ \forall t = \max(arr(i),arr(j))..\min(ddl(i),ddl(j))\end{cases} & (11)\\
dl_i, dr_i, pl_i, rw_{it} \geq 0 & \forall i \in S, \forall t = arr(i)..ddl(i) & (12)\\
nrCr_{it}, change_{it} \in \{0,1,\ldots,maxCr(i)\} & \forall i \in S, \forall t = arr(i)..ddl(i) & (13)\\
busy_{it} \in \{0,1\} & \forall i \in S, \forall t = arr(i)..ddl(i) & (14)\\
y_{ij} \in \{0,1\} & \forall i \in S, \forall j \in S(i) & (15)
\end{array}$$

Figure 5.1: The mathematical model

vessels service was delayed. Since we want to minimize vessel delays, the summation of $rwit$ from period $epd(i)$ onwards is part of the objective function. The more the handling of a vessel deviates from its ideal handling situation, the more it will be penalized. This handling time aspect is multiplied with each vessels priority (i). In the case that two vessels compete for the same location along the quay at the same moment, the priorities will ensure that the more important vessel will be assigned relatively more cranes to finish its handling earlier.

Because time is divided into discrete periods in the model, the service of a vessel can be slightly delayed without causing a penalty term. Consider the following example. A vessel has a workload of 500 containers and can be serviced by at most 3 cranes at the same time. The cranes have a productivity of 50 containers per period, so the minimum service time for the vessel is 4 periods. In the ideal situation, three cranes are assigned to this vessel upon its arrival, such that at the beginning of the fourth period, the remaining workload has dropped to 50 containers, and the vessel can leave during this fourth period. Now, assume that only two cranes are assigned to this vessel during the first two periods, and three cranes are assigned to the vessel during the third and fourth period. Then, the

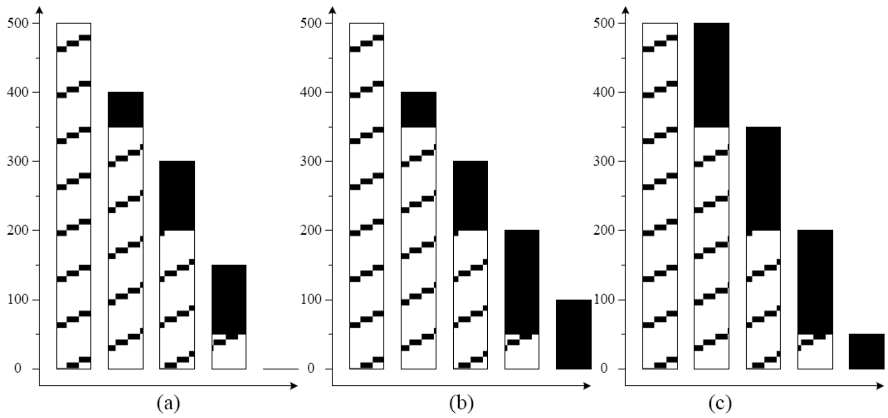


Figure 5.2: The ideal workload $iwl(i, t)$ (shaded) and actual workload $rwit$ (shaded + black)

remaining workload drops to zero right at the end of the fourth period. There is an actual delay then, because service only finishes at the end of the fourth period, instead of early in the fourth period as in the ideal situation. However, this delay is not penalized, because it is still true that the workload has dropped to zero at the beginning of the fifth period. To also penalize such a hidden delay, the term $rwit$ for period $epd(i)1$ is also added to the objective function. By doing so, the delay that was hidden because it could be made up during the last period within the minimum service time, is now no longer hidden. Figure 5.2a illustrates this hidden delay. In Figure 5.2b, there is a hidden and an actual delay because only two cranes are assigned, and Figure 5.2c represents a delayed start of service.

The second component of the objective function is related to the berthing position of the vessels and includes a penalty term for deviating from a vessels preferred berthing location. dli and dri represent the deviation to the left and to the right from this preferred berthing location. This deviation is first multiplied by the workload $wl(i)$ of the respective vessel, because the prime movers need to travel this extra distance for each of the containers being loaded or discharged. This way, the berthing location of a small vessel with many containers to be handled is more important than a long vessel with only a few containers to be handled. The second objective function component is further weighted with $\alpha(i)$. This weighting factor reflects the importance of the preferred berthing location compared to the handling times of the vessels. This factor can further be used to introduce several operational aspects into the objective function, e.g.: (a) it may be more important for vessels to be planned on their preferred location than barges, (b) when two vessels need to be handled at the same time but they have the same preferred berthing location it is possible to assign one of the vessels closer to its preferred berthing location than the other.

The third component of the objective function penalizes changes in the number of cranes assigned to a vessel during its service to reflect the (opportunity) cost of having to move cranes along the quay. This penalty cost is only charged when there is an increase in the number of cranes assigned to a vessel. If the number of cranes decreases, this is either because the vessels service is finished (which should not be penalized), or because the cranes move to another vessel (resulting in a penalty being charged for that vessel). Constraint (1) of the model in Figure 5.1 defines the binary variable $busy_{it}$ and links it with $rw_{it)}$: once started, handling continues for as long as the remaining workload has not reached zero. Constraint (2) defines the initial workloads while Constraint (3) defines the remaining workload on vessel i for the following intervals, being the remaining workload of the previous interval minus the number of cranes assigned to the vessel times their productivity. This constraint is implemented as an inequality to make sure that rw_{it} does not drop below zero. Constraint (4) imposes the deadline on a vessels handling. Constraint (5) ensures that no more than the maximum number of cranes is assigned to a vessel at any time, while Constraint (6) enforces that, in any period, no more than the maximum number of cranes present along the quay can be deployed. Constraint (7) defines the $change_{it}$ variable as the additional number of cranes allocated to a vessel from one period to the next. If the number of cranes allocated to a vessel does not change, or if it decreases, the $change_{it}$ variable will be zero. The position that a vessel takes alongside the quay is defined in Constraint (8) as the deviation to the left or right from its preferred berthing location. Constraint (9) ensures that all vessels are positioned within the available quay length. Constraints (10) and (11) avoid overlaps by ensuring that no two vessels are being serviced at the same place at the same time. These constraints are of course only defined for vessels that can be along the quay together, and only for the periods during which this can occur. Non-negativity is enforced in Constraints (12), Constraint (13) defines the integer variables, and Constraints (14) and (15) define the binary variables.

Illustrative example:
As an illustration of the model, consider the following example in which both quay length and the available cranes are scarce resources. The example includes five vessels, whose data is given in Table 5.1. The quay is 1,200m long, and 15 cranes are available with a productivity of 30 containers per hour. Periods of 1 hour are considered. The parameter ε is set at 150. The solution for this instance is shown in Figure 5.3. In this example, all three types of penalty costs are incurred (see Table 5.2). First, vessels VS2 and VS3 incur a penalty for delayed service because in periods 7 and 8 they do not receive their maximum number of cranes, such that their service finishes one period late. E.g., for vessel VS2, the ideal workload is 150 containers in period 10 and 0 in period 11 (the earliest possible departure period). Because only three cranes are deployed in periods 7 and 8, the service of VS2 is lagging behind two times 60 containers and the actual remaining workload for periods 10 and 11 is 270 and 120 containers, respectively. This leads to a total

i	*l(i)*	*arr(i)*	*wl(i)*	*π(i)*	*pbl(i)*	*maxCr(i)*	*α(i)*	*mst(i)*	*epd(i)*	*ddl(i)*
VS1	300	1	720	6	0	3	0.0004	8	9	24
VS2	300	1	1500	9	500	5	0.0001	10	11	24
VS3	300	2	1800	10	200	5	0.0001	12	14	25
VS4	300	3	200	5	600	2	0.0004	4	7	26
VS5	300	4	1500	9	400	5	0.0001	10	14	27

Table 5.1: Vessel data for the illustrative example

penalty cost of 9(270-150) + 9120 = 2160. Vessel VS5 also incurs a delay penalty, because its service starts late since it has to wait until vessel VS4 leaves.

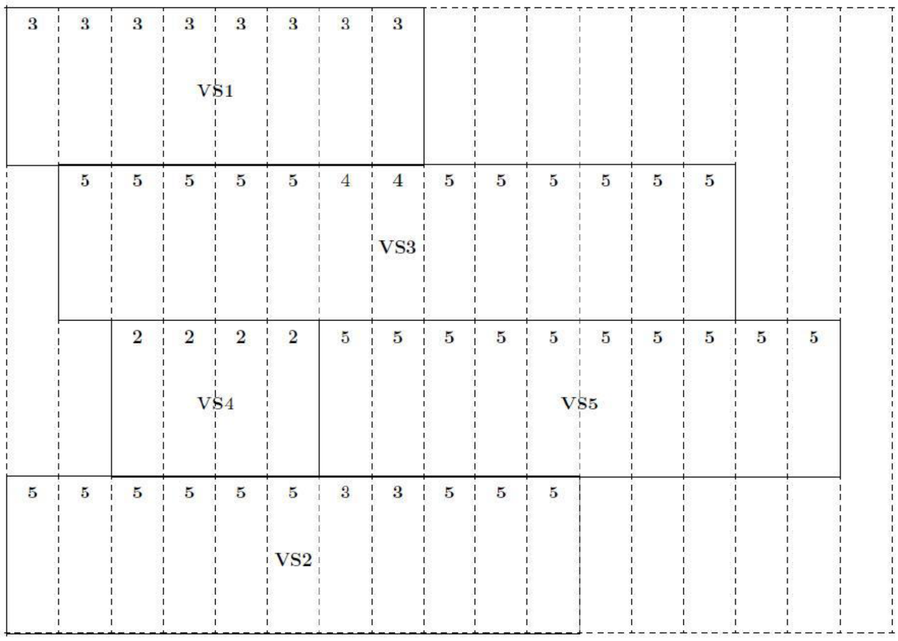


Figure 5.3: Solution of the illustrative example

The second type of penalty is for deviation from the preferred berthing location. Only vessels VS1 and VS4 are at their preferred location. For the other vessels, it is only possible to berth them at their preferred location by waiting for other vessels to leave. This would however lead to very high delay penalty costs. The model determines the best trade-off between the different penalty costs and it turns out that it is cheaper to berth those vessels as soon as possible, albeit away from their preferred berth.

To limit the crane changeovers during the service of a vessel, an increase of the number of quay cranes is penalized. This happens in period 9 for vessels VS2 and

i	$pl(i)$	$Time$	$Distance$	$Crane$
VS1	0	0	0	0
VS2	900	2160	60	300
VS3	300	1200	18	150
VS4	600	0	0	0
VS5	600	12150	30	0

Table 5.2: Penalty costs in the illustrative example

VS3.

A rolling horizon framework:
The model presented above is embedded in a rolling horizon framework. When solving the model at a certain moment (e.g. at 6 am), a time horizon of a given number of periods is considered (e.g. the next 48 hours). The solution of the model is then implemented for the first next period (e.g. the 2 hours between 6 and 8 am). By the end of that period (8 am), the same model is reused (with updated information, i.e. with decremented values for the $arr(i)$, $mst(i)$, $epd(i)$ and $ddl(i)$ parameters) and a new plan is generated for the same number of periods as before (i.e. the scheduling horizon is shifted one period ahead in time).

Embedding the model in a rolling horizon framework results in some complications for the implementation of the model. First, at the moment of (re)planning, there will be vessels along the quay whose service has already started. For these vessels, the position cannot be changed anymore, while the allocation of cranes to this vessel for the next periods still can. In terms of the model, the values of the variables $busyi1$, pli, dli and dri are fixed beforehand for these vessels (Constraints (1) and (8) below) and the workload $wl(i)$ has to be updated. Also, for these vessels, the change in number of cranes compared to the previous period has to be monitored (and penalized) in the initial period of the current planning horizon (Constraint (7) below).

The second complication appears at the end of the time horizon, where vessels appear whose deadline is beyond the planning horizon. For these vessels, the condition that handling must be finished by the deadline (Constraint (4)) is omitted.

For vessels whose earliest possible departure $epd(i)$ is beyond the time horizon T, no penalty terms for service delays are included in the original objective function. This means that the model can delay service of such a vessel beyond the time horizon at no additional cost, leaving more room (quay space and cranes) for planning the other vessels at minimal costs. To force the model to also take into account vessels whose earliest departure is beyond the scheduling horizon, we also penalize the delays of these vessels by adding the term $(i)rwiT$ to the objective function. Without this additional penalty term, these vessels would never receive service in the current scheduling horizon.

After the adjustments outlined above, the mathematical model for the integrated berth allocation and quay crane assignment problem is as presented in Figure 5.4.

The additional parameters in this model are:

$A(i)$	arrival of vessel i, given by $max(arr(i);1)$
$D(i)$	deadline for vessel i, given by $min(ddl(i);T)$
S_0	vessels whose service has already started
S_1	vessels whose deadline is beyond the time horizon, i.e. $ddl(i) > T$
S_2	vessels whose earliest departure is beyond the time horizon, i.e. $edp(i) > T$
$pos(i)$	position of vessel i along the quay (only for vessels in S_0)
$nrCr(i,0)$	number of cranes assigned to vessel i in the previous period (only for vessels in S_0)

$$\text{Minimize} \sum_{i\in S}\left(\left(\pi(i)\cdot\sum_{t=epd(i)-1}^{D(i)} rw_{it}\right)+\alpha(i)\cdot wl(i)\cdot(dl_i+dr_i)+\sum_{t=A(i)+1}^{D(i)}\varepsilon\cdot change_{it}\right)+\sum_{i\in S_2}\pi(i)\cdot rw_{iT}+\sum_{i\in S_0}\varepsilon\cdot change_{i1}$$

s.t.

$$\begin{array}{lll}
busy_{i1}=1 & \forall i\in S_0 & (1')\\
busy_{it}\geq \dfrac{rw_{it}}{wl(i)}+busy_{i,t-1}-1 & \forall i\in S, \forall t=A(i)+1..D(i) & (1)\\
rw_{i,A(i)}=wl(i) & \forall i\in S & (2)\\
rw_{it}\geq rw_{i,t-1}-nrCr_{i,t-1}\cdot P & \forall i\in S, \forall t=A(i)+1..D(i) & (3)\\
nrCr_{i,D(i)}\geq rw_{i,D(i)}/P & \forall i\in S\setminus S_1 & (4)\\
nrCr_{it}\leq maxCr(i)\cdot busy_{it} & \forall i\in S, \forall t=A(i)..D(i) & (5)\\
\sum_{i\in S:A(i)\leq t\leq D(i)} nrCr_{it}\leq totalCranes & \forall t=1..T & (6)\\
change_{it}\geq nrCr_{it}-nrCr_{i,t-1} & \forall i\in S, \forall t=A(i)+1..D(i) & (7)\\
change_{i1}\geq nrCr_{i1}-nrCr(i,0) & \forall i\in S_0 & (7')\\
pl_i=pbl(i)-dl_i+dr_i & \forall i\in S & (8')\\
pl_i=pos(i) & \forall i\in S_0 & (8)\\
pl_i+l(i)\leq L & \forall i\in S & (9)\\
pl_i+l(i)-L(1-y_{ij})\leq pl_j & \forall i\in S, \forall j\in S(i) & (10)\\
busy_{it}+busy_{jt}\leq 1+y_{ij}+y_{ji} & \begin{cases}\forall i\in S, \forall j\in S(i),\\ \forall t=\max(A(i),A(j))..\min(D(i),D(j))\end{cases} & (11)\\
dl_i, dr_i, pl_i, rw_{it}\geq 0 & \forall i\in S, \forall t=A(i)..D(i) & (12)\\
nrCr_{it}, change_{it}\in\{0,1,\ldots,maxCr(i)\} & \forall i\in S, \forall t=A(i)..D(i) & (13)\\
busy_{it}\in\{0,1\} & \forall i\in S, \forall t=A(i)..D(i) & (14)\\
y_{ij}\in\{0,1\} & \forall i\in S, \forall j\in S(i) & (15)
\end{array}$$

Figure 5.4: The revised mathematical model for the rolling horizon framework

5.5 Computational results

To validate the model and the rolling horizon framework presented above, a three-month data set was obtained on container handling operations from two important container terminals in the port of Antwerp. Table 5.3 summarizes the data from the two container terminals that have a quay length of respectively 2,000m and 1,500m, with 18 and 8 quay cranes. The number of containers to be handled by a vessel consists of containers to be discharged and containers to be loaded. At both terminals, both ocean going vessels and inland barges are serviced. The ratio of number of vessels/ barges and their respective workloads are 367/1,823 and 582,809/86,694 for the first terminal, and 249/1,951 and 169,387/70,393 for the second terminal. Both data sets thus include a large number of barges with a limited workload: at least 46% of the barges have a workload of less than 25 containers. Because a barges workload is usually smaller than a quay cranes handling capacity for a given time interval, the barges need to be aggregated to avoid barges with a small workload claiming a quay cranes productivity for an entire time interval. Therefore, all barges for a given interval are first consolidated into a minimal number of aggregated barges whose cumulative workload is not more than a quay cranes productivity. As the lengths of the barges were not recorded, a default length of 170 meters is assumed for all barges.

For both terminals, a quay crane productivity of 35 containers per hour is assumed. The actual quay crane productivity is higher than that, but we lowered it to 35 containers per hour to take into account the fact that quay cranes lose time every now and then to move from one vessel to the next (remember that quay crane scheduling is not included in the model).

At both terminals, deviations from the preferred berthing location are considered to be more important for vessels than for barges. In fact, for the barges, no preferred berthing location was specified. This is reflected in setting the value of $\alpha(i)$ to 0.001 for vessels and 0.00004 for barges. These small values guarantee that the place deviation penalties in the objective function remain small relative to handling time delay penalties. A sensitivity analysis is presented below to assess the influence of the $\alpha(i)$ values on the generated solutions.

The model was implemented in C++ using ILOG Concert Technology and solved using ILOG CPLEX 11 on an Intel T7300 2.00GHz processor with 2 GB of RAM.

5.5.1 Base case analysis

In a first experiment, we want to find out what the appropriate time interval and time horizon are for both datasets. Therefore, we ran the model over both entire three month data sets with different time intervals (2, 4 and 8 hours) and with different time horizons (24, 48 and 72 hours).

Figure 5.5 illustrates a planning with 2-hour time intervals and a time horizon

		#Vessels Dataset A	#Vessels Dataset B
Vessel length	-100	0	41
	101-150	8	100
	151-200	72	23
	201-250	80	29
	251-300	113	53
	301+	94	3
Workload vessels	-500	60	154
	501-1000	65	16
	1001-1500	97	24
	1501-2000	44	38
	2001-2500	32	14
	2500+	69	3
Workload barges	-25	847	1005
	26-50	427	469
	51-100	332	351
	101-150	105	96
	151-200	56	24
	200+	56	6

Table 5.3: Data summary

of 48 hours, in which B denotes (aggregated) barges and V vessels with the associated number of quay cranes per time interval of the service. The horizontal axis represents the time intervals during the planning horizon, the available quay length is shown along the vertical axis. It can be seen in Figure 5.5 that the number of cranes allocated to the vessels in the final period is zero. Assigning cranes in this final period would reduce the remaining workload for the next period, but that next period is beyond the current time horizon. Of course, as the rolling horizon progresses in the next iteration, that next period comes within scope and cranes will be assigned.

Table 5.4 illustrates the resulting computational complexity of the different configurations for both datasets by displaying the average gap, CPU time, number of binary variables and number of constraints per iteration of the model. As expected, this complexity increases (nonlinearly) for smaller intervals and a longer time horizon. The fact that the number of binary variables more than doubles when halving the time intervals (e.g. from 278 over 599 to 1373 for dataset A and a 48h time horizon) may seem strange at first, but this is due to the aggregation of barges. With longer time intervals, less aggregated barges remain.

The most important observation from Table 5.4 is that the average computa-

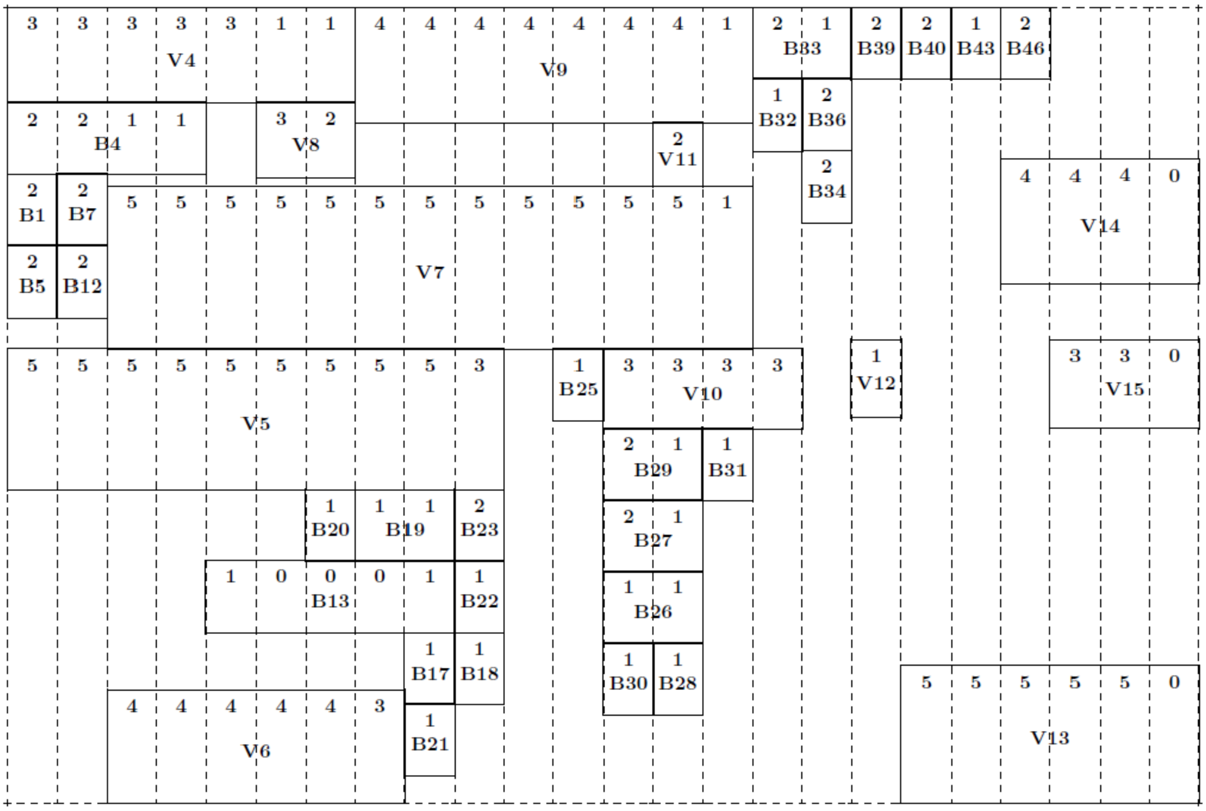


Figure 5.5: Illustrative planning, indicating number of cranes assigned to vessels per interval

tion times are very limited, even for the most complex 72-hour time horizon with 2-hour time intervals. We must mention that individual computation times have been limited because we gradually increase the allowed MIP gap during computation. After one minute the CPLEX standard relative gap of 0.01% is increased to 0.01%. After two and three minutes the gap is increased to 0.1% and 1% respectively. The maximal resulting MIP gap in these experiments, however, is only 0.4% and the longest computation time is only 3 minutes. Because the computation times are that small, the model can be used in an operational environment which requires frequent replanning as updated information on future arrival times becomes available.

When comparing the solutions with a time horizon of 24, 48 and 72 hours (see Table 5.5), it turns out that these are in general worse for the 24-hour horizon, but almost identical for 48 and 72 hours. Of course, when planning the service of vessels arriving in the current period, it makes sense to take into account vessels arriving in subsequent periods. Apparently, looking ahead for two days is

	Dataset A								
Interval		2h			4h			8h	
Horizon	24h	48h	72h	24h	48h	72h	24h	48h	72h
Avg gap	0.0051	0.0086	0.0169	0.0025	0.0041	0.0060	0.0015	0.0037	0.0079
Avg CPU (s)	2.47	7.71	18.60	0.11	1.72	4.86	0.02	1.48	5.49
Avg nrBin	614	1373	2127	276	599	920	128	278	424
Avg nrCstr	1911	5102	8286	670	1655	2638	261	620	976
Max gap	0.1487	0.1653	0.3986	0.0100	0.0634	0.0484	0.0100	0.1449	0.2640
Max CPU (s)	180.05	180.03	180.17	8.88	120.02	120.02	1.89	180.02	180.03
Max nrBin	1108	2188	3064	527	958	1389	269	483	627
Max nrCstr	3795	8385	12153	1385	2730	3989	562	1100	1436
	Dataset B								
Interval		2h			4h			8h	
Horizon	24h	48h	72h	24h	48h	72h	24h	48h	72h
Avg gap	0.0027	0.0038	0.0043	0.0012	0.0016	0.0017	0.0008	0.0014	0.0017
Avg CPU (s)	0.05	0.24	0.47	0.01	0.03	0.05	0.00	0.01	0.02
Avg nrBin	390	914	1428	143	328	508	60	134	206
Avg nrCstr	1161	3323	5442	345	906	1457	126	309	487
Max gap	0.0100	0.0100	0.0100	0.0098	0.0100	0.0099	0.0091	0.0099	0.0100
Max CPU (s)	1.09	5.58	8.30	0.06	0.11	0.17	0.02	0.05	0.05
Max nrBin	1071	1929	2590	312	632	809	162	314	400
Max nrCstr	3367	7188	10169	754	1698	2350	311	667	907

Table 5.4: Configuration complexity

sufficient. Looking further ahead (e.g. for three days) only increases computational complexity without improving solution quality. Therefore, we only consider a 48-hour planning horizon for the remainder of our experiments.

	Dataset A								
Interval		2h			4h			8h	
Horizon	24h	48h	72h	24h	48h	72h	24h	48h	72h
Time	72434	72164	72142	96544	95900	95900	178392	152544	152544
Place	37970	33101	33153	41382	34477	34491	47795	39317	38450
Crane	4165	4200	4235	5180	5320	5250	2940	4200	4200
Total	114569	109465	109530	143106	135697	135641	229127	196061	195194
	Dataset B								
Interval		2h			4h			8h	
Horizon	24h	48h	72h	24h	48h	72h	24h	48h	72h
Time	52656	52656	52656	78732	78664	78732	86712	86712	86712
Place	929	927	929	540	545	526	1194	1094	1144
Crane	4165	4130	4165	2100	2310	2170	0	0	0
Total	57750	57713	57750	81372	81519	81428	87906	87806	87856

Table 5.5: Objective function components

The total cost of the schedules is very different for varying time intervals. However, the structure of the actual solutions (i.e. berthing positions and quay crane assignments) are almost identical. The cost difference results from the following impact of time interval size on total cost. Because berthing locations are assigned to indivisible time intervals, locations are kept unavailable for other vessels even when the fully serviced vessels would in reality have already departed within the time interval. As such, with larger time intervals, the model considers berthing locations occupied longer than actually needed. As an example, consider a vessel requiring 6 hours of service. When a 2-hour time interval is used, the berthing location becomes available after 3 time intervals. If a 4 hour time interval would be used, 2 consecutive time intervals would be assigned to the service, although the ship service is actually finished after 1.5 time periods. The remaining 0.5 time period, the berthing location is unavailable for other vessels and could force these to be serviced later and/or at a less favorable berthing location, thus increasing both time and distance related costs. Based on the computational experiments ,we consider 2-hour time intervals and a time horizon of 48 hours to be the most appropriate configuration for generating berth allocation and quay crane assignment schedules for the terminals under consideration.

5.5.2 Sensitivity analysis

To further validate the model and illustrate its potential as decision support tool, three types of sensitivity analysis are performed for both datasets. In this set of experiments, we will vary the number of available cranes, the available quay length and the place penalty parameters $\alpha(i)$.

The first sensitivity analysis is done by changing the number of available quay cranes and rerunning the model over the entire time horizon. For dataset A, the number of cranes is varied between 16 and 20, for dataset B between 6 and 10. Results are shown in Table 5.6 and plotted in Figure 5.6.

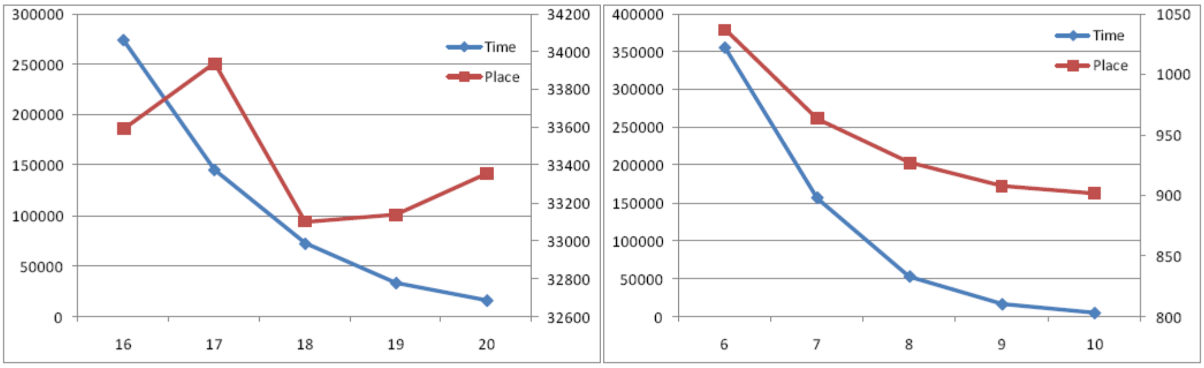


Figure 5.6: Time and place penalties for varying number of cranes

We can see that increasing the number of quay cranes significantly reduces overall costs. This is mainly because additional quay cranes allow vessels to be

Dataset A					
Cranes	CPU (s)	Time	Place	Crane	Total
16	20.28	274092	33595	19460	327147
17	6.66	145416	33939	12215	191570
18	7.65	72164	33101	4200	109465
19	3.73	33280	33136	2205	68621
20	0.93	15876	33355	1190	50421
Dataset B					
Cranes	CPU (s)	Time	Place	Crane	Total
6	3.32	354952	1037	15820	371809
7	3.53	156794	964	7140	164898
8	0.25	52656	927	4130	57713
9	0.14	15900	907	2205	19012
10	0.13	4356	902	350	5608

Table 5.6: Varying the number of cranes

serviced quicker (avoiding time penalty costs) which in turn frees up quay space possibly enabling vessels to berth closer to their preferred berthing position (reducing place penalty costs). As more quay cranes are available for a given number of vessels to be serviced, quay cranes become less a binding resource (avoiding crane penalties). Decreasing the number of quay cranes results in longer periods of congestion at the terminal and increases the overall cost of the schedule. In both cases, the model makes the best possible trade-off between these three cost components as illustrated in Figure 5.6.

Reducing the number of cranes not only increases total costs, it also increases the computational complexity and hence the average CPU time per iteration. This is again explained by the longer periods of congestion with less cranes, which means that some vessels remain along the quay somewhat longer and, on average, more vessels have to be planned per iteration. Figure 5.7 shows a sample of the quay crane utilization profile for dataset A for 15 and 20 cranes. Examples of such extended periods of congestion can be observed for the 15 cranes case in periods 1 to 11 and periods 14 to 22.

The second sensitivity analysis considers the available quay length. In this experiment, the quay lengths and preferred berthing locations are rescaled with a factor ranging from 0.8 to 1.2. Results are shown in Table 5.7 and plotted in Figure 5.8.

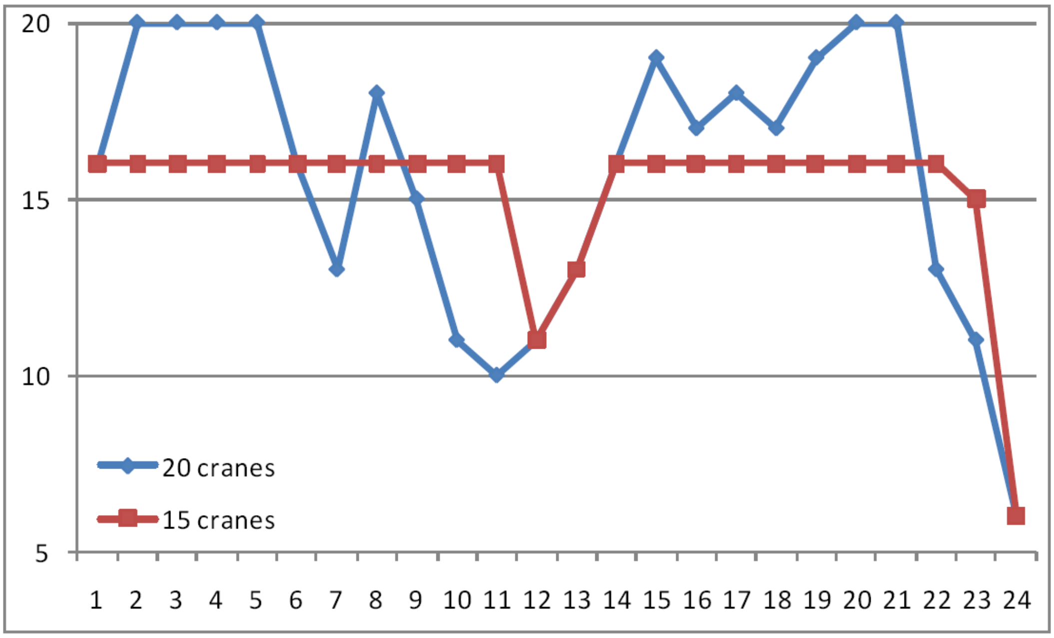


Figure 5.7: Sample quay crane utilization profile (24 periods for Dataset A)

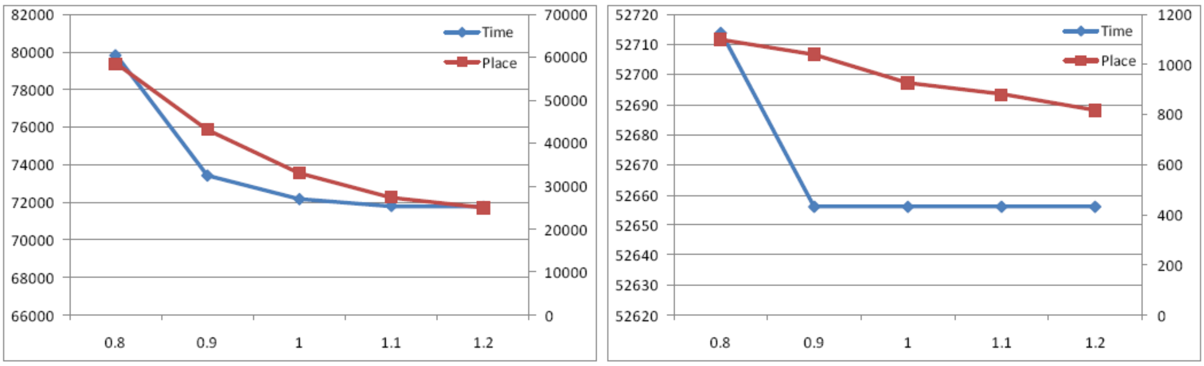


Figure 5.8: Time and place penalties for varying quay lengths

As expected, decreasing the available quay length increases both the place deviation costs and the service delay penalties. When the quay length decreases, it simply means that there is not enough place to serve all vessels during busy periods, let alone serve them near their preferred berthing location. For dataset A, the levels of congestion that are reached for factors 0.8 and 0.9 are that high that computational complexity becomes an issue. The maximum MIP gap reported for a single iteration amounts to 2.67% after 5 minutes of computation time. As

Dataset A					
Factor	CPU (s)	Time	Place	Crane	Total
0.8	22.99	79850	58573	3745	142168
0.9	12.46	73420	43233	4130	120783
1	7.65	72164	33101	4200	109465
1.1	3.39	71800	27352	4305	103457
1.2	3.16	71750	25042	4270	101062
Dataset B					
Factor	CPU (s)	Time	Place	Crane	Total
0.8	1.74	52714	1099	4165	57978
0.9	0.24	52656	1041	4165	57862
1	0.25	52656	927	4130	57713
1.1	0.22	52656	882	4235	57773
1.2	0.19	52656	819	4200	57675

Table 5.7: Varying the quay length

detailed earlier, the gap is gradually increased with increasing computation time, equaling 2 and 3% after respectively 4 and 5 minutes. In other words, the model is well capable of dealing with relatively short periods of congestion. It is only when the congestion persist over the entire planning horizon that computation times become too long to support real-time decision making.

It is striking that the place deviation costs for dataset A are relatively high compared to the situation in dataset B. This indicates that either the allocation of preferred berthing locations to vessels at the container terminal of dataset A or the schedule of shipping lines visiting this terminal (and their expected arrival times at the terminal) should be revised. In the current situation, it occurs far too often that vessels are arriving together for service at the same location, causing these unavoidable penalties. Although the redesign of preferred berthing locations and liner schedules is a complicated matter, both from an operational and from a commercial point of view, experiments like these illustrate that the proposed model is also useful for supporting terminal management decisions at more strategic levels.

The third and final sensitivity analysis experiment considers the weighting factors $\alpha(i)$. These are important management inputs to the model because they determine how time deviations (affecting the shipping lines profitability) and distance deviations (affecting the terminals profitability) are being balanced. For our experiments three different $\alpha(i)$ settings are considered: next to the original values of 0.001 and 0.0004 for vessels and barges, respectively, the relative weight of place deviations is multiplied with a factor of 5 and 10, giving values 0.005/0.002 and 0.01/0.004. The results of these final experiments, reported in Table 5.8 and plotted in Figure 5.9, show that the model successfully manages to make different trade-offs depending on the relative weights of both penalty terms. For dataset A, if place deviation penalties increase, less place deviation occurs at the expense of some additional service delays. For dataset B, the same trade-off remains optimal for the different relative weights. In Figure 5.9, the place penalties (right axis) have been divided by the $\alpha(i)$ values to compare the absolute figures.

Dataset A					
Factor	CPU (s)	Time	Place	Crane	Total
001	7.65	72164	33101	4200	109465
005	6.76	72236	33052	4130	109418
010	6.34	75114	32594	3850	111558
Dataset B					
Factor	CPU (s)	Time	Place	Crane	Total
001	0.25	52656	927	4130	57713
005	0.39	52672	916	4235	57823
010	0.44	52714	914	4060	57688

Table 5.8: Varying the relative weights $\alpha(i)$

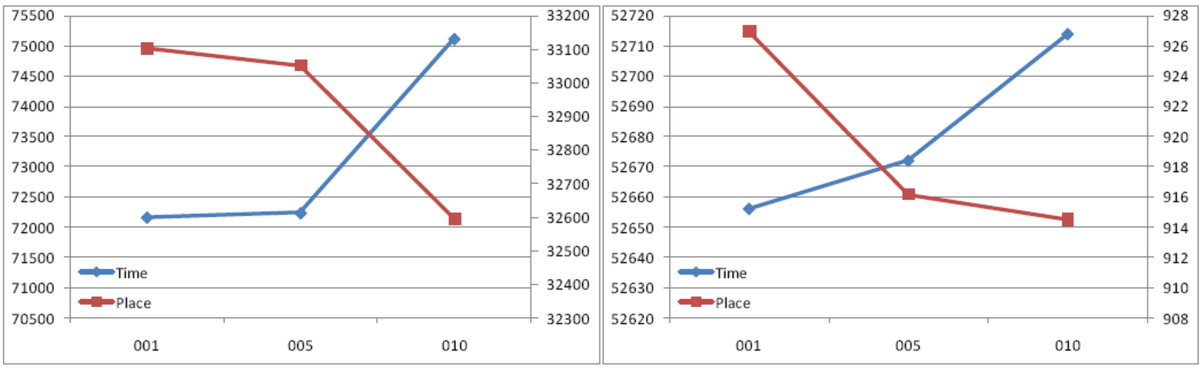


Figure 5.9: Service delay penalties and place deviation for varying relative weights (dataset A)

5.6 Conclusions

In this paper, an enriched model for the Berth and Crane Allocation Problem is presented that takes into account many real-life features often ignored in existing models, such as vessel priorities, preferred berthing locations and handling time considerations. As such, the proposed model can be used as a decision support tool: it automates and optimizes a decision that has to be made several times a day at a modern container terminal, leaving more time for planners to adjust the schedule to handle exceptional situations.

The model is successfully validated on real-life data. Computational results show that the proposed integrated berth and quay crane allocation model provides high quality solutions in reasonable computation times. Moreover, sensitivity analysis on available numbers of quay cranes, quay length and management parameters expressing the trade-offs between cost components illustrate the models capabilities to support managerial decision making.

The model is capable of solving real-life instances in short computation times. Further testing and evaluation on (artificial) datasets that exhibit higher levels of congestion must, however, indicate whether the model remains robust and scalable under extreme conditions or whether heuristic approaches should be preferred.

Other directions for further research include extending the model for handling transshipment operations, loading and unloading containers, and the staff planning for the quay cranes. If transshipments are common, then the preferred berthing position of a vessel will depend on the actual berthing positions of its feeder vessels, imposing handling precedence constraints and making the preferred berthing position a more dynamic issue. By making a distinction between the containers that are to be loaded and those that are to be discharged and by balancing the workload for the quay cranes to minimize labor costs, the model can be further enriched.

Bibliography

[1] AXS Alphaliner *Cellular fleet forecast 2008-2012.* 2008

[2] Bierwirth, C., Meisel, F. *A Survey of Berth Allocation and Quay Crane Scheduling Problems in Container Terminals.* European Journal of Operational Research, DOI 10.1016/j.ejor.2009.05.031

[3] Brown, G.G., Lawphongpanich, S., Thurman, K.P. *Optimizing ship berthing.* Naval Research Logistics 41 (1994) p1..15

[4] Brown, G.G., Cormican, K.J., Lawphongpanich, S., Widdis, D.B. *Optimizing ship berthingOptimizing submarine berthing with a persistence incentive.* Naval Research Logistics 44 (1997) p301..318

[5] Daganzo, C.F. *The crane scheduling problem.* Transportation Research Part B 23 (1998) p159..175

[6] Drewry Shipping Consultants *Annual Container Market Review and Forecast 2007/08.* London (2007) 250 pp

[7] Drewry Shipping Consultants *Container Forecaster 1Q08.* London (2008) 218 pp

[8] Grossmann H., Otto A., Stiller S., Wedemeier J. *Growth Potential for Maritime Trade and Ports in Europe.* Intereconomics (2007) p226..232

[9] Guan Y., Cheung R.K. *The berth allocation problem: models and solution methods.* Operations Research Spectrum 26 (2004) p75..92

[10] Imai A., Nishimura E., Papadimitriou S. *The simultaneous berth and quay crane allocation problem.* Transportation Research Part E: Logistics and Transportation Review 44, Vol.5 (2008) p900..920

[11] Imai A., Nagaiwa K.I., Tat C.W. *Efficient planning of berth allocation for container terminals in Asia.* Journal of Advanced Transportation 31 (1997) p75..94

[12] Imai A., Nishimura E., Papadimitriou S. *The dynamic berth allocation problem for a container port* Transportation Research Part B 35 (2001) p401..417

[13] Imai A., Nishimura E., Papadimitriou S. *Berth allocation with service priority.* Transportation Research Part B 37 (2003) p437..457

[14] Imai A., Nishimura E., Papadimitriou S. *Corrigendum to The dynamic berth allocation problem for a container port Transportation Research Part B 35 (2001) p401..417* Transportation Research Part B 39 (2005a) p197

[15] Imai A., Nishimura E., Papadimitriou S. *Berth allocation in a container port: using a continuous location space approach.* Transportation Research Part B 39 (2005b) p199..221

[16] Imai A., Nishimura E., Hattori M., Papadimitriou S. *Berth allocation at indented berths for mega-container ships.* European Journal of Operational Research 179 (2007) p579..593

[17] Kim K.H., Moon K.C. *Berth scheduling by simulated annealing.* Transportation Research Part B 37 (2003) p541..560

[18] Lai K.K., Shih, K *A study of container berth allocation.* Journal of Advanced Transportation 26 (1992) p45..60

[19] Legato P., Mazza R.M. *Berth planning and resources optimization at a container terminal via discrete event simulation.* European Journal of Operational Research 133 (2001) p537..547

[20] Li C.-L., Cai X., Lee C.-Y. *Scheduling with multiple-job-on-one-processor pattern.* Institute of Industrial Engineers Transactions 30 (1998) p433..445

[21] Lim A. *The berth planning problem.* Operations Research Letters 22 (1998) p105..110

[22] Liu J., Wan Y-W., Wang L. *Quay Crane Scheduling at Container Terminals To Minimize the Maximum Relative Tardiness of Vessel Departures.* Naval Research Logistics 53 (2006) p60..74

[23] Lokuge P., Alahakoon D. *Improving the adaptability in automated vessel scheduling in container ports using intelligent software agents.* European Journal of Operational Research 177 (2007) p1985..2015

[24] Moorthy R., Teo C. *Berth management in container terminal: the template design problem.* Operations Research Spectrum 28 (2006) p495..518

[25] Nishimura E., Imai A., Papadimitriou S. *Berth allocation planning in the public berth system by genetic algorithms.* European Journal of Operational Research 131 (2001) p282..292

[26] Park Y.-M., Kim K. H. *A scheduling method for Berth and Quay cranes.* Operations Research Spectrum 25 (2003) p1..23

[27] Peterkofsky R.I., Daganzo C.F. *A branch and bound solution method for the crane scheduling problem.* Transportation Research Part B 24 (1990) p159..172

[28] Stahlbock R., Vo S. *Operations research at container terminals: a literature update.* Operations Research Spectrum 30 (2008) p1..52

[29] Tang L., Li S. And Liu J. *Dynamically scheduling ships to multiple continuous berth spaces in an iron and steel complex.* International Transactions in Operational Research 16 (2009) p87..107

[30] Tavakkoli-Moghaddam R., Makui A., Salahi S., Bazzazi M., Taheri F. *An efficient algorithm for solving a new mathematical model for a quay crane scheduling problem in container ports.* Computers & Industrial Engineering 56 (2009) p241..248

[31] Vernimmen B., Dullaert W., Engelen S. *Schedule Unreliability in Liner Shipping: Origins and Consequences for the Hinterland Supply Chain.* Maritime Economics and Logistics 9 (2007) p193..213

[32] Wang F. and Lim A. *A stochastic beam search for the berth allocation problem.* Decision Support Systems 42 (2007) p2186..2196

Chapter 6

The berth allocation problem and quay crane assignment problem on container terminals using a CP approach

Authors:
Rowan Van Schaeren, *Antwerp Maritime Academy*, rowan.van.schaeren@hzs.be
Stephane Zampelli, *Dynadec*, szampelli@dynadec.com
Yannis Vergados, *Dynadec*, yannis.vergados@dynadec.com
Wout Dullaert, *University of Antwerp, ITMMA, Antwerp Maritime Academy*, wout.dullaert@ua.ac.be
Birger Raa, *Ghent University*, birger.raa@ugent.be

6.1 Abstract

We study the integrated berth allocation and quay crane allocation problem encountered at container terminals propose a novel approach based on constraint programming that is able to incorporate many realistic operational constraints. The costs for berth allocation, gang allocation, time windows, breaks and transition times during gang movements are optimized simultaneously. The model is based on a resource view where gangs are consumed by vessel activities. Side constraints are added independently to this core model. Experiments on both randomly generated and real-life problem instances show that the model can produce solutions with an additional cost of only 10% compared to an ideal operational setting in which all operational side-constraints are ignored.
Keywords: berth allocation, crane assignment, containers, terminal, constraint programming

6.2 Introduction

Already many articles have been written concerning the berth allocation problem and the quay crane assignment problem (also called tactical berth allocation problem TBAP). We therefore propose not to structure the literature review by means of solution techniques used but refer to the existing literature during the description of the operational constraints encountered.

First a detailed description of the berth and quay crane allocation problem is given in section 6.3. All relevant operational constraints are discussed in detail and references to the existing literature are provided. As this problem has been proven to be NP-hard by many authors ([3], [17]) we propose a constraint programming approach for tackling this subject. Together with this new approach we offer sample datasets that can be used for benchmarking purposes. How our model was implemented using constraint programming is described in Section 6.7. By means of a case study in the port of Antwerp, our proposed model and proposed benchmarking datasets are presented in Section 6.10.1. Computational results are represented in section 6.10.2 both of the case study and the proposed datasets together with a description of the software output by means of print-screens and a functional description. Section 6.11 concludes this paper and gives suggestions for further research.

6.3 The berth and quay crane allocation model

Allocating vessels to berths at a container terminal and assigning quay cranes for handling the vessels can seem straightforward, but the problem description together with the literature review will illustrate that multiple solution approaches exist and that the complexity increases drastically when more operational constraints are considered. In this section, we will also position our model vis-à-vis the classification for berth allocation and quay crane scheduling problems proposed by Bierwirth and Meisel [1] (6.4).

For clarity reasons the problem is decomposed in a berth allocation subsection (6.3.1), a crane allocation subsection (6.3.2) and a description of the proposed objective function (6.5). During the descriptions cross references are made to the other sections in order to capture the interrelation of the berth and quay crane allocation problems. The last subsection (6.6) of this section describes the main contributions of this paper.

6.3.1 BAP

Let us first discuss the berthing of vessels alongside a quay length: the question of where to berth a vessel depends on various aspects. What follows is a description of operational considerations encountered at a modern container terminal in Antwerp but most of them -possibly with minor differences - will apply to any modern container terminal in the world:

- Quay/vessel lengths and berthing positions: planning one vessel usually proves no problem but when multiple vessels require a berthing position at the same time, the available quay length must be considered. At any moment in time, the total length of all the vessels alongside should not be larger than the available quay length. In this calculation not only the lengths of the vessels need to be taken into account, but also the distances needed in front and after the vessels for safe mooring. The mooring ropes/wires used for securing the vessel along the quay length are attached to bollards on the quay and the length of these mooring ropes is in relation to the length of the vessel. For this reason, we propose not to work with a continuous quay length but with a hybrid one as described in Bierwirth and Meisel [1]: every vessel is assigned a mooring place or berth that is a multitude of bollard distances. Generally the distance between two bollards on the same quay is equal. We acknowledge that using a continuous quay could allow one to optimize some space. This additional distance will not be important as the mooring ropes always need to come ashore and they will hinder other vessels when mooring between the vessel and the bollards to which the vessel is moored. Figure 6.1 gives an example of using bollards for defining the space a vessel needs alongside the quay length. The position used in the figure is from bollard 2 till bollard 5. Other hybrid approaches in the literature include Cordeau

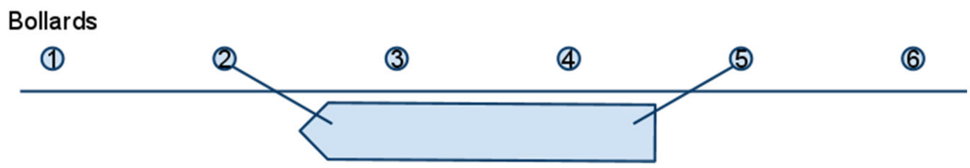


Figure 6.1: Using bollards for defining the quay length occupied by a vessel

et al. [6] who propose to start with a berth allocation with berths of a fixed length and afterwards allow for a dynamic repartition of berths when needed. They use a tabu search heuristic. The hybrid approach of Cheong et al. [5] splits the total berth length in discrete segments but considers the quay lengths in these segments as continuous. Their approach includes a local search heuristic, a hybrid solution decoding scheme and an optimal berth insertion procedure with a multi-objective evolutionary algorithm that incorporates the concept of Pareto optimality. Lokuge and Alahakoon [15] discuss a terminal with four main berths each with a fixed length. Each berth can contain one or two vessels as long as all other constraints are respected.

- Overlap: vessels on the same quay length should not overlap.

- Preferred berths: every vessel calling at a container terminal generally discharges and loads containers to and from the yard and has a preferred berth. This preferred berth is closely related to the use of a good yard management tool because when the vessel arrives at the terminal, all the containers to be loaded must be available on the yard and all the containers being discharged need to be placed on the yard. Managing the yard should be an optimization by itself with its own constraints. When looking to the yard from a berth allocation point of view, the yard can be considered as input because all positions need to be planned prior to the arrival of the vessel. Every discharged and loaded container moves between the yard and the vessel by a prime mover (e.g. straddle carrier). In order to minimize the transportation cost of feeding the quay cranes that handle the vessels, driving distances of the prime movers should be kept at a minimum as the customer pays a fixed price for the container loading/unloading regardless of the yard position the container will occupy. Figure 6.2 represents a bad berth allocation planning concerning the yard.

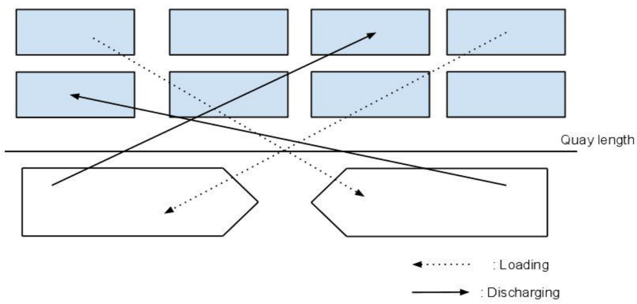


Figure 6.2: A bad example of berth allocation regarding the yard distance cost

For defining the preferred berth of a vessel not only the yard needs to be taken into account but also the:

  - type of quay cranes: it is possible that the available quay cranes for handling the vessel are not all of the same type. Certainly for the more recent container vessels this might prove important as they can stack their containers up to nine high on deck. Older types of cranes might not be able to reach them. If there is a mix of crane types it might also be interesting to take into account which container cranes are used for which vessels (e.g. with regard to spreader type, lifting capacity, productivity...).

- water depths: in the case of a long quay or quay in a river bend it is possible that the water depth is not the same for the complete length of the quay. The draft of the vessel needs then to be compared with the available water depth. By using a preferred berth you can also take this type of constraint in consideration.

Defining berthing places using bollards requires also the preferred berths to be defined in bollards. The preferred berth for the vessel depicted in Figure 6.1 could be 4-5 an not 3-4 (middle of the vessel). This would mean that the vessel is planned 20 meters from its preferred berth if the distance between 2 bollards is 20 meters. This would mean that on average every container discharged and loaded was moved 20 meters more than necessary by the prime movers.

An approach for generating the preferred berths on a container terminal is given by Moorthy and Teo [18]. The preferred berth is e.g. also considered in the following papers. Park and Kim[19] consider the preferred berth as the location nearest to the marshaling yard where outbound containers for the corresponding vessel are stacked. Wang and Lim [25] also count an additional cost if the vessel is not planned on the position with the lowest cost (preferred berth). Giallombardo et al. [7] approach the preferred berth from a transshipment point of view: vessels berthing within 600 meters from each other incur an incremental cost per meter. Whenever vessels are berthed more than 600 meters from each other, also housekeeping costs are counted.

- Time aspect: the time that a vessel occupies one or several berths depends on the handling time of the vessel. It is important when planning container vessels alongside the quay length that this time dimension is not forgotten. This aspect can be ignored more easily when solving instances with a time interval of e.g. twelve hours. It becomes more important though when working with time horizons of e.g. five days.

- Mooring direction/vessel height: when assigning more than one vessel alongside the quay it might also prove useful to look at the mooring side aspect of vessels: when planning vessels with many containers on deck or a high bridge it is best to plan them with their bows to each other. By doing this one allows more flexibility for assigning cranes as they probably wont́ need to lift or "top" their arms over the superstructure of the vessel in order to get from one vessel to the other. This "topping" would cause an idle time that could get as high as thirty minutes. We acknowledge that there are also probably containers stored behind the bridge that need to be handled but usually the number of containers before the bridge is more important. We also understand that it is not always possible to take this aspect into account as e.g. the current on a river might enforce a certain mooring side when (un)mooring. Figure 6.3 gives an impression of such a situation. This

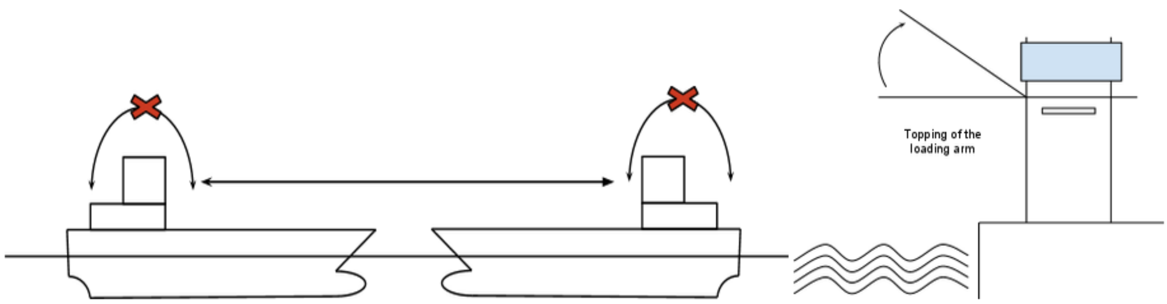


Figure 6.3: An example of a possible gain in crane usage by mooring vessels bow-to-bow and an example of crane "topping"

mooring side aspect is not taken into account by our proposed model as this would require more extensive datasets wherein also vessel characteristics are detailed together with the reaching heights of the cranes. This is also not easy to handle in a tidal port where tides can cause a change in vessel height of up to ten meters.

- Vessel setup times: when a vessel arrives at a terminal and is safely moored alongside the quay, the cranes cannot immediately start to discharge the containers. The securing of the containers, called lashings, first need to be undone and removed. The time needed for unlashing the containers differs per vessel and per stowage configuration. This time needs to be taken into account concerning the starting time for the gangs/cranes as they can not start to work on a vessel as long as the setup time for that vessel has not expired.

6.3.2 CAP

When assigning quay cranes to vessels several additional operational aspects need to be taken into account. An extensive overview of operational constraints is given here:

- Crane start: cranes can only start working on a vessel when the vessel has arrived

- Maximum available cranes: the available number of quay cranes for servicing vessels is limited at any moment in time

- Crane usage cost/handling time of a vessel: a quay crane is the most expensive piece of equipment on a container terminal. Managing the quay cranes is therefore an important aspect of container terminal operations. We propose to minimize the operational cost for handling each vessel. The terminal operator negotiates a time window for each vessel in which it needs to be handled when being alongside the quay length with the shipping lines. When the vessel does not arrive in the alloted time window (earlier or later), the terminal operator is less bound to handle the vessel in the agreed manner (e.g. assign continuously two quay cranes to the vessel). This would relax the required solution therefore this situation is not considered any further. When the vessel arrives on schedule and the terminal operator can not handle the vessel in the agreed time window, the terminal operator will have to pay a penalty to the shipper. By adding this additional contractual agreement to the model it is not necessary to handle each vessel as fast as possible: as long as the handling time windows are not violated it may be financially more interesting for the terminal operator to balance the used gangs over all the vessels in one shift. For the model, this means that another aspect of the terminal operator's cost should be added: the variable cost of using a gang over time. One gang consists e.g. of

  - one crane driver
  - one foreman: responsible for the whole gang
  - one checker: person controlling the container ID's being (un)loaded by the gang
  - two dockers: persons attaching/removing twist locks to the containers being (un)loaded by the gang
  - three drivers: persons driving the prime movers (e.g. straddle carriers) that service the crane of the gang

  The cost of using one such gang to handle a vessel depends in what shift (moment in time) the gang operates. An example of the relative gang costs are depicted in table 6.1. The implementation of commercial windows allows this shift-cost to be included in our objective function as described in section 6.5.

  Park and Kim [19] penalize every vessel that leaves after the predefined departure time using an integer programming model. Kim and Moon [12] penalized the late departure of a vessel together with the deviation from the preferred berth. They considered the handling time of each vessel as fixed though. A mixed-integer-linear-programming (MILP) model was for-

	Weekday	Saturday	Sunday
Day	1	1.50	2
Morning	1.05	1.50	2
Afternoon	1.15	1.50	2
Night	1.50	2	2

Table 6.1: Relative cost of a gang over the week

mulated. Meisel and Bierwirth [17] express the commercial time window by deviations from the arrival and departure times.

- Crane productivity: when considering the assignment of cranes to vessels, the purpose is to handle the vessels in an alloted time window. This time window has a direct link with the productivity of the quay crane considered. For the crane assignment this aspect is one of the most difficult to consider as a small variation on the crane productivity can have a huge impact on the scheduling of the other cranes. For modeling purposes an identical crane productivity is assumed for all quay cranes and all vessels. If one would like to model crane productivity in more details need to be taken into account such as:
  - weather: on modern high cranes the length of the cables required to pick up containers become very long. Due to wind and visibility considerations, the efficiency or productivity can be compromised.
  - crane driver: handling a container crane efficiently demands a lot of experience on the same crane. Two different crane drivers on the same crane could achieve a different productivity that is as high as ten containers per hour. This means that when two identical cranes work next to each other with a different crane driver, their productivity will not be the same.
  - number of prime movers servicing the crane: an other important aspect of the crane's productivity is the ability of the prime movers to service the quay crane. When something goes wrong or insufficient prime movers are available the maximal crane productivity can never be achieved.
  - stowage plan of the vessel and vessel characteristics: if all the containers in the same hold are for the same port it is easier to handle that hold. If there is a mix of destinations in one hold this will influence the productivity of the crane. When working close to the bridge or having to handle many containers with varying lengths the crane's productivity can also be influenced.

  - crane interaction: when many cranes have to work close to each other, also the prime movers will have a hard time to keep the same standard of safety thus forcing down the productivity.
  - crane and spreader characteristics: when there are different types of quay cranes on the same quay length, items like hoisting speeds and cad speeds could cause a deviation from the assumed productivity.
  - container securing equipment and crane shore gang: once a container is discharged, the twistlocks that secure the containers to each other on board need to be removed. Different types of twistlocks exist, each with a different handling time. If a container is discharged this container also needs to be physically inspected for external damage and the presence of a seal. When different types of twist locks are used this might lower quay crane productivity.

  Because of these considerations we work with a fixed average crane productivity.

- Transition times: when a crane finishes its work on a container vessel and moves to an other vessel this requires a repositioning idle time. We are aware of the fact that this idle time depends on many aspects like e.g. the height of the vessels, the mooring side of the vessels.... Without loss of generality we will assume a fixed average for the repositioning time.

- Crane breaks: a gang working on a crane generally works for a fixed amount of time with one break during that interval. In our model, we assume that each gang works for eight hours and that after four hours of work each gang gets a break of half an hour. During this period it is possible to have the cranes repositioned by electrician working at the terminal without causing additional idle time.

- Crane availability: as soon as one crane is available to start work on a vessel that vessel can be handled. There is no need to wait until all required cranes to handle the vessel are available simultaneously. Our proposed model allows cranes to be repositioned every minute. Transition times will limit these repositioning though. It is therefore possible that at the start of service a vessel receives three cranes and after one hour one crane moves to another vessel that just arrived. It might also occur that all the vessel's cranes are reallocated to the vessel.

- Number of cranes used during one shift: when a terminal operator needs to assign gangs for a certain shift he needs to order an integer number of gangs for the entire shift. It is therefore important that for each shift a fixed number of gangs is used to service all the vessels. Our model searches for solutions where the number of gangs used during a whole shift is the same. This is not always possible as the arrival and departure times of the vessels will not coincide with the starting and ending times of the shifts. For a solution to be practically useful the model needs to take these real-life considerations into account.

- Number of cranes on one vessel: each quay crane has a fixed width. For this reason only a maximal number of quay cranes can work on a vessel simultaneously. As for the considered terminals we visited the average width of the quay cranes was eighty meters. We propose to calculate the maximum number of quay cranes per vessel as follows: vessel length divided by eighty and rounded down.

- Crane ranges: each available quay crane cannot service the entire quay length. As the cranes are electrically powered the length of the power cables are chosen in such a way that an optimal coverage is obtained for the available quay length. An example is given in Figure 6.4 and an example for our proposed datasets is given in Table 6.4.

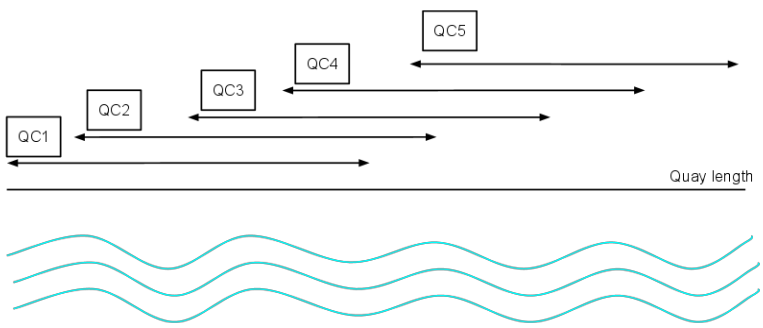


Figure 6.4: An example of quay crane (QC) ranges on a container terminal with five quay cranes

6.4 Model classification with regard to Bierwirth and Meisel

In order to situate our research in the academic world we also refer to the survey done by Bierwirth and Meisel [1]. The first proposed classification concerns the spatial attribute. Our proposed model is a variant of the hybrid layout defined in Bierwirth and Meisel [1]: the quay length is also partitioned into berths and large vessels may occupy more than one berth. The difference with our model is that one berth is much smaller than any vessel (twenty meters).

The second classification proposed by Bierwirth and Meisel [1] concerns the temporal attribute: our proposed model can be classified as *dynamic*. Each vessel has a fixed arrival time. The handling time attribute of our model can be classified as $GCAP$: the handling time of a vessel depends on the assignment of quay cranes. Our proposed quay crane assignment problem can be described as a variable-in-time assignment of cranes: the number of cranes assigned to a vessel can vary over time. As the quay crane assignment problem has a great impact on the berth allocation problem, crane assignments should be incorporated in the berth planning.

The last proposed classification by Bierwirth and Meisel [1] concerns the performance measure: our objective could be described as $\sum pos + tard + res$. It is a minimization of three weighed aspects: deviation from the desired berthing position of the vessels (pos), tardiness of a vessel against its desired departure time ($tard$) and resource utilization (res) affected by the service of a vessel. More detail is given in section 6.7.4.

Other authors that used the integration approach for the berth and quay crane allocation problem are e.g. Park and Kim [19] which propose a two-phase approach where the second phase details the quay crane schedules that were defined in the first phase together with the berthing allocations. A sub-gradient optimization technique is used for the first phase and a dynamic programming technique for the second phase. Meisel and Bierwirth [16] provide a heuristic approach and aim to minimize the idle time of the quay cranes. Imai et al. [11] use a genetic algorithm in which the two decision processes (berth and quay crane allocation) are iterated one by one. Liang et al. [14] use a hybrid genetic algorithm to find approximate solutions introducing vessel waiting times and delays to the objective function. Han et al. [8] use a mixed integer programming model and apply a simulation-based procedure to generate robust berth and QC schedules pro-actively.

For further information on the classifications and comparable papers we refer the reader to Bierwirth and Meisel [1].

6.5 Proposed objective function

Our proposed objective function is a total cost minimization of three parts:

1. Place deviation cost: as explained earlier in this paper (subsection 6.3.1) it is preferable for the terminal operator to berth the vessels as close as possible to their preferred berth.If a vessel deviates one meter from its preferred berth, all containers brought to or coming from the yard need on average to be driven one meter further than ideally possible. For our computational results we used a cost of half a euro per meter per container in either direction. As a prime mover has to travel the distance two times (to and from), this

value is multiplied by two. Each place deviation from the preferred berth of one meter will therefore cause an additional cost of one euro per container handled.

2. Lateness cost: this cost is also considered from the terminal operator's point of view as detailed earlier (subsection 6.3.2). For the lateness cost we based ourselves on the average operating cost of vessels. Therefore, we propose a penalty cost of five thousand euros per hour for violating the commercial time window. For our model we calculated the commercial time window as follows: ((total amount of containers to be handled / maximum number of cranes that can be deployed simultaneously) / crane productivity)*1.6. The last factor in the equation determines the relative size of the commercial time window. It is negotiated between the terminal operator and shipping lines. A value of 1.6 corresponds well to current practice at the Antwerp terminals. Without loss of generality we assume the lateness cost for all vessels to be equal.

3. Gang cost: a base cost of 2600 euro per gang per shift is used. This base cost is multiplied by the shift multiplier as detailed in Table 6.1.

When talking with the various container terminals we experienced that vessels longer than two hundred meters were considered more important for the berth and crane allocation problem than shorter ones. We therefore multiplied the two first terms of the objective function by ten for vessels longer than two hundred meters. The third term of the objective function is not multiplied by ten as the cranes within one shift can work on both categories of vessels.

By expressing all the components of the objective function in monetary terms, the model will be able to compare the cost impact of operational decisions such as placing a vessel at a certain berth or assigning an additional quay crane to a vessel. We acknowledge that all proposed values are approximations. For commercial reasons we are not allowed to publish any detailed cost figures.

6.6 Paper contributions

Several approaches are proposed in the literature for solving the berth and quay crane allocation problem (BAPCAP): genetic algorithms, hybrid parallel genetic algorithms, two-level heuristics ... We could find no paper, however, that approached the BAPCAP using constraint programming and believe that using this modeling approach would offer significant benefits for tackling real-life BAPCAP problems. Additional contributions of the current paper can be summarized as follows:

- The paper proposes an integrated approach for the berth allocation and quay crane allocation problem where cranes are allowed to move between vessels every minute while still solving instances with a time horizon of up to five days.

- When cranes move from vessel to vessel, transition times are taken into account which amongst others prevents the model from repositioning quay cranes too often.
- Whenever a vessel occupies a berth, a setup time is taken into account before quay cranes can start work on the vessels.
- For the gangs servicing the vessels, different labor costs are used depending on the shift in which they are deployed.
- Shift breaks are taken into account.
- The model levels the number of gangs used per shift as much as feasible in order to minimize idle time of the cranes.
- A graphical output of the results allows manual planners to quickly assess the quality of the provided solution, both in terms of berth/crane allocation as in gang usage.

6.7 Model description

Our proposed CP model is a composition of several submodels. Each submodel grasps a specific aspect of the problem. The core model, described in Section 6.7.1, allocates gangs to vessel activities across shifts, minimizing the total gang cost and the lateness. The crane allocation and the positioning of the vessels along the quay are ignored in the core model. Those two additional modular aspects are successively integrated into the core model, in section 6.7.2 and in 6.7.3. Section 6.9 shows an output of our model for a 3 vessel instance.

6.7.1 Gang allocation

In this section, we focus on the allocation of gangs to vessel activities. The following notations will be helpful.

Notations - A range R is a consecutive finite sequence of integers; its minimum (maximum) is noted $\underline{R}$ (resp. $\overline{R}$). The range of input vessels is denoted $vessels$, and for each $b \in vessels$, the range of vessel acitivities is denoted Act_b. The time horizon is represented by a range of time units, called $Horizon$. The range $Shifts$ indexes the shifts. The total shift duration (including breaks) is noted sd. The range $Gangs$ indexes the available gangs. The ranges $Gangs_b = [0, mc_b]$ with $b \in vessels$ represent the possible values for the number of cranes that can be allocated to a vessel. The ranges $Breaks$ is the ranges of breaks. Unless stated otherwise, we assume those ranges start at zero. The lower bound (resp. upper bound) of a finite domain variable x is denoted $\underline{x}$ (resp. $\overline{x}$).

Definition 1 (Crane Productivity) *The productivity of a crane is the number of containers per hour it can handle.*

For a given vessel b, the number of containers to be handled can be converted into workforce using the following relation:

Definition 2 (Workforce) *Given a crane productivity p, the workforce needed to handle c containers is defined by $(c * 60)/p$. The required workforce of a vessel, noted mw_b, is the workforce corresponding to its number of containers to handle.*

Workforce is a duration times a number of gangs and measures the effort required to handle a vessel, or alternatively is the number of minutes needed for a single gang to handle a given number of containers. The conversion of containers to workforce allows us to use cumulative constraints to handle the gang allocation.

The only drawback is that a crane may be reassigned while a container is being moved, since only the required time is considered. However, this limitation has no practical impact: transition times can be shortened or extended to handle that kind of limit cases in practice.

We consider the set of activities $a_{b,i}$ with $b \in vessels$ and $i \in Act_b$.

Definition 3 (Activity) *An activity $a_{b,i}$ is defined by five variables:*

- *$s_{b,i}$ is the starting time,*
- *$e_{b,i}$ is the completion time,*
- *$d_{b,i} = e_{b,i} - s_{b,i}$ is the duration,*
- *$cap_{b,i}$ is the amount of resource consumed by the activity between its starting time and its completion time.*
- *$wkf_{b,i}$ is the workforce delivered by the activity, with $0 \leq wkf_{b,i} \leq cap_{b,i} * d_{b,i}$.*

In our model, one activity $a_{b,i}$ is created per vessel b and per index $i \in Act_b$. The capacity $cap_{b,i}$ is the number of gangs used by the activity.

The equality of $wkf_{b,i}$ with $cap_{b,i} * d_{b,i}$ is not enforced because of breaks and transition times. For instance, if an activity overlaps a break, the delivered workforce is below this maximum. Breaks and transition times are handled at the end of this section. Activities can be interrupted and are also optional (they can have a zero duration).

Definition 4 (Time Window) *The time window of a vessel $b \in vessels$ is the couple (ta_b, td_b), where the integer ta_b denotes the arrival time of the vessel b and td_b the deadline of vessel b.*

For each vessel $b \in vessels$ and each index $i \in Act_b$, the arrival time is enforced:

Constraint 1 (Arrival) $\forall\, b \in vessels, i \in Act_b : s_{b,i} \geq ta_b$

Constraint 2 (Required Workforce) $\forall\, b \in vessels : \sum_{i \in Act_b} wkf_{b,i} \geq mw_b$

Let us ignore shifts for now. At any point in time, there is maximum $\overline{Gangs}$ gangs that can be hired. Given two variables s and d representing the starting time and the duration variables of an activity a_i, the mandatory part noted $mand(a_i)$ or $mand(s,d)$ is a range $[\overline{s} - \underline{d}, \underline{s} + \underline{d}]$ that can be empty if the mandatory range does not exist. This can be modeled by a cumulative constraint:

Definition 5 (Cumulative) *Consider a resource limited by a constant capacity c, and a set of activities $a_j \in A$. A constraint $cumulative(\{a_j \mid j \in A\}, c)$ ensures the following constraint: $\forall\, t \in Horizon\ \sum_{j \in I} cap_j \leq c$ where $I = \{j \in A \mid t \in mand(a_j)\}$.*

At any point in time, competing activities may not exceed the maximum number of available gangs:

Constraint 3 (Global Cumulative) *$cumulative(A, \overline{Gangs})$ where A is the set $\{a_{b,i} \mid b \in vessels,\ i \in Act_b\}$.*

Each vessel is also constrained on the maximum number of gangs at any point in time. To handle this, an additonal $\overline{vessels}$ number of cumulative constraints are posted:

Constraint 4 (Local Cumulative) *For each $b \in vessels$: $cumulative(A, \overline{Gangs_b})$ where A is the set $\{a_{b,i} \mid i \in Act_b\}$ and $Gangs_b$ is the possible gang range for vessel b.*

Let us introduce shifts in the model. For each shift, a variable denoting the number of gangs used can be created:

Definition 6 (Gang Shift) *For all $sh \in Shifts$, $nbGangs_{sh}$ is the number of gangs used in shift sh.*

For each shift, a fake activity is created that spans the whole shift and consumes the number of gangs that are not used by any activity during that shift.

Definition 7 (Fake Activities) *For all $sh \in Shifts$, a fake activity fa_{sh} is created with the following domains:*

- *starting time $s_{sh} = sh * sd$*

- *ending time* se_{sh}
- *duration* $d_{sh} = sd$
- *capacity* $cap_{sh} = \overline{Gangs} - nbGangs_{sh}$
- *workforce* $w_{sh} = 0$.

In the above definition, the variable $nbGangs_{sh}$ is linked with the fake activity fa_{sh}.

Let us introduce breaks and transition time. Two break intervals are present in each shift sh, a first break

$$[\frac{se_{sh}}{2} - bd, \frac{se_{sh}}{2}]$$

$$[se_{sh} - bd, se_{sh}]$$

where se_{sh} is the ending time of the shift sh and and bd is the constant break duration. Each break $r \in Breaks$ can be associated with such an interval noted b_r. A variable bi_r is equal to time intersection between b_r and $[s_{b,i}, e_{b,i}]$. The total intersection between an activity and the breaks can be measured:

$$bi_{b,i} = \sum_{r \in Breaks} bi_r \ .$$

Regarding transition times we consider a fixed and constant transition time denoted $transitionTime$ that is assigned to all activities. The transition time can be defined as

$$tt_{b,i} = max(0, transitionTime - fb_{b,i})$$

where $fb_{b,i}$ is defined as:

$$\begin{aligned} fb_{b,i} = \ & bi_r \quad \text{where } r = min\{r \in Breaks \mid bi_r \neq 0 \wedge s_{b,i} \in b_r\} \\ = \ & 0 \quad \text{if } r \text{ does not exist.} \end{aligned}$$

The variable $fb_{b,i}$ denotes the intersection of a break with the beginning of a vessel operation. Indeed cranes can be moved during breaks. Breaks occuring at the beginning of vessel operations hence shortens transition time.
The actual workforce of the activity (b, i) can be defined:

Constraint 5 (Workforce) *For each activity* (b, i), *the workforce is*

$$wkf_{b,i} = (d_{b,i} - bi_{b,i} - tt_{b,i}) * cap_{b,i} \ .$$

Regarding the setup time, the transition time assigned to the first activity of the vessel stands for both the transition time of the cranes and the setup time. In this core model, gangs across shifts are assigned to vessels, using preemptive activities. Breaks and transition times are taken into account. This first model is a relaxation of the problem as actual cranes along the quay are not assigned to vessels and vessel conflicting positions are ignored.

6.7.2 Space allocation

Along the quay, the vessels should not overlap. Let us define a vessel position along the quay:

Definition 8 (Position) *The position of vessel* b *along the quay is denoted* pos_b*.*

Let us define the starting and ending time of vessel:

Definition 9 (vessel Time Window) *The starting time of a vessel* c *is* $s_b = \min_{i \in Act_b} s_{b,i}$*, and its ending time is* $e_b = \max_{i \in Act_b} e_{b,i}$*.*

Non overlap between vessels is stated by enforcing that vessels overlapping in time should not overlap in space:

Constraint 6 (Non-overlap) $\forall\ (b, c) \in vessels \times vessels, b \neq c : (s_b < e_c) \wedge (e_b > s_c) \Rightarrow (pos_c \geq pos_b + length_b) \vee (pos_b \geq pos_c + length_c)$

6.7.3 Crane allocation

In this section a tractable submodel is presented for the crane allocation. This model can filter any inconsistent crane assignment value once the information is available from other submodels.

The assignment of cranes to a vessel can be respresented as a range since they are operated on rails and can not cross each other.

Definition 10 (Crane Range) *The crane range of a vessel* (b, i) *(*$i \in Act_b$*) is a range* $[sc_{b,i}, ec_{b,i}]$*, where* $sc_{b,i}$ *is the starting crane and* $ec_{b,i}$ *the ending crane. The variable* $nbCranes_{b,i}$ *denotes the number of cranes assigned to vessel activity* (b, i)*.*

The following constraint holds: $sc_{b,i} \leq ec_{b,i}$, and the number of cranes and the crane range are linked by: $nbCranes_{b,i} = ec_{b,i} - sc_{b,i} + 1$.

Each crane has a certain span along the quay because due to the reach of the electrical cables that service the cranes. This means that a crane can be assigned to a vessel if and only if the crane can reach the vessel along the quay. Given a vessel b, its length along the quay $length_b$, only a subset of crane ranges are available for vessel b. Let us define the $craneMin$ array indexed by bollard positions. The value $craneMin_p$ is the leftmost crane that can reach bollard range $[p, p+length_b]$.

Let us define the $craneMax$ array indexed by bollard positions. The value $craneMax_p$ is the rightmost crane that can reach bollard range $[p, p + length_b]$. The consistency between crane positions and vessel positions can be added to the model:

Constraint 7 (Crane Position) $\forall\, b \in vessels, i \in Act_b : sc_{b,i} \geq craneMin[pos_b]$ *and* $ec_{b,i} \leq craneMax[pos_b]$ *.*

The following set of constraints distribute the cranes among subactivities.

Constraint 8 (Crane Allocation) *For each pair of distinct activities* $((b,i),((c,j))$, *if they overlap in time their crane range must follow their relative position:*

$$[(s_{b,i} < e_{c,j} \wedge e_{b,i} > s_{c,j}) \wedge \ (pos_b < pos_c)] \Rightarrow ec_{b,i} < sc_{c,j}$$

and:

$$[(s_{b,i} < e_{c,j} \wedge e_{b,i} > s_{c,j}) \wedge \ (pos_b > pos_c)] \Rightarrow sc_{b,i} > ec_{c,j} \ .$$

Once the position, the time span and the number of cranes of pairwise activities are bound, the right side constraints from Constraint 8 form a linear chain of inequality constraints. Given a time $t \in Horizon$, a total order is enforced upon crane range variables of activities intersecting in time t. Ignoring distinction between vessel and activity indexes, we have at a given time $t \in Horizon$:

$$sc_1 \leq_{k1} ec_1 < sc_2 \leq_{k_2} ec_2 < \ \ldots \ \leq_{k_{n-1}} ec_{n-1} < sc_n \leq_{k_n} ec_n \qquad (A)$$

where n is the number of vessel activities intersecting in time with t. $\leq_{k_i}$ is a notation for the binary constraint $s_i \leq e_i - k_i + 1$, k_i is the bound value of variable $nbCranes_i$, and $<$ is the binary inequality constraint.

In the following, we prove that the chain of constraints (A) is tractable: the fixpoint computation only leaves values that can be extended to a solution. Consequently, if instantiation of crane range variables in (A) is impossible, the set of constraints (A) fails at fixpoint.

It is well-known [Jeavons, 1995] that max-closed (or min-closed) constraints and arc-consistency detect at fixpoint if a constaint system is satisfiable. Both constraints $x < y$ and $x \leq_k y$ are max-closed and min-closed. Let us define $max(a,b) = a$ if $a > b$, b otherwise; and $min(a,b) = a$ if $a < b$, b otherwise.

Definition 11 (Min/Max-closed) *A binary constraint* $B(x,y)$ *is max-closed iff given two tuples* (a_1,b_1) *and* (a_2,b_2) *valid for* B*,* $(max(a_1,a_2), max(b_1,b_2))$ *is still valid for* B*. A binary constraint* $B(x,y)$ *is min-closed iff given two tuples* (a_1,b_1) *and* (a_2,b_2) *valid for* B*,* $(min(a_1,a_2), min(b_1,b_2))$ *is still valid for* B*.*

Property 1 *The binary constraints* $x < y$ *and* $x \leq_k y$ *are min- and max-closed.*

Proof The inequality constraint $x < y$ is max-closed. Suppose $max(a_1,a_2) = a_i$ with $i = 1$ or $i = 2$, then if $max(b_1,b_2) = b_i$, $a_i < b_i$; otherwise, if $max(b_1,b_2) = b_j$ with $j \neq i$, $a_i < b_i \leq b_j$.

The inequality constraint $x < y$ is min-closed. Suppose $min(a_1,a_2) = a_i$ with $i = 1$ or $i = 2$; $a_i < b_i$ and $a_i \leq a_j < b_j$ with $j \neq i$.

The constraint $x \leq_k y$ is max-closed. Suppose $max(a_1,a_2) = a_i$ with $i = 1$ or $i = 2$. If $max(b_1,b_2) = b_i$, then $a_i \leq_k b_i$. If $max(b_1,b_2) = b_j$ with $j \neq i$, $a_i \leq_k b_i \leq b_j$, hence $a_i \leq b_j$.

The constraint $x \leq_k y$ is min-closed. Suppose $min(a_1,a_2) = a_i$ with $i = 1$ or $i = 2$. By hypothesis, $a_i \leq_k b_i$, and $a_i \leq_k a_j \leq_k$ with $j \neq i$. ∎

The following results are adapted from [Jeavons, 1995].

Property 2 *A CSP that contains only binary max-closed constraints and that is pair-wise consitent has a solution or fails at fixpoint. A solution can be obtained by selecting the max value of the domain of each variable. If the constraints are all min-closed, the property holds and a solution can be obtained by selecting the min value of the domain of each variable.*

The set of constraint (A) removes at fixpoint all impossible values from crane range variables.

Property 3 *Suppose the arc-consistent fixpoint has been computed for the chain of constraints (A) and the fixpoint does not fail. Then any value from any variable in the set of variables of (A) can be extended to a solution.*

Proof Let us rewrite the chain of constraints (A) in the following way:

$$x_1 < \ldots < x_i < \ldots < x_n \qquad (B)$$

where variables are ordered and indexed, and we do not distinguish between $x < y$ and $x \leq_k y$ constraints and we simply note $<$ for both, as we only use their min/max-closed property in the following.

Suppose we pick up a random variable x_i in the chain (B), and a random value v in the domain of x_i. If that value is the maximum of the domain, we are done because the constraints are all max-closed and because of property 2: select the max value from the domain of each variable. If that value if the minimum of the domain, we are done because the constraints are all min-closed and because of the property 2: select the min value from the domain of each variable.

Suppose the value v is not equal to one of the bounds of the domain of x_i. Because the contraints are all max-closed, we can build a partial solution $\overline{x_i}, \overline{x_{i+1}}, \ldots, \overline{x_n}$ for the variables $x_i, x_{i+1}, \ldots, x_n$. Because the contraints are all min-closed, we can build a partial solution $\underline{x_1}, \ldots, \underline{x_{i-1}}, \underline{x_i}$ for the variables $x_1, \ldots, x_{i-1}, x_i$. It is clear that $\underline{x_{i-1}} < \underline{x_i} < v$. Moreover, $v < \overline{x_i} < \overline{x_{i+1}}$. Hence there exists an instantiation $\underline{x_1} < \ldots < v < \overline{x_{i+1}} < \ldots < \overline{x_n}$ that satisfies the chain of constraints (B). ∎

This last property implies that the labeling of the crane range variables can be skipped as propagation will ensure crane ranges can be instantiated to a solution.

6.7.4 Objective

The three components of the objective includes the lateness cost, cost induced by the distance with the ideal position, and the total gang cost. The lateness of a vessel $b \in vessels$ is easily defined:

Definition 12 (Lateness) *The lateness* l_b *of a vessel* $b \in vessels$ *is equal to* $\max(0, e_b - td_b)$.

Lateness represents the time by which the commercial time window of a vessel is exceeded. A position difference can be defined similarly:

Definition 13 (Distance Gap) *The distance gap* dp_b *of a vessel* $b \in vessels$ *wrt its ideal position* ip_b *is equal to* $|ip_b - pos_b|$.

The number of gangs used in each shift is already defined by $nbGangs_{sh}$, see Section 6.7.1.

Constraint 9 *The objective variable obj is defined as*

$$obj = \sum_{b \in vessels} (l_b * lc_b) + \sum_{b \in vessels} (dp_b * dc_b) + \sum_{sh \in Shifts} (nbGangs_{sh} * gc_{sh})$$

where l_b *is the lateness cost per minute for vessel b,* dc_b *is the distance cost per meter for vessel b, and* gc_{sh} *is the cost of a single gang in shift sh.*

6.7.5 Heuristics

The primary goal of the heuristics is to minimize the total gang cost per shift while avoiding lateness. To minimize a resource in a cumulative constraint, a *fill hole* heuristic is used. The idea is to fill holes present inside the profile of the resource usage. The profile of a cumulative constraint can be defined as:

Definition 14 (Profile) *The profile of a cumulative constraint is a set tuples* (t_i, d_i, v_i), $i \in P$, *such that:*

- *(non-overlap)* $\forall\, i, j \in P,\ i \neq j : [t_i, t_i + d_i - 1] \cap [t_j, t_j + d_j - 1] = \emptyset$
- *(usage reflection)* $\forall\ t \in Horizon\ \exists\ i \in P : \sum_{k \in A} cap_k = v_i$ *where* $t \in [t_i, t_i + d_i - 1]$ *and* $A = \{j \in Act \mid t \in mand(a_j)\}$
- *(cover)* $\forall\, t \in Horizon\ \exists\, i \in P : t \in [t_i, t_i + d_i - 1]$

The set *Act* denotes the set of all activities. Tuples of a profile are called *segments.*

Definition 15 (Minimum Profile) *A cumulative profile is minimal if* $\forall\ i, j \in P, i \neq j,\ v_i \neq v_j$, *that is* $|P|$ *is minimum.*

In the following, we shall suppose that P is ordered wrt t_i. We note indifferently $i \in P$ and $(t_i, d_i, v_i) \in P$. A hole is an augmented segment and defined with respect to the left and right segments. The left (right) segment i of a profile P is the segment $i - 1$ (resp. $i + 1$). Its left (right) segment value is v_{i-1} (resp. v_{i+1}).

The left and right segment may be undefined if $i = \min(P)$ or $i = \max(P)$. If they are undefined, their left or right segment value is equal to $\overline{Gangs}$.

The profile segment is augmented with a depth information h:

$$h = \begin{cases} min(l - v_i, r - v_i) & \text{if } l - v_i > 0 \text{ and } r - v_i > 0 \\ l - v_i & \text{if } l - v_i > 0 \text{ and } r - v_i < 0 \\ r - v_i & \text{if } l - v_i < 0 \text{ and } r - v_i > 0 \\ 0 & \text{if } l - v_i < 0 \text{ and } r - v_i < 0 \end{cases}$$

where l and r are the left segment value and the right segment value resp. We say a segment is augmented by its hole value h.

The heuristic function uses a function called lmdh() for *leftmost deepest hole*. It returns an ordered sequence of holes based on the profile of the cumulative constraint that the next activity should try to fill. More specifically, considering the minimum profile P of the cumulative constraint, it returns a sequence O of augmented segments (t_j, d_j, v_j, h_j) such that:

1. O defines the same profile as P for C:

   $\forall\ t \in Horizon\ \exists\ j \in O : \sum_{k \in A} cap_k = v_j$ where $A = \{k \in Act \mid t \in mand(a_k)\}$.

2. O is not minimum since segments do not span shift limits:

   $\forall\, j \in O, \exists\, sh \in Shifts : t_j \geq sh * sd \wedge t_j + d_j - 1 \leq ((sh + 1) * sd) - 1$.

3. h_j is the augmented hole value from the segment $i \in P$ containing segment $j \in O$

4. the sequence O is sorted lexicographically on highest h_i and smallest t_i.

The labeling procedure is described in Algorithm 1. The vessels are scanned in increasing arrival time ta_b (line 1) and the activities of vessel b are scanned (line 3). The amount workforce still to be handled is computed (line 4), and if no workforce is left, the remaining activities Act_b are assigned to a duration of zero so that they do not appear in the solution (line 4 to 7). If there is work left to do on the current vessel, the profile holes are then computed based on the information of the cumulative constraint, by calling lmdh() (line 8). The holes are sorted according to their corresponding shift gang cost. The selected activity is forced to be included into the width of hole (line 9 to 11). The depth of the hole is ajusted if it is a border case. This can happen for instance if the left segment is undefined. Another possibility is that $h = 0$ because the segment is not a hole. In both cases, h_i is set to the maximum possible number of gangs for the activity (line 13 to 15). The number of gangs, based on the augmented segment, tend to be the number of gangs that would fill the hole vertically, if any. Then the number of gangs is assigned, the activity is pushed leftmost, and the workforce delivered is maximized, maximizing the width of the activity (line 17 and 19). The current index of the activity is added to the already scheduled activites (line 23). When all subactivities of current vessel have been scheduled, line 25 and 26 assign a position to the vessel along the quay. It should be stressed that the crane allocation range variables are not labeled, as the crane allocation submodel is tractable, see Section 6.7.3.

PROCEDURE label()

1: **for all** $b \in vessels$ by arrival order **do**
2: $\quad I \leftarrow \emptyset$ {I is the set of activities already used}
3: $\quad$ **for all** $i \in A_b : i \notin I$ **do**
4: $\quad\quad$ int $lw \leftarrow mw_b - \sum_{i \in A_b} \underline{wkf}_{b,i}$ {left workload}
5: $\quad\quad$ **if** $lw \leq 0$ **then** {if nothing to do for this vessel}
6: $\quad\quad\quad$ try constraint $d_{b,i} = 0$ {impose zero duration, as this activity is not used}
7: $\quad\quad$ **else**
8: $\quad\quad\quad$ **for all** $[t_i, d_i, v_i, h_i] \in$ lmdh() in increasing shift cost order **do**
9: $\quad\quad\quad\quad h_1 \leftarrow t_i$; $h_2 \leftarrow t_i + d_i - 1$;
10: $\quad\quad\quad\quad$ try constraint $s_{b,i} \geq h_1$ {restrict activity to the segment $[h_1, h_2]$}
11: $\quad\quad\quad\quad$ try constraint $e_{b,i} < h_2$
12: $\quad\quad\quad\quad h \leftarrow h_i$
13: $\quad\quad\quad\quad$ **if** $h_i = 0$ **or** $h_i > \overline{nbCranes}_{b,i}$ **then** {if it is not a proper hole}
14: $\quad\quad\quad\quad\quad h \leftarrow \overline{nbCranes}_{b,i}$ {set to max nbr of gangs for vessel b}
15: $\quad\quad\quad\quad$ **end if**
16: $\quad\quad\quad\quad$ **for all** gangs g from h down to $\underline{nbCranes}_{b,i}$ **do**
17: $\quad\quad\quad\quad\quad$ try constraint $nbCranes_{b,i} = g$ {impose nbr of cranes, starting from depth h}
18: $\quad\quad\quad\quad\quad$ try constraint $s_{b,i} = \underline{s_{b,i}}$
19: $\quad\quad\quad\quad\quad$ try constraint $wkf_{b,i} = \overline{wkf}_{b,i}$ {fix duration, as start and nbr of gangs are fixed}
20: $\quad\quad\quad\quad$ **end for**
21: $\quad\quad\quad$ **end for**
22: $\quad\quad$ **end if**
23: $\quad\quad I \leftarrow I \cup \{i\}$
24: $\quad$ **end for**
25: $\quad$ try constraint $diffPos_b = \underline{diffPos}_b$ {label position close to the ideal position}
26: $\quad$ try constraint $pos_b = \underline{pos}_b$ {diffPos is an absolute value}
27: **end for**

Algorithm 1: Dedicated heuristics for the global model.

The above heuristic obtains good solutions. Using a naive heuristics, where activities are pushed leftmost lead to worse result as demonstrated in the experiments. Moreover, we use large neighborhood search [22] where entire vessels are fixed with a 0.6 probability.

In order to be able to compare the proposed approach a similar MIP model is written using the same tools. The following section describes this MIP model.

6.8 Gang Allocation MIP Model

For the core submodel (see Section 6.7.1) in the MIP, the time windows are ignored and the gang assignment can be modeled as a flow problem. Considering all vessels, their required mw_b has to be distributed into eligible shifts (shifts intersecting with their vessel time windows) such that the total gang cost is minimized.

Let $x_{b,sh}$ be a float denoting the amount of workforce assigned to a vessel b in shift sh. As in the constraint programming model, at least mw_b workforce has to be spent on vessel b:

Constraint 10 $\forall\, b \in vessels : \sum_{sh \in Shifts} x_{b,sh} \geq mw_b$

There is also a limit in each shift on the workforce, given the work time available in the shift and the maximum number of cranes that can be assigned to vessel b:

Constraint 11 $\forall\, b \in vessels, sh \in Shifts : x_{b,sh} \leq mc_b * wt$

Worktime wt is equal to the shift duration minus the breaks. Given the float variable w_{sh} that represents the total workforce spent in the shift sh:

Constraint 12 $\forall\, sh \in Shifts : w_{sh} = \sum_{b \in vessels} x_{b,sh}$

The number of gangs g_{sh} needed in shift sh can be deduced:

Constraint 13 $\forall\, sh \in Shifts : g_{sh} = \frac{w_{sh}}{wt}$

The objective function, only considering the total gang cost, can be stated as follows:

Constraint 14 $minimize \sum_{sh \in Shifts} g_{sh} * gc_{sh}$

The time window can also be taken into account by computing the actual work time left in the shift sh because of the vessel time windows. This left work time noted $rt_{b,sh}$ is equal to the length of the range $[ta_b, td_b] \cap [sh * sd, ((sh+1) * sd) - 1]$. The maximum workforce available can be restricted:

Constraint 15 $\forall\, b \in vessels, sh \in Shifts : x_{b,sh} \leq rt_{b,sh} * mc_b$

	a	d	l	c	mc	pos	rw	lc	pc
vessel A	0	210	170	70	1	0	120	1	1
vessel B	100	860	190	791	4	0	1356	10	10
vessel C	300	400	170	70	2	0	120	1	1

Table 6.2: Input data for the three vessel examples. The column from left to right denotes: arrival, deadline, length, number of containers, maximum number of concurrent cranes working on the vessel, preferred position in meters, required workforce, lateness cost and position cost

The proposed MIP model is a lower bound of the global constraint programming model. It can be viewed as a model with no operational constraints.

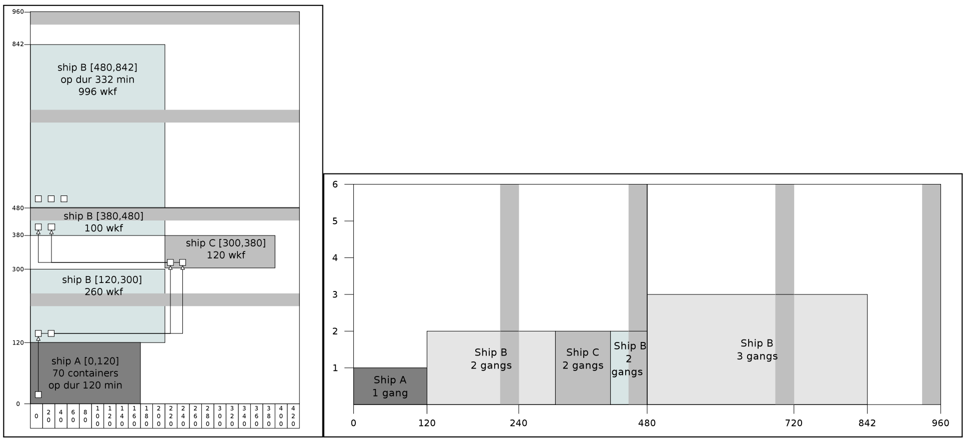
The model allows vessels to be positioned anywhere along the quay and assumes that quay cranes can service any vessel.

The proposed MIP model is also a lower bound relaxation of the gang allocation model from Section 6.7.1 because of vessel time windows. Consider an example with one shift of 4 time units [0,1,2,3], 4 available gangs and a single vessel b with a time window [1,2]. Suppose the required workforce mw_b of the vessel is 8. It is clear that 4 gangs are needed, because the time window of vessel b has a duration of 2 units. The MIP model finds the optimal and single solution $x_{b,sh} = 8$, $w_{sh} = 8$, and the number of gangs $g_{sh} = 8/4 = 2$. The MIP model answers that 2 gangs are needed instead of 4.

6.9 Output sample

To illustrate the process of crane allocation, suppose there are two shifts of eight hours, each shift containing two breaks of 30 minutes. The first break ends at the middle of the shift, while the second break terminates the shift. Hiring one gang in the first shift costs 200, while hiring one gang in the second shift costs 100. We have 6 cranes available which span the whole quay length. There are three vessels calling, vessel A, B andC. Table 6.2 summarizes the data. The required workforce is the time for one crane to complete the handling of the containers. For instance, the required workforce for vessels A and C is (60/35)*70=120 minutes for one crane. 35 being the crane productivity per hour and 70 the number of containers to be handled. The required workforce for vessel B is (60/35)*791=1356 minutes for one crane. Vessel C arrives and must be completed in the second part of the first shift. All vessels have their preferred position at the beginning of the quay.

Figure 6.5 shows the solution output by our model. Figure 6.5a shows the space and time arrangement of the vessels where the vertical axis measures time and the horizontal axis represents the quay. The quay is divided into segments



(a) (b)

Figure 6.5: Solution for the three vessels instance

of 20 meters. Time is expressed in minutes. Grey zones denote breaks. A square on a vessel activity represents one crane/gang operating on that vessel. There is a directed arrow between two cranes/gangs whenever it is the same crane/gang, denoting a crane/gang reallocation. Figure 6.5b shows a resource view, where the vertical axis represents the number of gangs, while the horizontal axis time.

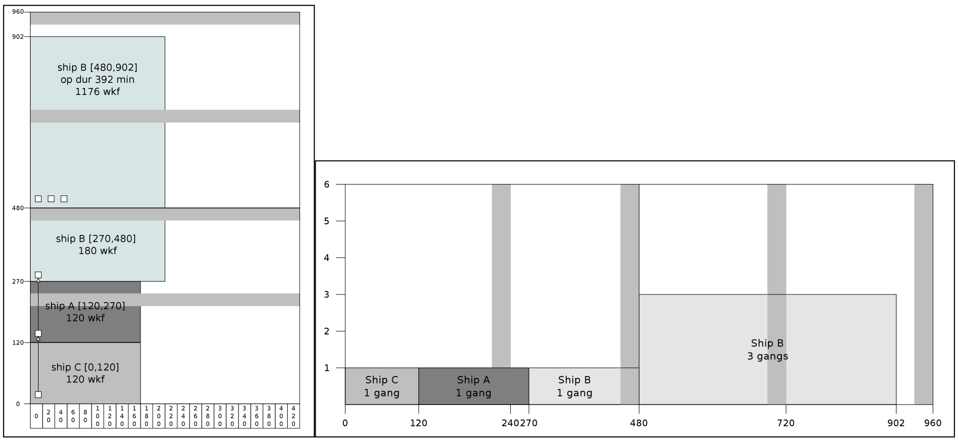
Both vessels A and B are placed at their preferred position, while vessel C is pushed after vessel B: vessel B has a position cost of 10, while vessel C has a position cost of 1. Because of its time window, vessel C is forced to use two gangs. Hence at least two gangs must be used in the first shift. At most one crane can operate on vessel A. Vessel B uses those two gangs in the first shift. Note that crane reallocation occurs between vessel B and vessel C: gangs processing vessel B are interrupted to handle vessel C.

Two gangs are used in the first shift, and three in the second shift. The gang cost is 700. There is no lateness cost. The position cost is 11, because vessel C is pushed after vessel B. Hence the total optimal cost is 711.

The output of the relaxed model is shown in Figure 6.6. Only one crane is needed for the first shift, since all operational constraints are ignored. The gang cost is 500. This represents a difference of 29.6% between the two solutions regarding gang cost.

6.10 Computational results

This section assesses the performance of the proposed model by means of an industrial dataset and generated datasets exhibiting real-life problem features. Section 6.10.1 describes how these datasets are generated. Section 6.10.2 analyses the performances by comparing the cost values with the lower bound MIP approach (see



(a) (b)

Figure 6.6: MIP solution for the three vessels instance

Section 6.8).

6.10.1 Datasets for validation

For validating the model datasets were generated, based on the authors' experiences and information found in various published academic papers. All details of our datasets are included in this paper in order to propose a benchmark to compare BAPCAP models in future research.

Our input for generating the datasets is as follows:

- Time horizon: We suggest using a time horizon of 5 days. The size of the planning window will therefore be 7200 minutes.

- Total quay length: This represents the horizontal line of the planning window. We propose datasets with an available quay length of 2000 meters.

- Number of vessels: The number of vessels is fixed to ten. The lengths and amount of containers to be handled are randomized but based on operational data obtained at a container terminal.

- Crane productivity: We propose an average crane productivity of 35 containers per hour or 0.5833 per minute. This value is an average for all containers handled: loading and unloading, full or empty containers for one TEU (twenty feet equivalent unit), one twin lift (two TEU at the same time) or one forty feet container. The total amount of quay cranes available is set to nineteen. All cranes are assigned to a certain shift in which they work. Details of the shifts (working hours and breaks) are detailed in table 6.3.

	Break 1	Break 2
Morning (06:00-14:00)	09:30-10:00	13:30-14:00
Afternoon (14:00-22:00)	17:30-18:00	21:30-22:00
Night (22:00-06:00)	01:30-02:00	05:30-06:00

Table 6.3: Gang working times and breaks

- Crane width: This value is used to calculate the maximum number of cranes that can service a vessel at the same time. In this crane width is also included the safety distance required to operate two cranes safely next to each other. We propose to set the crane width to 80 meters. This means that a vessel of 230 meters e.g. would have at most 2 cranes working on it simultaneously: rounddown(230/80).

- Bollard distance: This represents the distance expressed in meters between two consecutive bollards on the quay length. Our model uses distances between bollards to create berths for vessels. A distance of 20 meters is suggested. This also means that every vessel will be assigned a multitude of 20 meters of length at the quay. This distance is also used to add to the vessel's length fore and aft for allowing a safe mooring alongside the quay length. We understand that in extreme situations a surplus of 19 meters will be used for a vessel of 101 meters of length as 120 meters will be reserved and in addition 20 meters fore and aft of the vessel in order to moor safely.

- Quay crane position: The reach alongside the quay length of each quay crane is limited. Tabel 6.4 details the reach of each quay crane alongside the quay length in function of a begin bollard and an end bollard.

- Vessel length: the length of each vessel is the amount of space each vessel takes up alongside the quay length. Included in this value is also the safety distance before and after the vessel that is required to moor the vessel safely.

- Commercial time factor: this factor expresses the operational freedom a terminal operator has to handle each vessel compared to the minimum handling time. The minimum handling time for a vessel can be expressed as the total number of containers to be handled divided by the maximum number of cranes allowed simultaneously on that vessel and then again divided by the crane productivity. A 230 meter vessel that has 2,000 containers to be handled would have a minimum handling time of 19 hours or 1,142 minutes: $2000/3/35 = 19$. In line with commercial practice we propose a commercial time window factor of 1.6. This means for our previous example that the vessel can stay thirty point four ($19 * 1.6 = 30.4$) hours alongside the quay length without incurring any lateness costs.

Crane ID	From Bollard	To Bollard
Q1	1	51
Q2	3	52
Q3	4	53
Q4	6	55
Q5	7	57
Q6	8	59
Q7	14	69
Q8	15	71
Q9	16	72
Q10	18	73
Q11	27	86
Q12	28	87
Q13	39	91
Q14	40	93
Q15	42	94
Q16	44	95
Q17	62	97
Q18	64	99
Q19	79	100

Table 6.4: Quay crane reaches alongside the quay length expressed in bollards

- Lateness cost per vessel: if a vessel stays longer alongside the quay length than allowed by its commercial window, a lateness cost is incurred. This lateness cost is contractually negotiated. Without loss of generality we propose a lateness cost of 5000€per hour.

- Position cost: when a vessel is not positioned at its ideal berthing location an addition cost is incurred of one euro per meter of deviation per container.

- Shift cost: We propose 2600€as gang cost multiplied by a shift factor for each type of shift as detailed in table 6.1. A shift cost is therefore the sum of all the gangs used that shift times 2600 times the shift factor.

- Setup times: the setup times described in sections 6.3.1 and 6.6 are set to 20 minutes. This means that twenty minutes before start of operations the berth is occupied but no crane productivity can be used. This setup time also needs to be taken into account at the end of operations, meaning that here also the cranes are idle for twenty minutes or can move to another vessel before a new vessel can claim the same berth.

- Transition times: the transition times of a crane moving from one vessel to another is set to 20 minutes.

Table 6.5 gives an overview of the parameter values of each dataset.

6.10.2 Results

The goal of our experiments it to measure the operational distance between the constraint programming model handling all operational constraints and the lower bound MIP model which ignores all operational constraints and focuses on balancing the workforce. All runs were performed on a 2.53Ghz Intel CPU with 1GB of RAM. A time limit of 10 minutes is imposed and per vessel ten activities are allowed. The MIP solver is SCIP [23] and the constraint programming solver is Comet. Table 6.6 shows the results for both the generated datasets and the industrial dataset.

Three models were used. All models use a LNS procedure that fixes randomly vessels with a 0.6 probability. The first one is the *fill-hole* model that uses the *fill hole* heuristic (see section 6.7.5). The second model is the *naive* model where a naive heuristic is used that assigns activities in a leftmost manner without considering the profile. The last one is the *fill-hole-relax* model where there is no crane range constraints, no non-overlap constraints, no transition time and time windows are relaxed to the boundary of the shift. This *fill-hole-relax* is used to measure the performance of the CP approach against the MIP approach. Because of the time windows relaxation, constraint 15 in the MIP model is also relaxed. The MIP solution is thus different in line *relax*. Constraint 15 cannot be easily stated in our CP model.

In Table 6.6 the time in seconds is given for the best solution found. If the MIP approach finished before the timeout of 600 seconds, optimality has been proven by the MIP. The distance in percentage with the MIP objective value is given. Finally, the number of additional gangs hired with respect to the lower bound MIP approach is printed. When a line is marked '-' it means the constraint programming model did not find any solution before the timeout.

A first remark justifying the *fill hole* heuristics (see Section 6.7.5) is that naive heuristics performs poorly compared to the *fill hole* heuristics. The *naive* model did not find any solution before the timeout in 3 out of 4 random instances and uses two times the number of gangs in the industrial instances. The *naive* model tends to have a lower position cost. The *fill-hole-relax* CP approach is trapped in local optima, but finds good solutions up to 2% of the best generated result. This is expected as MIP is known to be stronger for flow-like problems. The proposed CP model is able to handle all the additional operational constraints for an additional

Vessel ID	Length	Arrival	Workload	Priority	Berth	Cranes
V1	176	531	1968	1	764	2
V2	229	670	1731	1	1155	2
V3	180	2824	1586	1	98	2
V4	108	3000	1119	1	334	1
V5	135	3386	780	1	399	1
V6	157	3553	874	1	237	1
V7	116	4000	782	1	242	1
V8	235	4047	2669	1	427	2
V9	177	4748	2035	1	881	2
V10	208	5436	1582	1	1060	2

(a)

Vessel ID	Length	Arrival	Workload	Priority	Berth	Cranes
V1	136	176	867	1	105	1
V2	138	502	1002	1	1304	1
V3	107	2046	729	1	78	1
V4	126	2111	906	1	460	1
V5	195	2173	2004	1	109	2
V6	150	2882	910	1	416	1
V7	209	2993	1921	1	1315	2
V8	240	3415	2982	1	919	3
V9	201	3862	1414	1	918	2
V10	190	4520	1656	1	841	2

(b)

Vessel ID	Length	Arrival	Workload	Priority	Berth	Cranes
V1	217	346	1955	1	1144	2
V2	240	648	4109	1	1369	3
V3	215	830	2398	1	1359	2
V4	211	896	1429	1	348	2
V5	165	1033	2189	1	1316	2
V6	144	2203	1266	1	374	1
V7	184	2446	1710	1	817	2
V8	240	2796	3731	1	465	3
V9	114	3630	922	1	863	1
V10	197	3654	2806	1	826	2

(c)

Vessel ID	Length	Arrival	Workload	Priority	Berth	Cranes
V1	206	219	2333	1	814	2
V2	206	1107	2477	1	312	2
V3	107	1484	883	1	303	1
V4	87	1557	1219	1	943	1
V5	190	2076	1537	1	158	2
V6	95	2267	1086	1	210	1
V7	241	2379	3329	1	1239	3
V8	247	2691	2233	1	525	3
V9	250	2825	3450	1	659	3
V10	123	2925	885	1	830	1

(d)

Table 6.5: Datasets used for the experiments. For each set are given the ID of the vessel, vessel length (meters), arrival time (minutes), workload (containers), priority (numerical value), preferred berth (meters), maximum amount of cranes (numerical value)

1/10 (7.8%) to 1/5 (18.8%) cost when compared to an ideal operational world where no operational constraints exist.

H	Time (sec) CP	Time (sec) MIP	MIP Dist.	Objective Value Total	Objective Value Gang	Objective Value Pos.	Objective Value L.	Extra Gangs
Random1, 10 vessels								
fill-hole	504	600	7.8	20648	20589	59	0	5 (67/62)
naive	600	600	-	-	-	-	-	-
fill-hole-relax	175	243	0.4	18522	18522	0	0	0 (62/62)
Random2, 10 vessels								
fill-hole	483	8	11.0	20553	20446	107	0	6 (65/59)
naive	385	7	27.8	25356	25321	35	0	7 (66/59)
fill-hole-relax	93	6	0.4	18314	18314	0	0	0 (59/59)
Random3, 10 vessels								
fill-hole	542	343	18.8	36433	36265	168	0	12 (104/92)
naive	600	356	-	-	-	-	-	-
fill-hole-relax	364	600	0.7	28587	28587	0	0	0 (92/92)
Random4, 10 vessels								
fill-hole	582	600	13.6	29998	29473	525	0	6 (86/80)
naive	600	600	-	-	-	-	-	-
fill-hole-relax	211	600	0.4	26509	26509	0	0	0 (80/80)
Industrial, 15 vessels								
fill-hole	458	2	11.9	15857	15666	191	0	4 (48/44)
naive	428	3	23.3	18209	18078	131	0	8 (52/44)
fill-hole-relax	501	2	0.9	14030	14030	0	0	0 (44/44)
Industrial, 30 vessels								
fill-hole	60	12	16.5	29884	29050	834	0	11 (90/79)
naive	338	12	41.1	42335	41530	805	0	26 (105/79)
fill-hole-relax	12	11	1.8	25878	25878	0	0	1 (80/79)

Table 6.6: Results for all instances

Figures 6.7 and 6.8 show screenshots of our tool for a solution in which 15 vessels are scheduled.
Figure 6.7 shows the sub-activities of the vessels. Horizontal axis is the time, while vessels are placed vertically from the bottom to the top according to their vessel ID. Shifts are represented by consecutive white and gray boxes in the background. Each shift displays its id and cost. For instance, the first shifted is numbered 0 and has a cost of 273, noted in the figure '0-273'. In each shift, there are two breaks depicted. Vessels are sliced into sub-activities. Each sub-activity is labeled with its vessel number, its sub-activity id, and its number of cranes and crane range. Time windows are also drawn. In this figure, the gang usage profile can also be seen. This profile is the sequence of horizontal lines. At each minute, the height of the profile represents the number of gangs used.
Figure 6.8 shows the positions of the vessels along the quay. The horizontal axis represents the space expressed in the bollards along the quay. The vertical axis represents the time. Each shift is depicted with its shift id and its cost. Breaks are also displayed with black horizontal lines. For each vessel, a line is drawn between its ideal position and its actual position (bottom left corner of each rectangle).

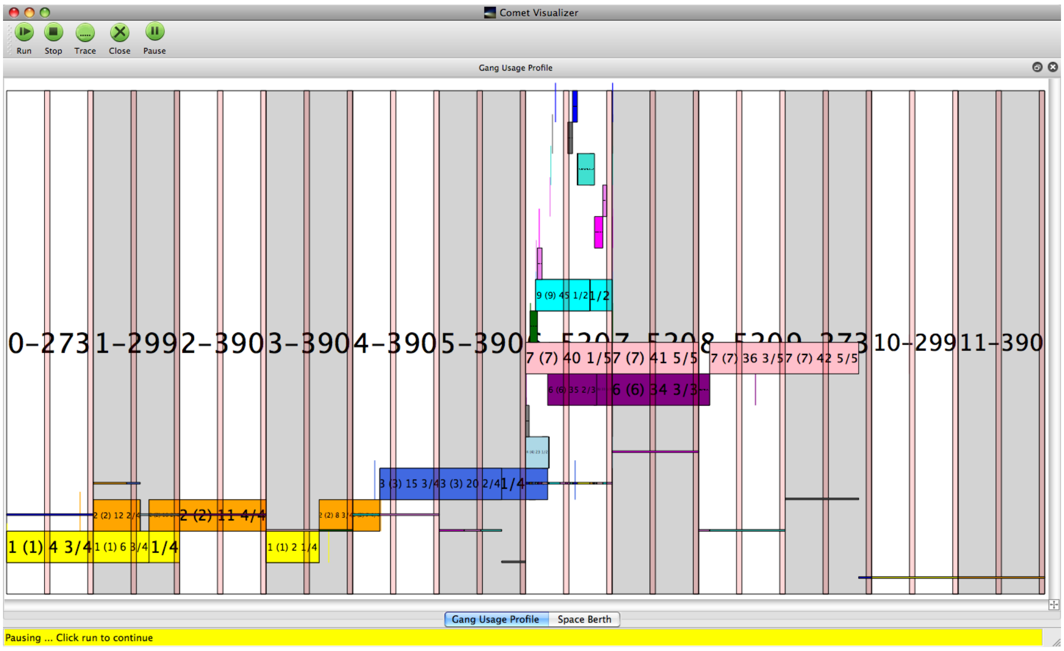


Figure 6.7: Workforce allocation for the 15 vessels example

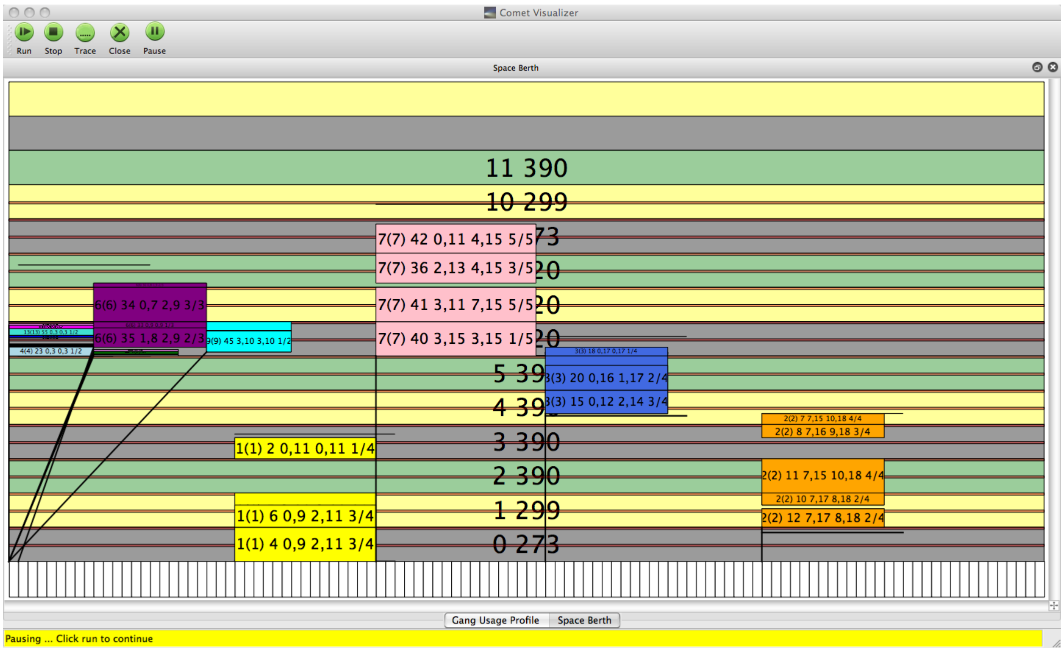


Figure 6.8: Vessel positions along the quay for the 15 vessels example

Figure 6.9 represents a percentile cumulative increase of the operational or MIP distance when adding additional constraints.

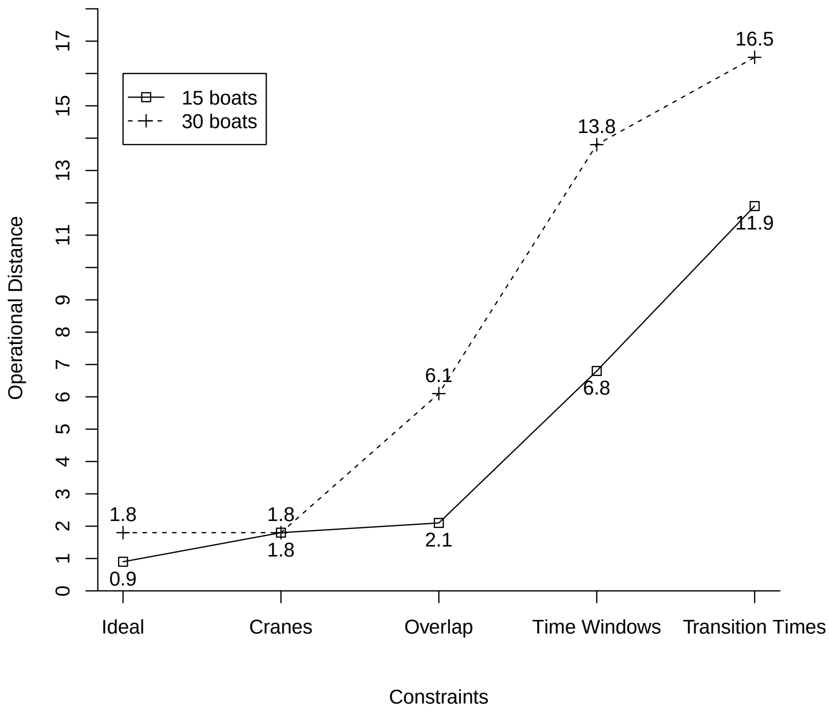


Figure 6.9: Evolution of operational distance when additional constraints are taken into account

6.11 Conclusion

In this paper, we have shown that taking into account critical operational and realistic constraints (crane transition times, variable labor costs, vessel setup times) for the BAPCAP can be done using a constraint programming approach. This proposed CP approach is modular in the sense that each set of operational constraints can be separated. The key idea is to take the gang allocation process as the main component and view it as a resource. Other side constraints can be integrated around this basic model. Experiments show that the CP model can produce solutions close to 1/5th and 1/10th from the lower bound MIP model having no operational constraints.

Future research includes using alternative heuristics centered on the profile or additional LNS procedures. The resource view of the model opens the possibility to use many scheduling tools from the OR/CP community to improve the performances or to integrate new types of side constraints.

Bibliography

[1] Bierwirth C., Meisel F. *A survey of berth allocation and quay crane scheduling problems in container terminals.* European Journal of Operational Research 202 (2010) p615..627

[2] Buhrkala K., Zuglianb S., Ropkeb S., Larsenc J., Lusb R. *Models for the Discrete Berth Allocation Problem: A Computational Comparison.* Transportation Research Part E Volume 47 (2011) p461..473

[3] Chang D., Jiang Z., Yan W., He J. *Integrating berth allocation and quay crane assignments.* Transportation Research Part E 46 (2010) p975..990

[4] Chazelle B. *The bottomn-left bin-packing heuristic: an efficient implementation.* IEEE Transactions on Computers 32 (1983) p697..707

[5] Cheong C.Y., Tan K.C., Liu D.K., Lin C.J. *Multi-objective and prioritized berth allocation in container ports.* Annals of Operations Research 180 (2010) p63..103

[6] Cordeau J.F., Laporte G., Legato P., Moccia L. *Models and tabu search heuristics for the berth-allocation problem.* Transportation Science 39 (2005) p526..538

[7] Giallombardo G., Moccia L., Salani M., Vacca I. *Modeling and solving the Tactical Berth Allocation Problem.* Transportation Research Part B 44 (2010) p232..245

[8] Han X.-L., Lu Z.-Q., Xi L.-F. *A proactive approach for simultaneous berth and quay crane scheduling problem with stochastic arrival and handling time.* European Journal of Operational Research 207 (2010) p1327..1340

[9] Imai A., Nishimura E., Papadimitriou S. *The dynamic berth allocation problem for a container port.* Transportation Research Part B 35 (2001) p401..417

[10] Imai A., Nishimura E., Hattori M., Papadimitriou S. *Berth allocation at indented berths for mega-containerships.* European Journal of Operational Research 179 (2007) p579..593

[11] Imai A., Chen H.C., Nishimura E., Papadimitriou S. *The simultaneous berth and quay crane allocation problem.* Transportation Research Part E 44 (2008) p900..920

[12] Kim K.H., Moon K.C. *Berth scheduling by simulated annealing.* Transportation Research Part B 37 (2003) p541..560

[13] Lee D.H., Chen J.H., Cao J.X. *The continuous Berth Allocation Problem: A Greedy Randomized Adaptive Search Solution.* Transportation Research Part E 46 (2010) p1017..1029

[14] Liang C., Huang Y., Yang Y. *A quay crane dynamic scheduling problem by hybrid evolutionary algorithm for berth allocation planning.* Computers and Industrial Engineering 56 (2009) p1021..1028

[15] Lokuge P., Alahakoon D. *Improving the adaptability in automated vessel scheduling in container ports using intelligent software agents.* European Journal of Operational Research 177 (2007) p1985..2015

[16] Meisel F., Bierwirth C. *Integration of berth allocation and crane assignment to improve the resource utilization at a seaport container terminal .* Operations Research Proceedings vol. 2005 no. IV (2006) p105..110

[17] Meisel F., Bierwirth C. *Heuristics for the integration of crane productivity in the berth allocation problem.* Transportation Research Part E 45 (2009) p196..209

[18] Moorthy R., Teo C.-P. *Berth management in container terminal: the template design problem.* OR Spectrum 28 (2006) p495..518

[19] Park Y.-M.and Kim K. H. *A scheduling method for Berth and Quay cranes.* OR Spectrum 25 (2003) p1..23

[20] Raa B., Dullaert W., Van Schaeren R. *An enriched model for the integrated berth allocation and quay crane assignment problem.* Expert Systems with Applications Volume 38 Issue 11 (2011) p14136..14147

[21] Seyedalizadeh Ganji S. R. , Babazadeh A., Arabshahi N. *Analysis of the continuous berth allocation problem in container ports using a genetic algorithm.* Journal of Marine Science and Technology (2010) 15 p408..416

[22] Shaw, P. *Using constraint programming and local search methods to solve vehicle routing problems* Proceeding of the International conference on principles and practice of constraint programming, Pisa vol. 1520 (1998) p340..354

[23] Tobias Achterberg, Timo Berthold, Thorsten Koch, Kati Wolter *Constraint Integer Programming: a New Approach to Integrate CP and MIP* Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems, CPAIOR 2008, LNCS 5015 (2008)

[24] Vacca I., Salani M., Bierlaire M. *Optimization of operations in container terminals: hierarchical vs integrated approaches.* http://transp-or2.epfl.ch

[25] Wang F., Lim A. *A stochastic beam search for the berth allocation problem.* Decision Support Systems 42 (2007) p2186..2196

[26] Zhang C., Zheng L., Zhang Z., Shi L., Armstrong A.J. *The allocation of berths and quay cranes by using a sub-gradient optimization technique.* Computers and Industrial Engineering 58 (2010) p40..50

Chapter 7

Tank allocation for liquid bulk vessels using a hybrid constraint programming approach

Authors:
Rowan Van Schaeren, *Antwerp Maritime Academy*, rowan.van.schaeren@hzs.be
Pierre Schaus, *n-Side*, pierre.schaus@n-side.com
Wout Dullaert, *University of Antwerp, ITMMA, Antwerp Maritime Academy*, wout.dullaert@ua.ac.be
Birger Raa, *Ghent University*, birger.raa@ugent.be
Pascal Van Hentenryck, *Brown University*, pvh@cs.brown.edu

7.1 Abstract

This papers considers the allocation of cargoes to the tanks of chemical liquid bulk vessels. Currently, no journal article or commercial software is capable of handling the multitude of side constraints that need to be considered from a practical point of view. These constraints include a.o. segregation constraints for the chemicals and detailed vessel stability considerations that limit the volumes that can be loaded in the tanks. A hybrid CP-LP model is presented in which large neighborhood search (LNS) based on a constraint programming (CP) model is used to determine possible cargo-to-tank allocations, after which linear programming (LP) is used to determine the actual volumes being loaded such that the vessel stability is guaranteed. The validity and practical usefulness of this model is illustrated for three real-life problem instances which are fully disclosed to support further research.

Keywords: tank allocation problem, constraint programming, chemical tankers, load planning, stability

7.2 Introduction

The chemical industry is characterized by a very strong competitive environment [14]. This leads to increased pressure to provide consistent quality, fast delivery and cost-cuttings. Since chemicals are transported across the whole world, it is not surprising that the special, dedicated chemical tanker vessels form an important aspect of this liquid bulk chemicals trade. This transport segment is dominated worldwide by three key players: Stolt-Nielsen SA (69 vessels), Odfjell ASA (58 vessels) and Sovcomflot Group (46 vessels) [6]. The number of chemical tankers available on the market is steadily increasing. Figure 7.1 illustrates this increase over the last ten years, both in absolute numbers and in DWT (Deadweight Tonnage).

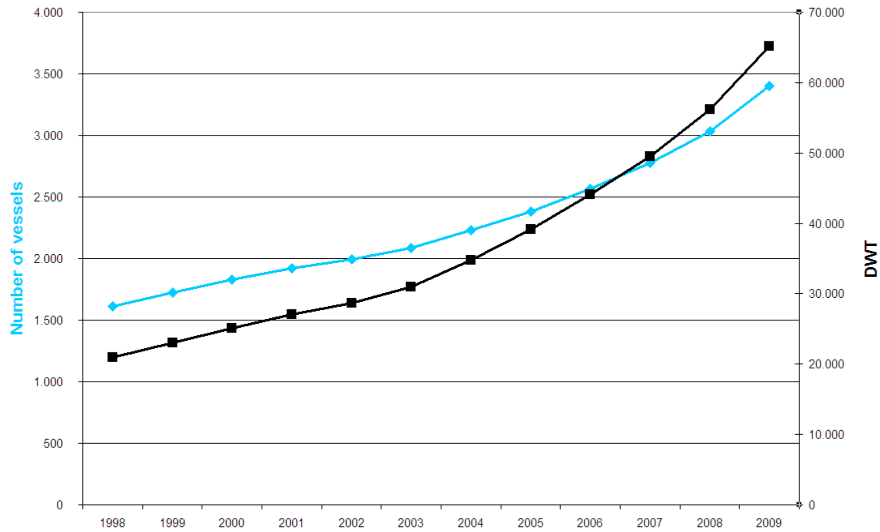


Figure 7.1: Chemical tanker fleet evolution (number of vessels and DWT) [6]

On average, chemical tanker vessels have a deadweight tonnage (DWT) of 19,000 and a length of 134 meters. This is considerably smaller than the 82,000 DWT and 208 meters of an average tanker [6]. The reason for this is the specialized nature of the cargo (shipment) and the port depths where these vessels have to berth. Chemical tankers also distinguish themselves from other tankers in the large number of separate tanks available to load cargo. Some chemical tankers have more than 30 individual cargo tanks. These large numbers of cargo tanks allow for many different cargoes to be transported simultaneously. Table 7.1 illustrates the average number of individual tanks that can be found on chemical tankers in function of their age and DWT.

	Vessel age (years)					Average	
DWT range (tonnes)	20+	15-19	10-14	5-9	0-4	Nr. tanks	DWT
1-4,999	11.5	10.6	10.7	10.5	10.1	10.7	2,808
5-9,999	14.7	16.2	15.7	13.9	12.4	14.6	7,282
10-19,999	18.8	18.7	19.8	18.2	14.7	18.0	14,929
20-29,999	22.3	19.2	27.7	22.4	19.0	22.1	25,569
30-39,999	28.9	29.2	30.3	16.4	13.3	23.6	36,369
40,000+	16.9	15.5	13.5	13.7	13.4	14.6	47,160
Total Average	18.9	18.2	19.6	15.9	13.8	17.3	22,353

Table 7.1: Average number of tanks in function of the vessel's age and DWT [6]

Each cargo tank needs to connect its own pump and piping system to the shore installation to prevent mixing or contaminating individual cargoes. This has a significant impact on the planning of cargoes on board these chemical tankers as cargo interactions can result in dangerous situations. Almost all chemical products can be considered dangerous one way or the other (being e.g. corrosive, marine pollutant, or toxic). These products must therefore be stored in accordance with stringent regulations, specifically with regard to segregation. Segregation is not only important between the different products themselves (certain products such as e.g. caustic soda and sulfuric acid cannot be stowed in adjacent tanks) but also with respect to the tank coatings that protect the tanks from the products stored in them. In addition to this, the vessel's stability constraints complicate the capacity planning even further.

Loading plans are generally generated manually by the vessel planners and then checked by a stability software program in order to ensure that it is safe for the vessel to sail. Because of the multitude of constraints, regulations and "good practices" it is obviously very difficult to generate high quality loading plans manually. Optimization methods capable of handling these side constraints and generating high quality solutions can therefore greatly support vessel planners and free up time for handling non-standard scheduling issues.

Academic literature on this tank allocation problem (TAP) [11] or operational planning is limited. Most of the existing research considers both the TAP and vessel routing of chemical tankers. However, only a few papers deal with segregation and stability constraints simultaneously in their TAP, even though these are essential in real-life applications.

Vouros et al. [18] propose a theoretical framework for the TAP of chemical product carriers. They propose to split the constraints into three categories: (i) stability and vessel structure; (ii) cargo allocation and (iii) cargo handling. Bausch et al. [4] present a decision support system for tanker scheduling where cargoes are not mixed (different cargoes are shipped in different tanks) and vessels can have up to 7 tanks. They consider both barges and small vessels.

Barbucha [2] proposes three approximation algorithms for the storage of dangerous cargoes taking into account segregation constraints. He suggests the use of two segregation matrices: one for the cargoes and one for the compartments. Both matrices are filled with distances. The cargo matrix represents the minimal distance required by two respective products and the compartment matrix represents the distances between the different compartments. By assigning a cost to each product for each individual compartment, the total cost of a loading plan can therefore be minimized. Vessel stability criteria are not considered in the suggested algorithms.

Jetlund et al. [12] propose a mixed integer linear programming (MILP) model for a chemical tanker fleet scheduling problem, where the tanker's loading capacity is limited to the number of tanks and the maximal carrying capacity of the vessel. It is assumed that the TAP is already addressed at another level. Neo et al. [13] extend this model of Jetlund et al. [12] to include vessel stability constraints, cargo loading and unloading, compartment cleaning requirements and draft limitations. They apply their model to two case studies to illustrate the significance of cargo compatibility and vessel compatibility. Their results show that calculation times become considerable when using mixed integer programming for routing and scheduling a chemical tanker when considering additional operational constraints.

Al-Khayyal et al. [1] also propose a MILP model for scheduling and routing liquid bulk vessels. They only take capacity constraints into consideration. They show that the problem is NP-hard and express the need for specialized algorithms. Christiansen et al. [5] propose various models for scheduling problems in industrial and tramp shipping. Their models cover vessels with full shiploads, multiple cargoes with fixed cargo size, multiple cargoes with flexible cargo size, multiple products and optional cargoes. Their TAP also considers only capacity constraints.

Hvattum et al. [11] present a model to determine whether a given route is feasible for a given vessel carrying bulk cargoes in tanks, and show that it is NP-complete. As an illustration, the model is tested on several randomly generated instances for two different vessel sizes. They use simplified stability constraints based on evidence presented by Pintens [16] and consider several objective functions: (i) minimizing operating costs of the vessel considering fuel consumption whereby the problem is reduced to a pure routing problem with the TAP being limited to feasibility checking; (ii) minimizing the costs and inconvenience of tank cleaning; (iii) maximizing the number of unused cargo tanks in order to be more flexible in the subsequent ports of the vessel's route concerning the loading of additional cargo.

The above literature review illustrates the difficulties of simultaneously addressing both the TAP and vessel routing aspects for chemical vessels even if no or only simplified vessel stability constraints are taken into account.

This paper focuses on the Tank Allocation Problem and has a twofold contribution. First, the stability of chemical tankers is modeled in full detail, by further extending constraints described in Hvattum et al. [11]. Second, an elegant hybrid solution approach is proposed that combines (i) large neighborhood search (LNS) based on a constraint programming (CP) model with (ii) linear programming (LP) for optimizing vessel stability.

The paper is further organized as follows: Section 7.3 details the vessel stability criteria that should be considered when approaching the tank allocation problem. The proposed model is presented in Section 7.4 and is implemented in COMET (www.dynadec.com). Section 7.5 presents the computational experiments. Detailed vessel data along with three datasets from a leading chemical tanker company are given in full detail to support further research on this topic. Computational results together with the consequences of the stability constraints on the values of the objective function conclude this section. Conclusions and suggestions for further research are given in Section 7.6.

7.3 Tank allocation and vessel stability

As mentioned above, this paper focuses on the TAP of a chemical tanker where all cargoes (shipments) are loaded in one port and discharged in another port. This means that vessel routing is not considered here, but is left as an avenue for further research.

This section presents the specific tank allocation problem for chemical vessels, including constraints that are simplified or neglected in the existing papers. These constraints can be classified into the following three categories discussed below: (i) segregation of cargo, (ii) cargo-tank compatibility, and (iii) vessel stability.

7.3.1 Cargo segregation

Barbucha [2] describes the cargo segregation constraint in terms of distance requirements between dangerous cargoes. For chemical tankers, all the cargo on board is liquid bulk, so the distance between any pair of cargoes can be reduced to the following two possibilities:

- There are no specific segregation requirements for both cargoes, so they can be stored in adjacent tanks.
- The two cargoes can interact when coming into contact with each other and pose a risk, so they cannot be stored in adjacent cargo tanks except when these tanks are separated by a watertight bulkhead.

It is also assumed that cargoes can never be mixed together in a single cargo tank. For the segregation of the cargo, a matrix is built indicating that the considered cargoes have specific requirements or not. An example of such a cargo segregation matrix is shown in Figure 7.2 with 'X' indicating that specific requirements need to be considered.

Cargoes	1	2	3	4
1				
2	X			
3	X			
4		X		

Figure 7.2: Cargo segregation matrix

These specific requirements are obtained by consulting the International Maritime Dangerous Goods Code (IMDG) segregation matrix (see Appendix) using the segregation ID of each cargo (e.g. Table 7.8). When constructing this segregation matrix, not only the chemical interactions between the different cargoes need to be considered but also the temperature at which they need to be transported. E.g., it is possible that two cargoes have no specific requirements concerning their chemical characteristics but that the first cargo must be transported under ambient conditions (i.e. 18°C) and that the second cargo must be heated up to 40°C during transport. If these two cargoes would be in adjacent tanks, part of the second cargo might solidify due to cooling off by the first cargo or part of the first cargo may become chemically unstable due to heating of the second cargo. Segregation of these cargoes is therefore necessary because of this temperature issue.

7.3.2 Cargo-tank compatibility

These constraints represent the compatibility between the tanks and the cargoes loaded into them as discussed by Jetlund et al. [12]. We propose to classify tank compatibility requirements into the following four categories:

- Temperature: Some of the chemicals transported need to have their temperature managed (e.g. tallow needs to be transported at 75°C). In order to heat the cargo, the tank needs to be equipped with a heating system. On modern chemical tankers almost all tanks have heating capabilities. Next, if heating is required, one must also consider which heating medium is used to heat the cargo (e.g. water/steam, or thermal oil). A third temperature consideration is the location of the cargo tank in relation to ballast tanks.

If cargo that needs to be heated is allocated to a wing tank next to a ballast tank, some of the cargo may cool down too much because of the lower temperature of the ballast water. Allocating the cargo to a tank which is also heated would therefore be preferred.

- Tank material: When storing chemicals in a tank, one must ascertain whether the material of the tank is resistant to that chemical. Most of the modern tanks are made of stainless steel which can accommodate most of the chemicals transported. However, tankers with coatings such as epoxy and zinc could be damaged by certain cargoes (e.g. hydrochloric acid or tallow). Care must be taken that the chemicals in the tank do not damage the tanks or their coatings.

- Previous cargoes: As cargo is planned in a certain tank, it is also important to check if the tank may still be contaminated by previous cargoes. Another constraint concerning previous cargoes in a tank is the fact that some cargoes may not be loaded several times consecutively in the same tank because of the danger of impregnating the tank's walls.

- Tank structure: When storing a chemical in a cargo tank, the structural conditions of the tank must also be considered. The structural integrity of the tank allows only for a certain maximum mass to be loaded. This mass is calculated using the volume and density of the cargo to be loaded in that particular tank.

7.3.3 Vessel stability

Vessel stability constraints describe conditions to be met that prevent a vessel from capsizing or breaking when at sea. Detailed information can be found in Derrett et al. [8]. Six types of stability constraints can be distinguished. To the best of our knowledge, we are the first to take all of these into account in our TAP model which significantly increases its value for industry professionals. One of the most comprehensive models by Neo et al. [13], only considers the first three.

- Maximum Trim (Figure 7.3): This is the maximum difference between the drafts fore and aft. The draft aft must always be larger than or equal to the draft fore, otherwise the vessel would experience more resistance cutting through the water. The trim also needs to be positive because in most cargo tanks the pumps are located at the back of the tank. A negative trim would make it hard to completely empty the tanks using the cargo tank pumps.

- Maximum draft (Figure 7.4): Every vessel has a maximum draft (d_{max}) or immersion in the water to ensure that the vessel is not overloaded and that enough freeboard remains for safe navigation and to cope with heavy weather. The maximum immersion in the water could also be restricted

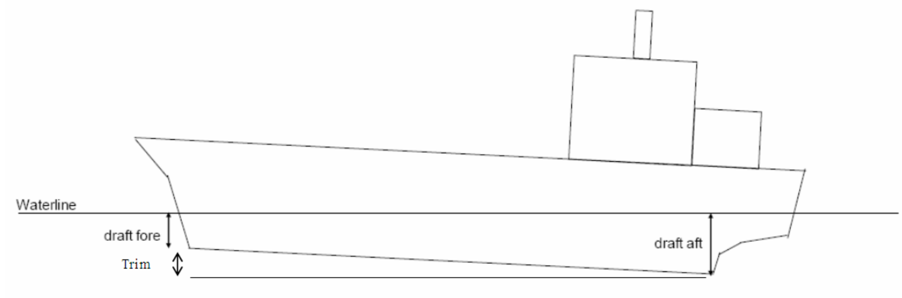


Figure 7.3: Schematic representation of the maximum trim constraint

by the maximum water depth in a port as the vessel would otherwise run aground.

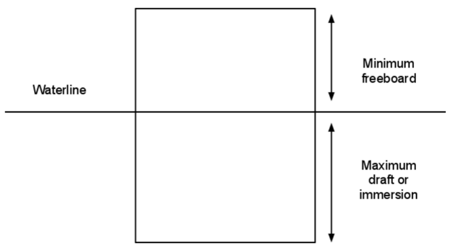


Figure 7.4: Schematic representation of the maximum draft

- Maximum heel or list (Figure 7.5): The maximum list represents the maximum inclination a vessel can have to port (left or negative) or starboard (right or positive). The list is represented as the angle θ. Too high a list would make the vessel hard to navigate and would make it harder to service all the tanks as liquids would accumulate at the sides. It is therefore industry practice to plan a liquid bulk vessel with a list of zero degrees.

- The metacentric height or GM (Figure 7.6): This is the distance between the center of gravity of a vessel (G) and its metacenter (M). The larger the GM, the quicker a vessel will come back to its vertical position when pushed over by an external force like the wind or the waves. Each time before leaving a port this GM must be calculated and must be at least 15 centimeters in accordance with international regulations.

- Shear forces (SF) and Bending moments (BM): SFs and BMs result from all the up- and downward forces affecting the vessel and are measured at the so-called frames of the vessel. These frames are transversal reinforcements

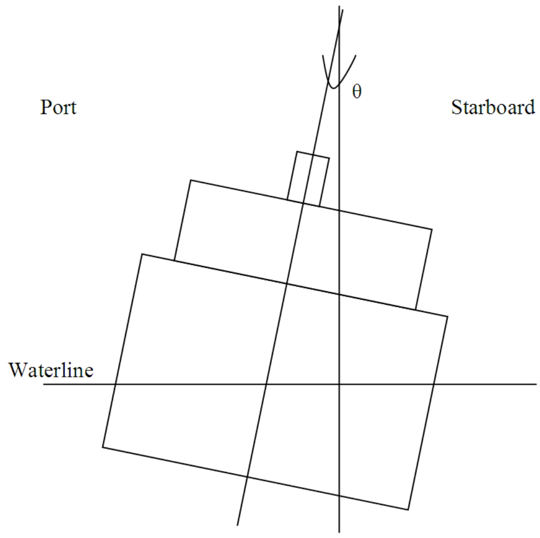


Figure 7.5: Representation of the list of a vessel

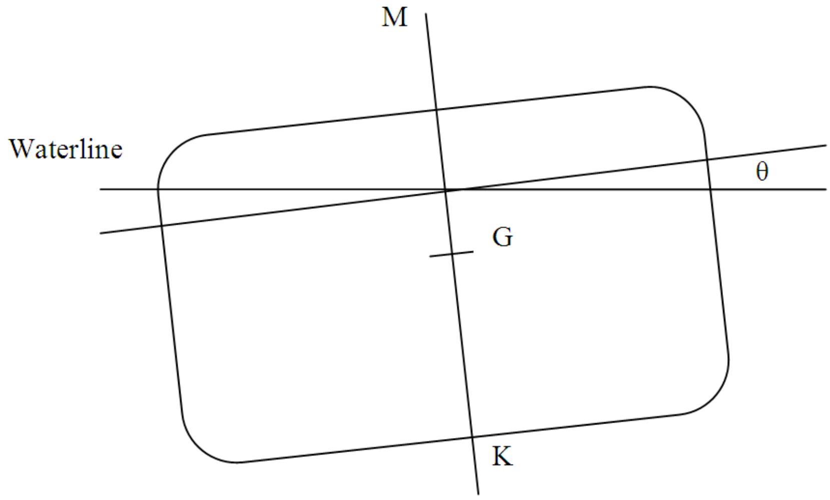


Figure 7.6: Schematic representation of the metacentric height GM

that strengthen the vessel and divide it into compartments. When a vessel violates its BM and SF constraints, it is not allowed to sail because the vessel might actually break.

- Sloshing: A non-empty tank must either be filled below a given lower threshold level or above a given upper threshold level to avoid excessive sloshing of the liquids in the tank during sailing. Hvattum et al. [11] simplify this constraint and only consider the upper threshold level.

7.4 Solution approach

The tank allocation problem deals with allocating a given set of cargoes C to the set of available tanks T on a vessel such that the free space on the vessel is maximized while all of the constraints discussed in the previous section are satisfied.

The objective of maximizing the free space is taken from [11] and is justified as more free space leads to more flexibility for loading additional cargo in the vessel's port of arrival. This free space can be represented by the total capacity in the unused tanks or by the number of unused tanks. These two are not necessarily the same as the tanks may have different dimensions. In the remainder of this paper, we adopt the first alternative, but the model and solution approach can very easily be adjusted to adopt the second alternative, or even other objective functions.

To solve this tank allocation problem, a hybrid approach is presented that combines constraint programming (CP) and linear programming (LP).

Constraint programming is used to assign cargoes to the vessel's tanks taking into account the first two categories of constraints (cargo segregation and cargo tank compatibilities) whilst maximizing the amount of free space left. Each time CP finds a solution, an LP is solved that determines the amount of cargo to be put in the allocated tanks such that the third category of constraints, i.e. the vessel's stability criteria, are satisfied. In other words, the CP part iteratively searches for alternative tank-to-cargo assignments, which are subsequently validated in the LP part. Both parts of the solution approach are described in detail below.

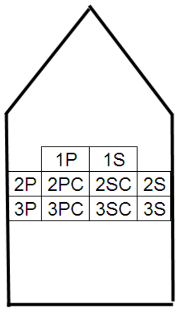
7.4.1 The CP model

The CP model is implemented in Comet (see Listing 7.1). In this model, Cap_t represents the capacity of cargo tank t (in m^3), Vol_c represents the volume of cargo c to be loaded (in m^3), and the variables are defined as follows.

- $cargo_t$: represents the type of cargo assigned to cargo tank t. The domain of $cargo_t$ contains only those cargoes that can be placed into that specific cargo tank, as derived from the segregation matrix, and a dummy cargo 0 denoting that the tank remains empty.

  Figure 7.7 illustrates the structure of a segregation matrix with 'X' indicating whether a certain product can be stored in a certain tank.

- $load_c$: represents the total capacity of tanks allocated to cargo c (in m^3). The minimum value of $load_c$ is set to Vol_c, the volume of cargo c. For dummy cargo 0, this minimum is $Vol_0 = 0$.



Cargo Tanks	Cargoes 1	2	3	4
1P		X		X
1S		X	X	X
1C		X		X
2P	X	X		X
2S	X	X	X	X
2PC	X	X		X
2SC	X	X		X
3P		X		X
3S		X	X	X
3PC		X		X
3SC		X		X

Figure 7.7: Cargo and tank compatibility example

Assigning cargo to tanks is handled by the COMET 'multiknapsack' global constraint that expresses the volume requirements of each cargo. This global constraint enforces the following relation, linking the two sets of variables $cargo_t$, $load_c$ and the tank capacities Cap_t:

$$load_c = \sum_{t \in T} Cap_t \cdot (cargo_t = c) \;, \forall c$$

For chemical tankers, the segregation constraints state that incompatible cargoes cannot be stored in adjacent tanks. With $\mathcal{A} \subset T \times T$ the set of pairs of adjacent tanks ($adjTanks$ in Listing 7.1) and $\mathcal{C} \subset C \times C$ the set of pairs of cargoes which are compatible ($possComb$ in Listing 7.1), the segregation constraint is as follows:

$$(cargo_i, cargo_j) \in \mathcal{C}, \quad \forall (i,j) \in \mathcal{A}$$

This is handled by the COMET 'table' global constraint.

As mentioned above, the objective of the CP model is the maximization of the total unused capacity:

$$\max load_0 = \sum_{t \in T} Cap_t \cdot (cargo_t = 0)$$

The skeleton of the COMET model is given in Listing 7.1.

7.4.2 Large Neighborhood Search

To solve the CP model presented above, a Large Neighborhood Search (LNS) [17] is used instead of an exhaustive CP search. LNS combines the expressiveness of Constraint Programming and the speed of Local Search, and has been used successfully on various combinatorial optimization problems (see for instance [7, 9, 10, 15, 17]). Like most local search approaches, LNS maintains a current best solution. At each restart of the LNS, a neighborhood of the solution is explored

Listing 7.1: Comet Model

```
set{int} adjTanks[T] = [{2,3,5},{1,4}, ...];
int possComb [1..nbComb,1..2] = [[1,4],[1,6], ... ];
Solver<CP> cp();
/* specify possible cargoes in each tank */
var<CP>{int} cargo[t in T](cp, possCargo[t]);
/* load variables */
var<CP>{int} load[c in C](cp, Vol[c]..bigM);
maximize<cp>
   sum(t in T) (cargo[t]==0)*Cap[t]
subject to {
  cp.post(multiknapsack(cargo,Cap,load));
  forall(t1 in T,t2 in adjTanks[t1]: t2>t1)
     cp.post(table(cargo[t1],cargo[t2],possComb));
} using {
  labelFF(cargo);
  if(!solveStability())
    cp.fail();
}
```

with CP in an attempt to find a better solution. If such a solution is found, the current best solution is updated. The LNS neighborhood is obtained by relaxing a subset (called fragment) of the variables to their original domain. The rest of the variables are fixed to their value in the current solution. LNS thus consists of the repetition of the following two steps until a stopping criterion is met (like a timeout limit or maximum number of iterations):

1. Neighborhood definition: choosing the fragment of variables that will be relaxed to their original domains.

2. Neighborhood exploration: using CP to explore the restricted problem defined by the relaxation of the fragment. A limit on the number of failures is specified to avoid spending too much time exploring the neighborhood.

LNS requires only three simple parameters:

1. The size of the fragment: typically a percentage of the number of variables (usually 5 to 20%),

2. The fragment selection procedure, which is typically a random selection, and

3. The limit on the CP exploration step: typically a limit on computation time or the number of backtracks per restart.

The main advantages of LNS over classical local search techniques are twofold. First, LNS does not require the design of potentially complicated local moves (such as swaps, exchanges, . . .) as it is able to achieve arbitrarily complex moves. Second, there is no need for sophisticated techniques to escape local optima (as e.g. in Simulated Annealing or Tabu Search) since large neighborhoods are explored with CP.

7.4.3 The LP model

As soon as the CP search finds an assigment of tanks to each cargo that satisfies the segregation constraints, the LP part is called upon to check if the stability conditions can be met ('solveStability()' in Listing 7.1). In this sense the LP acts as a final constraint checker performed in the leaf nodes of the CP search tree.

Although the CP solution allocates sufficient volume to each cargo, the stability conditions (checked by the LP model) may prevent some of the allocated tanks from being completely filled with the designated cargo, which could make it impossible to load the required volume of that cargo and renders the solution infeasible.

The first four stability conditions (trim, list, draft and metacentric height) and the sixth (sloshing) are implemented as constraints in the LP model, whereas the fifth condition (bending moments and shear forces) is dealt with in the objective function.

In the LP model, the following sets and parameters are used. An illustrative dataset is presented below in Tables 7.2, 7.4, and 7.5.

$c \in C$:	the set of cargoes to be planned
$t \in T$:	the set of tanks
$f \in F$:	the set of (transversal watertight) frames of the vessel
$i \in E$:	the set of constants and consumables on board the 'empty' vessel
Vol_c:	volume of cargo c that needs to be planned (m^3)
δ_c:	density of cargo c (ton/m^3)
LCG_t:	longitudinal center of gravity of tank t (m)
TCG_t:	transversal center of gravity of tank t (m)
VCG_t:	vertical center of gravity of tank t (m)
I_t:	inertia moment of tank t (m^4)
Cap_t:	capacity of tank t (m^3)
$cargo_t$:	cargo that is planned in tank t (this is a variable in the CP part that is an input parameter for the LP part)
$Frame_t$:	frame to which tank t belongs
α_t:	lower threshold level for tank t
β_t:	upper threshold level for tank t
W_i:	weight of constant or consumable i (ton)
δ_i:	density of constant or consumables i (ton/m^3)
LCG_i:	longitudinal center of gravity of constant or consumables i (m)
TCG_i:	transversal center of gravity of constant or consumable i (m)
VCG_i:	vertical center of gravity of constant or consumables i (m)
I_i:	inertia moment of constant or consumables i (m^4)
Δ:	total weight of the vessel (ton)
$MCTM$:	the moment to change trim of the vessel one meter ($ton \cdot m$)
LCB:	longitudinal center of buoyancy of the vessel (m)
t_{max}:	maximum trim of the vessel (m)
$Draft$:	mean draft of the vessel (m)
d_{max}:	maximum draft of the vessel (m)
KM:	metacenter of the vessel (m)
GM_{min}:	minimum metacentric height of the vessel (m)
GM_{max}:	maximum metacentric height of the vessel (m)
W_f:	structural weight of the vessel at frame f (not including cargo) (ton)
S_f:	under water surface of the hull at frame f (m^2)
δ_0:	density of the water in which the vessel finds itself (ton/m^3)
sf_{max}:	maximum shear force on frame f (ton)

The variables used in the model are the following:

$fill_t$:	weight of the cargo planned in cargo tank t (ton)
$trim$:	longitudinal inclination of the vessel (cm)
GM:	metacentric height of the vessel (m)
sf_f:	absolute value of the weight at frame f that generates the shear force (ton)

X_t: binary variable, 0 if tank t is empty or filled below the lower threshold level, 1 if tank t is filled above the upper threshold level

$$\min Z = \sum_{f \in F} sf_f \tag{7.1}$$

$$\text{s.t.} \forall c \in C: \quad \sum_{t \in T | cargo_t == c} fill_t / \delta_c = Vol_c \tag{7.2}$$

$$trim = \frac{1}{MCTM} \left(\sum_{i \in E} W_i LCG_i + \sum_{t \in T} fill_t LCG_t - \Delta LCB \right) \tag{7.3}$$

$$0 \leq trim \leq t_{max} \tag{7.4}$$

$$\frac{1}{\Delta} \left(\sum_{i \in E} W_i TCG_i + \sum_{t \in T} fill_t TCG_t \right) = 0 \tag{7.5}$$

$$Draft + trim/2 \leq d_{max} \tag{7.6}$$

$$\begin{aligned} GM = KM - \frac{1}{\Delta} & \left(\sum_{i \in E} W_i VCG_i + \sum_{t \in T} fill_t VCG_t \right) \\ & + \frac{1}{\Delta} \left(\sum_{i \in E} \delta_i I_i + \sum_{t \in T} \delta_{cargo_t} I_t) \right) \end{aligned} \tag{7.7}$$

$$GM_{min} \leq GM \leq GM_{max} \tag{7.8}$$

$$\forall f \in F: \quad sf_f \geq W_f + \sum_{t \in T | Frame_t == f} fill_t - \delta_0 S_f Draft \tag{7.9}$$

$$\forall f \in F: \quad sf_f \geq -W_f - \sum_{t \in T | Frame_t == f} fill_t + \delta_0 S_f Draft \tag{7.10}$$

$$\forall f \in F: \quad sf_f \leq sf_{max} \tag{7.11}$$

$$\forall t \in T: \quad \frac{fill_t}{\delta_{cargo_t}} \leq \alpha_t (1 - X_t) + Cap_t X_t \tag{7.12}$$

$$\forall t \in T: \quad \frac{fill_t}{\delta_{cargo_t}} \geq \beta_t X_t \tag{7.13}$$

$$\forall t \in T: \quad fill_t \geq 0, X_t \in \{0, 1\} \tag{7.14}$$

Constraint (7.2) ensures that all cargo is effectively loaded.

To calculate the maximal draft of a vessel, the mean draft and the trim are needed (see also Figure 7.3). The mean draft can be found using the total weight Δ in the hydrostatic tables of the vessel (see e.g. Table 7.4). Each of the weights on board (constants and consumables, as well as cargo) generates a downward moment (given by multiplying its weight with its longitudinal center of gravity). On the other hand, the displaced water (whose weight equals Δ) generates an upward moment in the center of buoyancy. By dividing the resulting moment with $MCTM$ (the moment needed to change trim one meter), the corresponding trim is obtained. This trim must be between 0 and t_{max} (Constraint 7.4), and the maximum draft (i.e. mean draft plus half of the trim) cannot be greater than d_{max} (Constraint (7.6)).

Constraint (7.5) calculates the average transversal center of gravity of all weights on board, which causes the heel or list, and imposes that it is zero.

The metacentric height GM is calculated in Constraint (7.7) (see Figure 7.6). The metacenter KM (first term of the right hand side) is a vessel specific parameter based on the water displacement that can be found in the stability booklet of the vessel (see e.g. Table 7.4). The difference (distance) between KM and GM is caused by the average vertical center of gravity of all weights on board (second term of the right hand side), and by the free surface effect, caused by the inertia of each element on board (third term of the right hand side). The metacentric height GM must be between GM_{min} and GM_{max} (Constraint (7.8)).

Constraints (7.9) and (7.10) represent the calculation of the weights sf_f that generate the downward and upward forces at frame f. The downward forces at frame f are caused by the sum of the structural weights of the frame and the weights of the cargoes in the frame's tanks. The upward force is caused by the displaced water, whose weight is approximated by multiplying the underwater surface of the frame with the draft of the vessel and the density of the water. The resulting weight must remain between $-sf_{max}$ and $+sf_{max}$ ton (Constraint(7.11)).

Constraints (7.12) and (7.13) prevent sloshing by imposing that a tank is either filled between 0 and α_t (when $X_t = 0$) or between β_t and Cap_t (when $X_t = 1$).

The objective function (7.1) minimizes the absolute values of the shear forces. The exact formula for calculating the shear forces is nonlinear, but a linear approximation of the shear forces is used, which was validated by Pintens [16]. When the shear forces are minimized, the bending moments will also be minimized as they are dependent on each other. Because shear forces can be positive or negative, it is their absolute values that are being minimized.

When a feasible solution is found by the LP part, the bending moments are calculated by integrating the shear forces across the frames along the length of the vessel and verified. If the absolute values of the bending moments are all below the (vessel specific) maximum BM_{max}, the LP solution is accepted. Otherwise, the LP solution is discarded and the leaf of the CP search tree is still declared infeasible.

7.5 Computational results

To show the validity and practical relevance of the suggested model and solution approach, this section reports on two sets of experiments. In the first, the approach is applied to three real-life datasets, while the second experiment illustrates the consequences of incorporating the complete set of stability constraints.

7.5.1 Vessel details

This section introduces all vessel specifics which are necessary to model the stability constraints. Consider the following vessel with 28 cargo-tanks and a total capacity of $19837m^3$. Table 7.2 gives a description of the 28 cargo-tanks and their characteristics. Given the limited size of the tanks, sloshing effects are minimal and hence non-critical. Therefore, sloshing is not considered by setting $\alpha_t = Cap_t$ and $\beta_t = 0$ for all tanks.

t	$Cap_t(m^3)$	$VCG_t(m)$	$LCG_t(m)$	$TCG_t(m)$	$I_t(m^4)$
1	680	7.866	121.295	-2.92	233.4
2	674	7.856	121.255	2.99	227.5
3	949	7.59	111.408	-4.08	490.5
4	949	7.595	111.413	4.09	488.2
5	316	7.488	102.202	-7.22	40.4
6	420	7.516	102.296	-2.69	40.4
7	431	7.516	102.057	-2.7	93.2
8	316	7.488	102.057	7.21	94.3
9	833	7.487	95	-5.1	450
10	846	7.497	94.9	-5	450
11	370	7.487	87.788	-7.67	62.4
12	428	7.495	87.778	-2.64	62.4
13	429	7.495	87.815	2.58	93.4
14	370	7.488	87.815	7.67	98.2
15	853	7.488	75.506	-7.66	146.2
16	991	7.494	75.505	-2.67	146.2
17	991	7.494	75.504	2.68	223.3
18	853	7.488	75.504	7.66	221.7
19	792	7.488	63.296	-5.14	440
20	803	7.49	63.191	-5.15	446
21	545	7.488	54.672	-7.69	93.5
22	626	7.494	54.682	-2.65	93.5
23	627	7.494	54.728	2.61	138.2
24	545	7.488	54.728	7.69	142.7
25	1083	7.6	44.024	-5.15	822
26	1083	7.683	44.078	5.16	822
27	1017	7.967	33.658	-4.81	822.2
28	1017	7.967	33.658	4.82	817.7

Table 7.2: Cargo tank characteristics of the proposed vessel

Figure 7.8 depicts the vessel and the location of its tanks. It can be used to derive the tank adjacency table (Table 7.3). When an apparent adjacency is not mentioned (e.g. between cargo tanks 3 and 5), this means that there is a void space segregating the tanks.

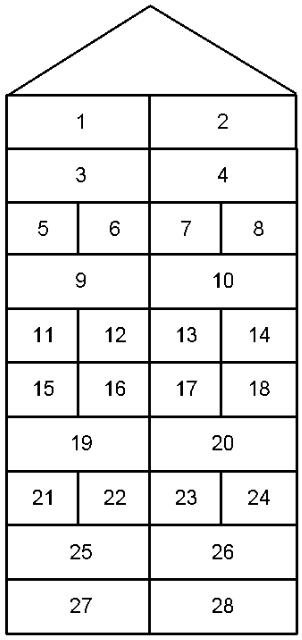


Figure 7.8: Adjacency of the vessel's 28 cargo tanks

Next, Table 7.4 gives the hydrostatic data of the proposed vessel for various total weights (or displacements) Δ.

When calculating the stability of a vessel, all of the weights on board apart from the cargoes being loaded also need to be taken into account. This is the set E of so-called constants (lightship) and consumables (e.g. fresh water, fuel) represented in Table 7.5.

To calculate the shear forces and the bending moments, the positions of the frames are required. These are given, together with the weights and surfaces of the frames, in Table 7.6.

7.5.2 Model validation

The proposed model is validated using three real-life datasets provided by one of the largest liquid chemical shipping companies in the world. A summary of the three datasets is given in Table 7.7. Full details of each dataset are given in Tables 7.8, 7.9 and 7.10.

For each dataset, the model was run with a maximum number of restarts ranging from 1000 to 5000 and the following LNS settings (see also section 7.4.2):

- fragment size: 10%

t	Adjacent tanks				
1	2	3			
2	1	4			
3	1	4			
4	2	3			
5	6	9			
6	5	7	9		
7	6	8	10		
8	7	10			
9	5	6	10	11	12
10	7	8	9	13	14
11	9	12			
12	9	11	13		
13	10	12	14		
14	10	13			
15	16				
16	15	17			
17	16	18			
18	17				
19	20	21	22		
20	19	23	24		
21	19	22			
22	19	21	23		
23	20	22	24		
24	20	23			
25	26				
26	25				
27	28				
28	27				

Table 7.3: Adjacency of the vessel's 28 cargo tanks

- fragment selection: random
- limit of the CP exploration step: 2000 backtracks

Each instance was run 100 times and the average amount of free space together with the average runtime was recorded. Results are represented in Figures 7.9, 7.10 and 7.11 for datasets 1, 2 and 3 respectively and also detailed in Table 7.11. In general, it can be noted that increasing the number of restarts improves solution quality at an acceptable increase of computation time. The impact of increasing the number of restarts on solution quality tapers off and - due to the randomness of the LNS - no longer guarantees a higher solution quality (see e.g. Dataset 2 for 4000 and 5000 restarts). The practical validity of the results has been confirmed

$\Delta(ton)$	$Draft(m)$	$LCB(m)$	$KM(m)$	$MCTM(ton \cdot m)$
7000	3	74	16	1.80
8000	3.5	74	14	1.90
9000	4	74	13	1.95
10000	4.5	73	12	2.00
11000	5	73	12	2.05
12000	5	73	11	2.10
13000	5.5	73	11	2.20
14000	6	73	11	2.30
15000	6.5	73	10	2.35
16000	6.5	72	10	2.40
17000	7	72	10	2.50
18000	7.5	72	10	2.55
19000	7.5	72	10	2.65
20000	8	71	10	2.70
21000	8.5	71	10	2.80
22000	9	71	10	2.90
23000	9	70	10	2.95
24000	9.5	70	10	3.00
25000	10	70	10	3.05
26000	10	70	10	3.15
27000	10.5	69	10	3.15
28000	11	69	10	3.20
29000	11	69	10	3.25

Table 7.4: Hydrostatic data of the proposed vessel

i	$W_i(ton)$	$LCG_i(m)$	$VCG_i(m)$	$TCG_i(m)$	$I_i(m^4)$	$\delta_i(ton/m^3)$
Lightship	8000	60	9	0	0	1
Fresh water port	114	1	12	-7	190	1
Fresh water starboard	145	1	12	7	249	1
Heavy fuel oil port	500	23	7	-8	77	0.970
Heavy fuel oil starboard	350	25	7	8	75	0.970
Provisions	15	15	18	0	0	1
Lubricating oil	35	20	1	0	6	0.70
Diesel oil	60	11	8	9	47	0.88

Table 7.5: Constants and consumables of the proposed vessel

f	$Pos_f(m)$	$W_f(ton)$	$S_f(m^2)$
F1	30	2500	240
F2	40	430	190
F3	50	450	210
F4	60	450	220
F5	70	340	150
F6	80	700	360
F7	90	340	160
F8	100	310	160
F9	110	320	150
F10	120	330	200
F11	130	200	140
F12	150	290	120

Table 7.6: Frames information of the proposed vessel

	Dataset1	Dataset2	Dataset3
Number of cargoes to be planned:	15	18	20
Volume to be planned (m^3):	13683	14048	12983

Table 7.7: Summary of the three datasets

c	$Vol_c(m^3)$	$\delta_c(ton/m^3)$	Segregation ID	Loading temp (°C)	Max. adjacent temp (°C)
1	1460	1.216	4	40	45
2	1670	0.898	14	0	0
3	1127	0.887	0	0	60
4	271	0.921	14	0	0
5	953	1.049	4	32	38
6	575	0.870	4	0	49
7	800	0.785	20	0	40
8	252	0.992	4	0	35
9	831	1.203	12	32	40
10	685	1.460	36	0	65
11	225	1.111	8	43	55
12	385	0.650	0	0	48
13	3430	0.933	13	0	36
14	568	0.881	34	0	37
15	451	0.887	14	0	0

Table 7.8: Dataset 1

by the tanker company's operations manager. Computational times also proved

c	$Vol_c(m^3)$	$\delta_c(ton/m^3)$	Segregation ID	Loading temp (°C)	Max. adjacent temp (°C)
1	1014	0.9036	34	35	20
2	921	0.9036	33	35	20
3	487	1.03004	4	38	20
4	1276	0.9	14	20	20
5	233	0.89	40	20	20
6	503	0.896	7	35	20
7	551	0.847	32	20	20
8	530	0.776	19	35	20
9	502	0.724	41	35	20
10	1009	0.769	20	78	20
11	1082	1.03	21	20	20
12	641	0.74	30	20	20
13	938	0.99266	20	37	20
14	1249	1.034	4	28	20
15	792	0.887	14	20	20
16	771	0.72	30	20	20
17	739	0.87	32	20	20
18	810	1.0594	8	65	20

Table 7.9: Dataset 2

c	$Vol_c(m^3)$	$\delta_c(ton/m^3)$	Segregation ID	Loading temp (°C)	Max. adjacent temp (°C)
1	114	0.870	30	82	88
2	979	0.885	34	58	80
3	1068	0.885	34	58	80
4	300	0.789	18	20	99
5	381	1.030	40	20	49
6	508	0.903	0	20	48
7	581	0.903	0	20	48
8	577	0.910	0	20	48
9	720	1.133	41	60	70
10	1073	0.825	0	15	60
11	593	0.885	0	38	60
12	594	0.858	0	60	70
13	793	0.965	37	20	38
14	450	1.167	17	30	32
15	826	1.211	12	49	50
16	500	0.821	16	25	99
17	1000	1.002	34	38	65
18	701	1.212	4	40	45
19	673	1.014	34	52	65
20	552	1.028	9	25	38

Table 7.10: Dataset 3

	With stability constraints		Without stability constraints	
Restarts	Freespace	Runtime	Freespace	Runtime
Dataset 1				
1000	4220,28	63,68	5209,21	9,83
2000	4425,21	91,68	5225,92	17,89
3000	4563,88	121,47	5233,51	25,89
4000	4604,32	148,79	5246,00	35,34
5000	4622,10	168,05	5243,57	41,29
Dataset 2				
1000	3887,32	59,23	4871,38	16,37
2000	3982,64	88,00	4949,64	30,00
3000	4017,22	107,00	4996,53	42,94
4000	4051,55	137,09	4992,11	53,87
5000	4051,10	162,45	5027,99	66,82
Dataset 3				
1000	4212,24	77,32	5085,15	16,76
2000	4273,32	112,45	5144,06	29,73
3000	4362,98	148,45	5171,50	43,03
4000	4413,66	178,81	5183,38	54,38
5000	4459,70	210,77	5187,39	66,18

Table 7.11: Average free space and runtime vs. number of restarts for datasets 1, 2 and 3

to be operationally acceptable.

7.5.3 Consequences of stability constraints

As mentioned in the literature review in Section 7.2, previous articles on the Tank Allocation Problem did not include all of the stability constraints discussed in this paper. To examine the importance of including stability constraints, the three datasets were also solved without taking into account maximum trim and draft, list, metacentric height, shear forces and bending moments. This means that only the LNS was run, without the validation of stability by the LP model in the leaf nodes of the CP search.

Figures 7.9, 7.10 and 7.11 give a comparison of the results obtained with and without taking the stability constraints into account.

Although the solutions being generated without considering the stability constraints have, on average, a higher amount of free space within lower computational times, the extended problem formulation and solution framework presented in this paper are clearly to be preferred from an operational point of view as they elimi-

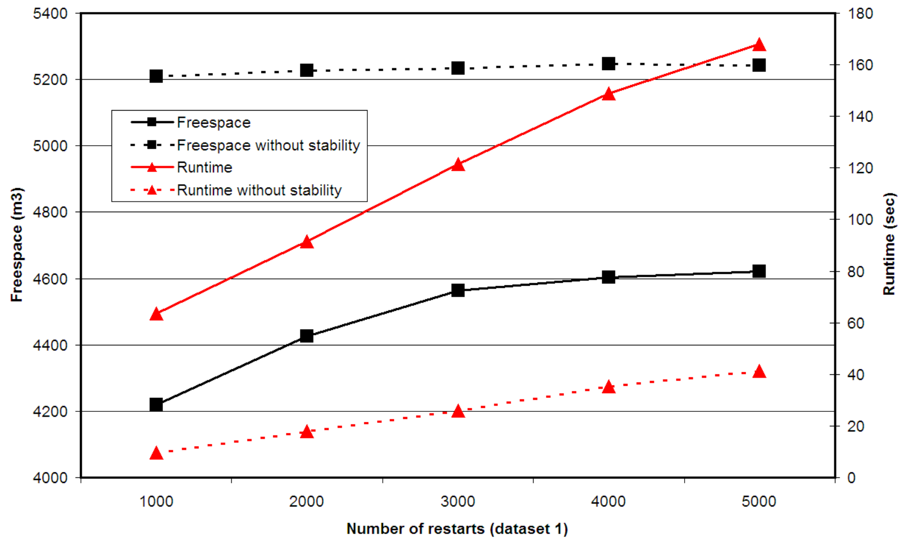


Figure 7.9: Average free space and runtime vs. number of restarts for dataset 1 (Full lines: with stability, Dotted lines: without stability)

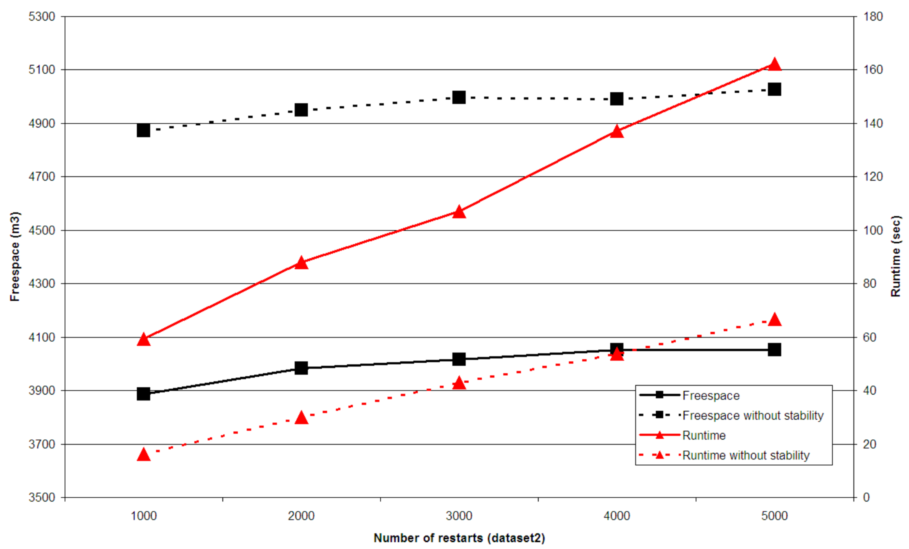


Figure 7.10: Average free space and runtime vs. number of restarts for dataset 2 (Full lines: with stability, Dotted lines: without stability)

nate the time-consuming task of validating the stability constraints and manually repairing solutions that are infeasible due to stability constraints.

Table 7.12 illustrates that most of the solutions generated without considering

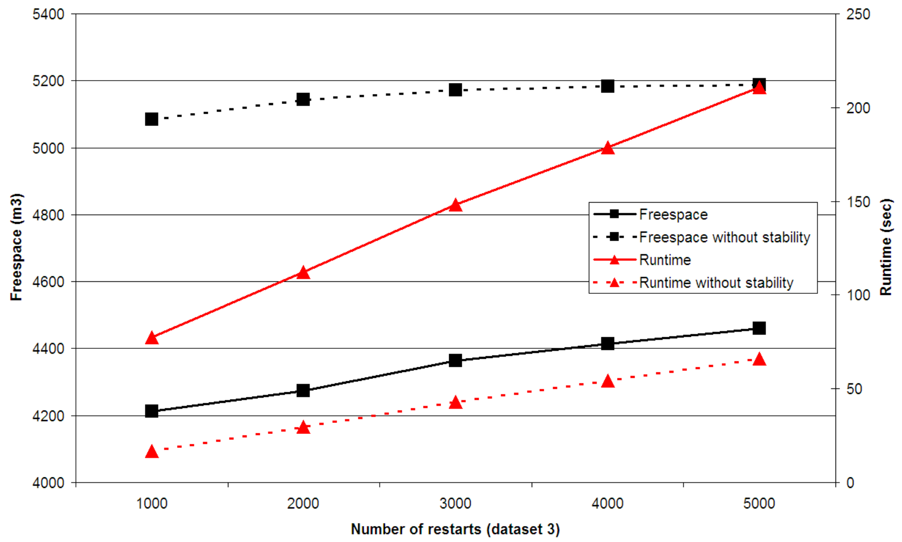


Figure 7.11: Average free space and runtime vs. number of restarts for dataset 3 (Full lines: with stability, Dotted lines: without stability)

	Feasible	Infeasible due to			
Number of restarts		Trim and draft	Trim	Draft	Shear forces or bending moments
5000	47	21	8	22	2

Table 7.12: Solution feasibility for dataset one using five thousand restarts

stability constraints are likely to be infeasible. Out of hundred solutions obtained for Dataset 1 using 5000 restarts, only 47 meet all the stability considerations required in practice. Eight solutions violated specifications on the trim, 22 on the required draft and 21 solutions violated both of these constraints. The fact that only 2 out of 100 solutions violated the crucial shear force and bending moments specifications can be explained by the relatively large amount of cargo that needed to be loaded in the three real-life data sets. If less cargo would have been available, it is more likely that these constraints would be violated if not taken explicitly into account during the optimization process.

7.6 Conclusion

This paper presents a hybrid constraint programming approach to solve a real-life tank allocation problem in the chemical shipping industry. By using constraint programming to consider possible cargo-to-tank allocations and determining the actual volumes by linear programming, the model is the first to take all relevant segregation and stability constraints into account.

Three real-life problem instances are analyzed in detail to validate the performance of the proposed solution approach. It is shown that existing models from the literature, which ignore some stability constraints, would yield infeasible solutions for the problem at hand. Both in terms of speed and solution quality, the new solution approach has been shown to meet the requirements of the industry. To support further research in this field, the details of the real-life problem instances are fully disclosed in the paper.

Future research will be aimed at incorporating the navigation between the various ports and the cargo to be loaded and unloaded in the subsequent ports of call.

Bibliography

[1] Al-Khayyal F., Hwang S.J. *Inventory constrained maritime routing and scheduling for multi-commodity liquid bulk, Part I: Applications and model.* European Journal of Operational Research 176 (2007) p106..130

[2] Barbucha D. *Three approximation algorithms for solving the generalized segregated storage problem.* European Journal of Operational Research 156 (2004) p54..72

[3] Bartak, R. *Constraint Programming: In Pursuit of the Holy Grail in Proceedings of the Week of Doctoral Students WDS99.* MatFyzPress IV (1999) 555-564; http://www.cs.sfu.ca/CourseCentral/417/havens/papers/CPintro.WDS99.pdf (accessed on 08 May 2009)

[4] Bausch D.O., Brauwn G.G., Ronen D. *Scheduling short-term marine transport of bulk products.* Maritime Policy and Management 25:4 (1998) p335..348

[5] Christiansen M., Fagerholt K., Nygreen B., Ronen D. *Maritime Transportation, Chapter 4.* Handbooks in Operations Research and Management Science 14 (2007) p189..284, DOI: 10.1016/S0927-0507(06)14004-9

[6] Clarksons.net *Historical development of chemical tanker fleet (01.01.2009)* available at: http://www.clarksons.net/tables/tables.asp?pubFilter=Chemical+Tanker+Register (accessed on 08 May 2009)

[7] Danna E., Perron L. *Structured vs. Unstructured Large Neighborhood Search : A Case Study on Job-Shop Scheduling Problems with Earliness and Tardiness Costs.* Lecture Notes in Computer Science 2833 (2003) p817..821

[8] Derrett D.R., Barrass CB. *Ship Stability for Masters and Mates, Edition: 6 consolidated.* Elsevier, 534 pages, 2006, ISBN 978-0-7506-6784-5

[9] Desaulniers G., Prescott-Gagnon E., Rousseau L.M. *A Large Neighborhood Search Algorithm for the Vehicle Routing Problem with Time Windows.* Metaheuristics International Conference (2007)

[10] Godard D., Laborie P., Nuijten W. *Randomized Large Neighborhood Search for Cumulative Scheduling.* Association for the Advancement of Artificial Intelligence (2005)

[11] Hvattum L.M., Fagerholt K., Armentano V.A. *Tank allocation problems in maritime bulk shipping.* Computers and Operations Research 36:11 (2009) p3051..3060

[12] Jetlund A.S., Karimi I.A. *Improving the logistics of multi-compartment chemical tankers.* Computers and Chemical Engineering 28 (2004) p1267..1283

[13] Neo K.H., Oh H.C., Karimi I.A. *Routing and cargo allocation planning of a parcel tanker.* 16th European Symposium on Computer Aided Process Engineering and 9th International Symposium on Process Systems Engineering, W. Marquardt, C. Pantelides (Editors), Published by Elsevier B.V. (2006) p1985..1990

[14] Partovi F.Y. *An analytical model of process choice in the chemical industry.* International Journal of Production Economics 105 (2007) p213..227

[15] Perron L., Shaw P., Furnon V. *Propagation Guided Large Neighborhood Search.* Lecture Notes in Computer Science 3258 (2004) p468..481

[16] Pintens T. *Geoptimaliseerde laadplanning aan boord van tankers en bulkschepen.* Master's thesis, Antwerp Maritime Academy, Antwerp, Belgium (in Dutch) (2008)

[17] Shaw P. *Using Constraint Programming and Local Search Methods to Solve Vehicle Routing Problems.* Lecture Notes in Computer Science 1520 (1998) p417..431

[18] Vouros G.A., Panayiotopoulos T., Spyropoulos C.D. *A Framework for Developing Expert Loading Systems for Product Carriers.* Expert Systems With Applications, Vol. 10:1 (1996) p113..126

.1 IMDG segregation matrix

CLASS DESCRIPTION	CLASS	1.1 1.2 1.5	1.3 1.6	1.4	2.1	2.2	2.3	3	4.1	4.2	4.3	5.1	5.2	6.1	6.2	7	8	9
EXPLOSIVES	1.1 1.2 1.3	*	*	*	4	2	2	4	4	4	4	4	4	2	4	2	4	X
EXPLOSIVES	1.3 1.6	*	*	*	4	2	2	4	3	3	4	4	4	2	4	2	2	X
EXPLOSIVES	1.4	*	*	*	2	1	1	2	2	2	2	2	2	X	4	2	2	X
FLAMMABLE GASES	2.1	4	4	2	X	X	X	2	1	2	X	2	2	X	4	2	1	X
NON-TOXIC , NON-FLAMMABLE GASES	2.2	2	2	1	X	X	X	1	X	1	X	X	1	X	2	1	X	X
TOXIC GASES	2.3	2	2	1	X	X	X	2	X	2	X	X	2	X	2	1	X	X
FLAMMABLE LIQUIDS	3	4	4	2	2	1	2	X	X	2	1	2	2	X	3	2	X	X
FLAMMABLE SOLIDS	4.1	4	3	2	1	X	X	X	X	1	X	1	2	X	3	2	1	X
SUBSTANCES LIABLE TO SPONTANEOUS COMBUSTION	4.2	4	3	2	2	1	2	2	1	X	1	2	2	1	3	2	1	X
SUBSTANCES , WHICH , IN CONTACT WITH WATER , EMIT FLAMMABLE GASES	4.3	4	4	2	X	X	X	1	X	1	X	2	2	X	2	2	1	X
OXIDIZING SUBSTANCES	5.1	4	4	2	2	X	X	2	1	2	2	X	2	1	3	1	2	X
ORGANIC PEROXIDES	5.2	4	4	2	2	1	2	2	2	2	2	2	X	1	3	2	2	X
TOXIC SUBSTANCES	6.1	2	2	X	X	X	X	X	X	1	X	1	1	X	1	X	X	X
INFECTIOUS SUBSTANCES	6.2	4	4	4	4	2	2	3	3	3	2	3	3	1	X	3	3	X
RADIOACTIVE MATERIAL	7	2	2	2	2	1	1	2	2	2	2	1	2	X	3	X	2	X
CORROSIVE SUBSTANCES	8	4	2	2	1	X	X	X	1	1	1	2	2	X	3	2	X	X
MISCELLANEOUS DANGEROUS SUBSTANCES & ARTICLES	9	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Figure 12: IMDG segregation matrix

Chapter 8

Conclusion

Operations at a modern terminal servicing ocean going vessels have to cope with varrying amounts of goods handled, heavy infrastructure costs and so-called fore-cast information that can change any minute. This fore-cast information concerning e.g. the destination of the goods or vessel arrival times is crucial in any terminal process planning and process optimization. The maritime industry is characterized as very dynamic where decisions have to be made on the spot while taking many operational constraints into considerartion. It is for this reason that any decision suppor system can prove useful. Not for replacing personnel but for helping process decision makers make better decisions and allowing them to focus on the bottlenecks of the considered process.

In this work three processes of a terminal are studied and for all three optimization strategies are proposed.

For the gate-in process a simulation was build and the question was asked what would happen if inputting times of visit characteristics were reduced. The results indicated that waiting times could be significantly reduced (-62%). Further reasearch and a cost-benefit analysis will have to prove whether this approach may be interesting to implement or not. Using a simulation tool as proposed in this work could also be used to answer other managerial questions: What would be the impact on the average waiting times when the trucks were to follow an other arrival pattern then the historical daily averages used now? This arrival pattern could also be imposed by contractual agreements between transporting companies and the terminal operator where arrival slots could be agreed upon. This agreement on arrival slots could i.e. allow the terminal operator to better manage peak moments. An other approach could be to search for the minimum amount of truck visits required to keep the gate hours opened during night hours. Together with the arrival time slots concept for the night time trucks this simulation might prove useful.

The second process concerns the allocation of berths and quay cranes to vessels in order to service them as efficient as possible. The efficiency parameters or key performance indicators (KPI's) here are crucial. Literature shows that these KPI's may vary depending on local customs. In the first proposed solution approach vessels are handled as soon as possible. The second approach takes the aspect of commercial windows into consideration. The first approach uses a mixed integer linear programming approach (MILP) while for the second approach constraint programming (CP) was used. This change in approach was done in order to be able to include more operational constraints into the model and to keep computational times operationally acceptable. The new approach also allowed to reduce the time intervals in which the model looks for repositionings of the cranes to be dropped from hours to minutes. For further development tidal windows could be added thus forcing deep draught vessels only to leave within certain tidal or time windows. Using the output of the proposed model one could also measure what the impact of adding another vessel service to the terminal might have on the crane utilization or berth congestion.

The third and last process is that of deciding/planning where to load which cargo in cargo tanks on board of a chemical tanker. A hybrid CP approach is used for creating a model that takes all stability and seggregation constraints into consideration. Introducing all the stability constraints into the cargo planning aspect allows operators to create better cargo plans that have a much better change to be accepted by the shipboard personnel while still optimizing the cargo tank useage. For further research it might prove very interesting to add the port scheduling aspect together with the tank cleaning cost. When these functions are included it might be possible to not only maximize tank useage but also minimize fuel and cleaning costs. When also the shipping freights of each chemical is known it might even become possible to have the model select the products to be loaded in order to maximize profits. This model was, like the others, validated together with personnel that handled the processes on a daily basis. All the proposed models also take many operational constraints into consideration that were fully disclosed in this work together with generated datasets allowing the academic community to benchmark future research when tackling the mentioned problems.

This research proved that it is possible to include many operational constraints when developing simulation or optimization models. It is the integration op operational knowhow, academic research and strong modeling capabilities that makes models like these presented interesting for the industry.

List of Figures

List of Tables