

Linkosaari, Teemu

Conference Paper

A Functional Mathematical Optimization Algorithm for the Integration of the Tactical Berth, Quay Crane and Vehicle Scheduling

Provided in Cooperation with:

Hamburg University of Technology (TUHH), Institute of Business Logistics and General Management

Suggested Citation: Linkosaari, Teemu (2014) : A Functional Mathematical Optimization Algorithm for the Integration of the Tactical Berth, Quay Crane and Vehicle Scheduling, In: Blecker, Thorsten Kersten, Wolfgang Ringle, Christian M. 978-3-7375-0341-9 (Ed.): Innovative Methods in Logistics and Supply Chain Management: Current Issues and Emerging Practices. Proceedings of the Hamburg International Conference of Logistics (HICL), Vol. 18, epubli GmbH, Berlin, pp. 85-97

This Version is available at:

<https://hdl.handle.net/10419/209226>

Standard-Nutzungsbedingungen:

Die Dokumente auf EconStor dürfen zu eigenen wissenschaftlichen Zwecken und zum Privatgebrauch gespeichert und kopiert werden.

Sie dürfen die Dokumente nicht für öffentliche oder kommerzielle Zwecke vervielfältigen, öffentlich ausstellen, öffentlich zugänglich machen, vertreiben oder anderweitig nutzen.

Sofern die Verfasser die Dokumente unter Open-Content-Lizenzen (insbesondere CC-Lizenzen) zur Verfügung gestellt haben sollten, gelten abweichend von diesen Nutzungsbedingungen die in der dort genannten Lizenz gewährten Nutzungsrechte.

Terms of use:

Documents in EconStor may be saved and copied for your personal and scholarly purposes.

You are not to copy documents for public or commercial purposes, to exhibit the documents publicly, to make them publicly available on the internet, or to distribute or otherwise use the documents in public.

If the documents have been made available under an Open Content Licence (especially Creative Commons Licences), you may exercise further usage rights as specified in the indicated licence.



<https://creativecommons.org/licenses/by-sa/4.0/>

A Functional Mathematical Optimization Algorithm for the Integration of the Tactical Berth, Crane and Vehicle Scheduling

Teemu Linkosaari

Abstract

In this paper, the main problem considered is a tactical discrete berth allocation, including both the quay mobile crane scheduling and vehicle dispatching as sub-problems. Berths, cranes and yard vehicles are the most important resources used in container terminals. The objective is to reduce the vessel turnaround times with fewer resources. We assume that the vessel's stowage plan and yard templates or their estimates are given.

We integrate using functional decomposition, where sub-problems are solved sequentially and in parallel, resulting in more modifiable and detailed parameters to the main problem. We have chosen the sub-problems and their solution algorithms primarily so that they are mathematically proven to be optimal or have proven properties. Different techniques are utilized: mixed integer linear programming, max plus algebra, greedy algorithms, dynamic programming written in functional language and general algebraic modeling system.

In addition, we extend the features of the sub-problems. While allocating the cranes, we use the concept of the quay crane profile. We reformulate the crane scheduling problem so that the mobile cranes can be heterogeneous, and there can be both quay and mobile cranes. Some features are optional: for example, double cycling and yard remarshaling. Alternatively, we can reduce the number of vehicles by deciding how we unload or load stacks. The study is motivated by the practical needs of Finnish port operators.

Keywords: container terminals, scheduling, optimization, functional programming

1. Introduction

In maritime transportation, container terminals are a source of many interesting large-scale optimization problems. There needs to be deeper insight into these complex systems and they require a larger set of solution techniques. The main function of the container terminal is to transfer containers from one mode of transportation to another. A container is a rectangular metal box, usually 20 or 40 feet long. The second function of the terminal is to provide a temporary storage facility of a few day's duration.

Container activities can be divided into the following categories: export, import and transshipment. We will focus here on import-export terminals in Finland, which are smaller than transshipment terminals and hence easier to optimize. Container vessels can also be categorized as feeders and mother vessels. Here we will focus on feeders.

Almost all relevant problems in concerting container terminals are NP-hard or NP-complete in nature. Fortunately, restricted special cases have polynomial or pseudo-polynomial time complexity and as such, can be exploited.

The time it takes to handle a container vessel depends on many factors due to interdependencies between different processes. Berth planning is highly interrelated with vessel, yard, equipment and workforce planning. A good berth plan saves time, money and resources. The duration of berthing of a vessel depends on the number of quay or mobile cranes allocated to the vessel. Vessels are partitioned into bays which contain deck and hold. The processing time of the vessel depends on the strategy for how bays are handled and on the amount of yard resources the strategy uses and how far or close the containers are positioned relative to the vessel. When there is a limited number of berths and limited number of quay and mobile cranes, these resources must be allocated wisely.

The main objective of our research problem is to find a way to use existing literature directly, by modification, or by integration in order to solve real-world problems in Finnish ports. Our research is aimed toward creating an interactive

optimization and planning tool for container terminals. We review only those research papers that have been implemented or are relevant for this work. We try to reduce each vessel's turnaround times in addition to the resources described above.

Clearly, there are multiple objectives, which are treated lexicographically: first, minimize the vessel turnaround time; second, minimize the number of handling resources within a time window. The contract model between the terminal and the shipping company may affect the objectives. If, for example, the time window of one vessel is relaxed, then we can it will be possible to give more of its resources to other vessels. We do not consider how containers are loaded onto trucks or trains, since that is not critical to the turnaround time. There could be other objectives such as: minimizing travel distances, fuel maintenance, and remarshaling costs.

In the literature, the trend is to integrate the resources-related sub-problems. In deep integration, sub-problems are merged into a monolithic problem and can be hard to solve or modify. We use functional modular integration, in which sub-problems are solved sequentially and in parallel, resulting in creation of more modifiable and detailed parameters to the main problem. The port planner should also have the opportunity to lend his insight and experience to the optimization process (see Bruggeling et al. (2011) for more). Therefore we use modularization, in which the operator can give a rough estimate of some parameters and more detailed data to for others.

In the forthcoming sections we present relevant literature, models, and solution methods. One can think of the berth and resource allocation problem as a three-level optimization problem. First, we introduce the vehicle dispatching algorithms used, followed by an examination of quay crane scheduling problems. Third, we address the tactical berth allocation problem. The final section concludes the paper.

2. The structure of the modular algorithm

In this section we outline the input, the main algorithm and the chosen programming paradigm. We assume that the vessel's stowage plan and yard templates or their estimates are given. Container vessels can be modeled as a 3-dimensional matrix, in which containers are stacked on top of one another and arranged in rows. In the yard storage space, containers are freely positioned in the yard in any orientation. Usually, they can be stacked to form blocks. To calculate distances between different container locations, one may construct travel paths and then determine the shortest paths, in which the triangle inequality holds between distances.

The structure of the program is as follows: first, we compute the set of strategies that can be used to process one ship bay. This computation can be done in parallel. Next, we calculate the optimum vessel schedules for different quay crane profiles. On the third level, we attach the crane schedules to berth allocation and to the week schedule.

Here, sub-problems are treated as functions in a more mathematical sense, which is the advantage of functional programming that is a declarative way to write a program with functions. Programs are coded in F#. In this study, functions have deterministic behavior and no side-effects. Therefore, it is also easier to reason about the program and even to provide a formal verification of the system. Parallel and asynchronous programming are also easier.

The non-functional part of the program uses the general algebraic modeling system (GAMS), which is also a declarative way to model optimization problems. It supports a variety of commercial solvers such as CPLEX, which is utilized in mixed integer linear programming. Vector graphics is used as a communication tool and to read and write data (e.g., time windows, container locations). Spreadsheets or databases can also be used.

3. Yard side scheduling

Vehicle dispatching problems can be considered as the first level of the algorithm. They are usually quickly solved due to polynomial time algorithms and hence suitable for a sub-problem. They were introduced early in container literature: see Bish et al. (2005), Li et al. (2004) and Zhang et al (2005). They are designed for automated guided vehicles in order to schedule a given number of homogenous vehicles and a crane sequence, but they can be used more generally. Here we refer to their results and how they are used.

Here we use these methods to ask: Given the yard template and the stowage plan what is the minimum number of vehicles necessary to unload or load one bay? We may also ask: what is the minimum number of vehicles necessary if the vessel's total handling time or makespan is fixed.

In manned terminals, the yard transportation vehicles used are yard trucks, straddle carriers and reach stackers. We assume that there is no buffer time below the crane. The work is usually organized as a team, usually called a gang, which services one crane and consists of a chosen combination of the aforementioned vehicles and other employees in the yard.

3.1 Vehicle dispatching problems

In solving vehicle dispatching problems we first need to determine the crane processing time for a container job: how long it takes to move a container from the ship to the yard or vice-versa. An exact method would be to use control theory to model the crane movement and minimize its travel path. Alternatively, one can use an average time of the container.

Secondly, we need to calculate the vehicle processing time: how long it takes to pick up a container, move it to the yard location and then drop it there and come back or the same operation, but in another direction. Several factors affect the drop-time: how high the stack is and whether one uses intermediate buffers or a yard servicing vehicle. We assume that there is no congestion in the yard vehicle traffic.

Thirdly, Zhang et al. (2005) formulated unloading phase as a mixed integer program, but the problem structure enables us to solve it by using a greedy algorithm, that is, by the first available truck rule as appears in Li et al. (2004). While Bish's algorithm does not assume job starting times, Zhang's algorithm does. The other difference is that in Bish's algorithm the crane begins to unload a container after the next vehicle arrives, not earlier. In Zhang's algorithm, the crane is allowed to start the next job immediately and then wait for the vehicle. It is therefore more efficient.

Next, Bish et al. (2005) also provided a loading lemma, which gives the reversed greedy algorithm. It is the same as the latest busy truck rule by Li and Vairaktarakis (2004).

Lastly, the unloading and loading phases can also be combined. For small problems, total enumeration works. Li and Vairaktarakis (2004) also studied heuristics. However, combining phases is sometimes unnecessary since port planners may want first to unload all the bays and then load them, because the purpose of the land trucks is to transport their cargo immediately to its final destination (see Figure 2, section 4.2). These strategies are sufficient when the distance between crane and container is short. The processing of one bay is not really a deterministic process; thus, stochastic methods could also be used to solve this problem.

3.2 Servicing a block

Gilliambardo et al. (2010) utilized a piecewise linear function for calculating the cost of transporting a container depending on the distance. For short distances, a greedy structure is sufficient, but when the block and the crane are situated far from each other, then a more complicated crane-mover-vehicle assignment is necessary. The next tour optimization method by Vis and Roodbergen (2009) can be used in combination with greedy algorithms to reduce bay processing time when containers are situated far from the vessel. Thus we have one more strategy to handle a ship bay.

Next we describe the problem and its solution methods. Let us imagine a straddle carrier in one yard block for storage and retrieval of containers. The block consists of a number of yard-bays or rows. The stack has only one layer. The outcome is defined as an optimal tour for storage and retrieval requests. One heuristic is to use the first-come-first-served rule for every requested container, but a block-scheduling approach is more effective. In it, we optimize the tour of the next few containers ahead in the straddle carrier's task.

The modeling technique for tour optimization is interesting and complex. A row can be understood as a special case of the Directed Rural Postman problem, and then converted into an asymmetric Steiner Traveling Salesman Problem, use optimal assignment with Monge matrices and convert back. We can combine the rows from different directions with dynamic programming with utilizing Bellman's optimization principle.

We note that terminal productivity will not benefit much from faster vehicle operations without effective storage yard strategies. When there is time, it is usually possible to reorder the yard or some of it before the next ship arrives and therefore reduce turnaround time. This could serve as one additional strategy.

4. Berth side scheduling

Now we have methods for calculating one ship-bay using different techniques and different vehicle configurations. Next we consider how to use them in the next level.

4.1 Crane sequences

The usual bottleneck of the terminal is a quay or mobile crane which loads and unloads containers to and from a vessel. The quay cranes are rail mounted and cannot cross each other producing non-crossing constraints to the mixed integer model. Mobile cranes, on the other hand, are usually slower but are not rail-mounted and therefore are easier to move across the terminal. Also, mobile cranes can cross other cranes. The crane sequence (the order in which

containers are handled), also affects the number of yard vehicles. We assume that a crane can handle only one container a time, although this could be changed in the future.

Pap et al (2011) noticed that one can minimize the number of yard vehicles by unloading or loading stack from the ship in a different order while the bay makespan remains the same. They used max-plus algebra, which is an attractive method for modeling non-linear problems linearly. Unfortunately, Pap's study did not attempt to find the optimum stack handling order. We have used max-plus algebra in a very similar fashion as we used the greedy algorithms mentioned above. By combining a basic genetic algorithm with a max-algebraic formulation we have found a near-optimal stack handling order which minimizes the number of yard vehicles.

Another type of crane sequence is a double cycling sequence in which we unload and load containers at the same time. Goodchild and Daganzo (2006) reformulated this handling method as a two-machine flow shop scheduling problem which is solved to optimality using Johnson's rule. When to use and not use double cycling would be a decision variable in this instance. The crane's cycle time is longer but the reduction of turnaround time can be even up to one fifth depending on the structure of the vessel. This technique is not always applicable due to limits of buffer size in some terminals.

4.2 Crane scheduling

We turn now to the second level of our algorithm: how to use these different bay strategies and their times to calculate the processing time or makespan of one vessel.

With regard to the problem of quay crane scheduling, our goal is to determine a handling sequence of holds for quay cranes assigned to a container vessel considering interference between quay cranes, at the same time minimizing the makespan. There are additional restrictions we must take into consideration. For example, two cranes cannot be too close to each other. That is, there must be a minimum separation constraint, usually at least one bay. The input data for the

quay crane scheduling consists of the vessel stowage plan, the loading plan, and a yard map showing the stowage locations of containers to be loaded on the vessel. This problem has been shown to be NP-complete. Recent research has aimed at adding features of practical relevance and developing efficient solution ideas (see Bierwirth and Meisel (2010) for more details).

Next we model and extend the problem proposed by Lee et al. (2008), who modeled the quay crane scheduling problem using a homogenous set of quay cranes. However, if processing time depends on the crane used, the worker or workers, the number of vehicles, or any particular strategy involved, then clearly we have a heterogeneous set of cranes. The processing time depends also on how long it takes to move containers with using a chosen strategy. Since the yard locations are known, then we can calculate the processing time of every bay in the unload and load phases.

Thinking in this way, we can minimize the number of yard vehicles used in the gang working in a bay. Here we assume that there is one gang per one crane, but it would be interesting to consider how vehicle pooling, in which vehicles work together with many cranes, would integrate to this setting. Figure 1 illustrates a solution to an instance of this problem with involving vehicles. In future research a more advanced quay crane scheduling model can be used as a basis.

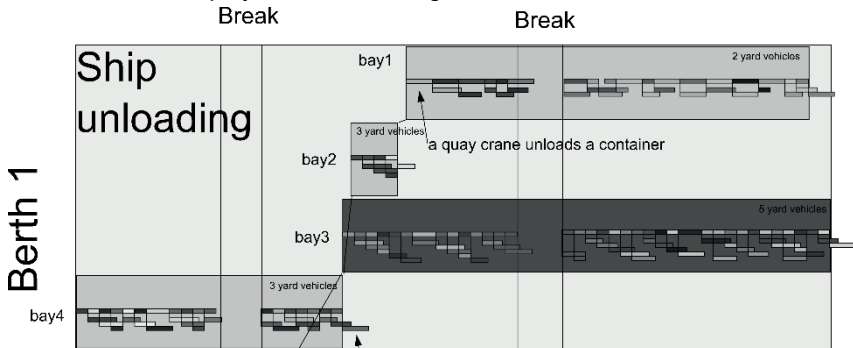


Fig. 1: A schematic example of the unloading phase with indicating breaks and amount of yard vehicles used. Crane one processes bays 5 and 3. Crane two processes bays 4, 2, and 1.

In the literature, crane models assume that the work is done non-stop. In manned terminals, work shifts and breaks also affect the model. We have primarily considered minimizing a vessel's turnaround time and the number of vehicles involved in unloading and loading, but there are other factors to consider as well. We can assume that night shifts are more expensive than day shifts. One decision variable could be whether we skip the night shift or not. Also if the contracted time window is exceeded, then a penalty has to be modeled.

Modular design enables us to solve special cases such as the single crane version (that is, a Traveling Salesman Problem with Precedence Constraints), separately. Otherwise, depending on the accuracy available, the crane scheduling problem can be very time-consuming to solve. In the case of small-sized instances our formulation can be solved by using CPLEX. For larger problems such as branch-and-price, branch-and-cut and still larger problems, metaheuristics would have to be utilized for their solution.

4.3 Berth allocation

We will conclude our study by considering the third level of the algorithm, berth allocation problems. These problems are highly interrelated with quay crane scheduling. A berth allocation problem involves assigning arrival ships to good berthing positions. It belongs to the class of NP-hard problems. Problems can be classified as either discrete or continuous. If we can assign only one ship at a time, the problem is discrete; if there is more than one ship, it is continuous.

In a quay crane assignment problem, we assign a number of quay cranes to vessels. The number of quay cranes assigned to a vessel often depends on contracts between the terminal and shipping companies. We say that the berth allocation problem is tactical if it involves quay crane assignment, in which we allocate cranes to vessels over time.

There have been many attempts to integrate crane allocation with berth scheduling. We refer here to the model articulated in Gilliambardo et al. (2010). That particular model was designed for a transshipment terminal, but it can be modified for an import-export terminal. Gilliambardo's study used the idea of quay

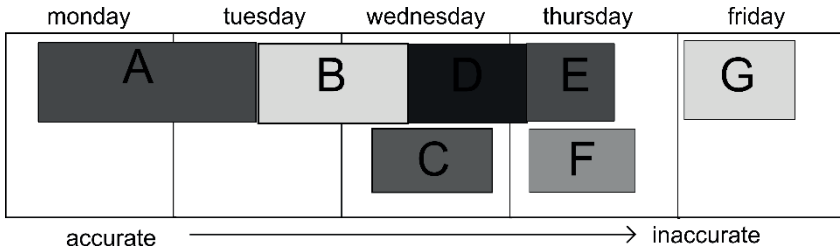


Fig. 2: Real-life time windows for vessels A to G and accuracy of information

crane profiles (how to assign cranes over time), for a discrete berth allocation model. Profiles can start either at the beginning of the shift or in the middle of the shift.

Figure 2, illustrates typical time windows of vessels. In our example, there are a few overlapping vehicles. The objective is to schedule and forecast the week's schedule that is the planning horizon is at most one week.

In practice, the terminal does not have all the information about the following week. The actual arrival times are uncertain and stochastic in real life. For example, winter affects the estimated schedules. Due to incomplete data, it is essential to be able to give both an exact and a rough estimate of the parameters. When crane scheduling is considered from a berth allocation point of view, the model comes more complex. In real life, for one ship the number of crane varies over time, with different amount of quay cranes used in different workshifts. Therefore, a new quay crane model is needed. Difficulties also arise if multiple vessels are berthed at the same time. The main concern here is: for any two ships and their windows, what the minimum number of resources is such that the two ships are processed within their time windows. If the vessel is moored within the contracted time window, then the terminal is responsible for processing a certain amount of moves per hour, otherwise operators are free to choose a berthing time outside the time window.

We extend the idea of quay profiles to contain different set of cranes: quay crane and mobile cranes. In a smaller terminal, it could be possible to treat cranes individually. We may also assign a different set of vehicles and their strategies.

Small problems can be solved with commercial mixed integer programming solvers, but a different method might be needed for larger instances.

5. Conclusion

In this paper, we considered how to integrate tactical the berth allocation problem, the quay crane scheduling problem and a set of related sub-problems into one functional algorithm. Our study was conducted from the point of view of Finnish ports. We focused primarily on how to calculate different bay times with different strategies and how to use these values in quay crane scheduling.

We can summarize the main conclusion of this paper as follows. We proposed a method to integrate the vehicle dispatching problem with its variations to the quay crane scheduling problem. This was done by calculating the bay processing times with different strategies, which provides a variety of parameters. Like Gilliambardo et al. (2010) who applied quay crane profiles to the berth allocation problem, the same idea can be used to provide vehicle profiles for ship bays. Therefore we used a heterogeneous set of cranes in our model. We also considered stack reordering and double cycling as tactics for minimizing the number of yard vehicles. We also used a quay and mobile cranes and considered breaks and skipping night shifts. Other tactics could be reordering the yard and providing a yard server. We also considered berth allocation with a different quay crane profile that contains both the quay and mobile cranes.

With modular design it is now easier to add new features and remove unnecessary elements. It is possible to use both exact information and rough estimates together. We also noticed that functional programming is a succinct and efficient way to combine different algorithms. In the future we will perform extensive computational experiments and present a detailed mathematical description of the model and the algorithm.

References

- Bierwirth C., Meisel F. (2010) A survey of berth allocation and quay crane scheduling problems in container terminals, *European Journal of Operational Research*, Vol. 202, pp. 615-627
- Bish, E.K., Chen, F.Y, Leong, Y.T., Nelson, B.L., Ng, J.W.C., and Simchi-Levi, D. (2005), Dispatching vehicles in a mega container terminal, *OR Spectrum*, Vol. 27, pp. 491-506
- Bruggeling, M., Verbraeck, A., and Honig H. J. (2011) Decision Support for Container Terminal Berth Planning: Integration and Visualization of Terminal Information, *Proceedings van de Vervoerslogistieke Werkdagen*, pp. 263-283
- Goodchild, A. V., Daganzo C. F. (2006) Double-cycling strategies for container ships and their effect on ship loading and unloading operations, *Transportation Science*, Vol. 40, No. 4, pp. 473-483
- Giallombardo, G., Moccia, L., Salani M., Vacca I. (2010) Modeling and solving the Tactical Berth Allocation, *Transportation Research Part B: Methodological*, Vol. 44, No. 2, pp. 232-245
- Li, C., and Vairaktarakis, G. L. (2004) Loading and Unloading Operations in Container Terminals, *IIE Transactions*, Vol. 36, No. 4, pp. 287-297
- Lee D., Wang H. Q., and Miao L. (2008) Quay crane scheduling with non-interference constraints in port container terminals, *Transportation Research Part E*, Vol. 44, pp. 124-135
- Pap, E., Bojanić, V., Georgijević, M., Bojanić, G. (2011) Application of Pseudo-Analysis in the Synchronization of Container Terminal Equipment Operation, *Acta Polytechnica Hungarica*, Vol. 8, No. 6, pp. 5-21
- Vis, I.F.A., and Roodbergen, K.J. (2009), Scheduling of container storage and retrieval, *Operations Research* 57, Vol. 2, pp. 456-467
- Zhang L., Ye R., Huang S., and Hsu W. (2005) Mixed integer programming models for dispatching vehicles at a container terminal, *J. Appl. Math. & Computing* Vol. 17, No. 1-2, pp. 145-170

