

Using service-time profiles for distributed planning of container barge rotations along terminals

Albert Douma*, Peter Schuur, Marco Schutten
University of Twente, The Netherlands

June 30, 2008

Abstract

We consider the barge rotation planning and quay scheduling problem in the Port of Rotterdam. This problem concerns the alignment of barge rotations and terminal quay schedules. We use a Multi-Agent based approach for the problem, since such a Multi-Agent based system can mirror to a large extent the way the business network is currently organized. Moreover, it can provide a solution that is acceptable for each of the actors involved. We develop a model to deal with restricted opening times of terminals, unbalanced networks, sea vessels, and closing times of containers. To deal with these realistic issues, we introduce the concept of *service-time profiles*, which enables us to deal with time-dependent service times. This allows barges to plan a rotation along terminals, taking into account the availability of terminals and the effect on the service time. We perform several simulations to analyze the effect of different factors on the performance of barges and terminals, like restricted opening times and unbalanced networks. We compare the results with an off-line scheduling algorithm. The results indicate that a Multi-Agent based control using service-time profiles is able to perform close to the off-line benchmark. The model in this paper forms a basis for implementation in practice.

Keywords Barge planning, quay scheduling, multi agent based control, simulation, dynamic assignment

1 Introduction

The barge rotation planning and quay scheduling problem we consider is based on a real problem in the Port of Rotterdam. The Port of Rotterdam is the largest container port of Europe and it handled about 6.5 million containers in 2007. Most of these containers are shipped to the hinterland by means of train, truck, and barge. In this study, we focus on the barge container sector, which has become an increasingly important transportation mode over the past decades.

1.1 The problem

The barge rotation planning and quay scheduling problem is rather complex. In the port, there are about 30 terminals, whereas per day 75 to 100 barges visit the port, exchanging containers with, on average, eight terminals. Some of the terminals also handle sea going vessels and some are closed during the night, which limits the possibilities to handle barges. Currently, the planning of a barge rotation, i.e., a sequence of terminal visits, is done by the barge operator that has contracted the barge. The

*A.M. Douma, University of Twente, School of Management and Governance, P.O. Box 217, 7500 AE Enschede, The Netherlands, a.m.douma@utwente.nl.

quay planning is done by the terminal operator of the corresponding terminal. On arrival of a barge in the port, the corresponding barge operator contacts all terminals that need to be visited and tries to make appointments such that the barge can leave the port in time. Communication with the terminals is done by means of phone, E-mail, and fax. However, this takes much time and due to changes and disturbances, it is often not possible to execute a barge rotation or quay schedule as planned. This is due to, among others, the high interdependency of terminals and barges. A delay at one terminal has a dominoes effect on the rest of the terminals.

Currently, a poor alignment of activities, uncertainties during operations, and strategic behavior of both terminals and barge operators lead to inefficient use of quays and long sojourn times of barges in the port. For freight forwarders, this affects the reputation of the port and the attractiveness of using barges for transportation. This is a serious issue nowadays.

In the past, several attempts have been made to establish a central trusted party that coordinates the activities of both terminals and barges. However, these attempts were not successful for the following reasons:

- *Autonomy.* Every terminal and barge operator wants to stay autonomous, i.e., in control of its own operations.
- *No contractual relationships.* There exist no contractual relationships between barges and terminals, which means that barge operators and terminal operators cannot contractually force each other to deliver a certain service or charge each other for poor services.
- *Limited information sharing.* Since barge operators compete with each other (as terminal operators do), they are very reluctant to share information that possibly undermines their competitive position.

However, the problem is even more complex since we deal with

- Different actors each having different interests;
- A highly dynamic environment, i.e., there are a lot of events and disturbances;
- A lowly structured and loosely coupled network, i.e., there exists no clear hierarchy in the market and every day different players can be involved (different barges or even new terminals).

A solution for the barge rotation planning and quay scheduling problem should mirror the current market structure. It is important that the solution is acceptable to all parties involved and optimizes their operations. Acceptance is possibly even more important than optimization. Optimization is a hard concept in this context, since parties have different interests and we deal with a dynamic environment, which means that decisions are made with little knowledge about the future.

1.2 Contribution and previous work

The barge rotation planning and quay scheduling problem is introduced in Douma et al. (2008). In their contribution they develop and evaluate a Multi-Agent based system and compare the results with a central off-line benchmark based on more traditional scheduling techniques. Results indicate that a communication protocol based on waiting profiles is promising. In this study we extend the model proposed in Douma et al. (2008) by introducing the following practical extensions, which make the model more realistic:

- *Restricted opening times of terminals.* Terminals can be closed during a certain period of the day.
- *Unbalanced networks.* Terminals can have a different capacity and utilization degree, and in a port we can have busy and quiet regions.
- *Sea vessels.* Terminals process, besides barges, also sea vessels.
- *Closing times of containers.* Certain containers have a specific time at which they need to be at the terminal, e.g., because they need to be loaded on a sea vessel that is about to leave.

The aim of this study is i) to develop and extend our Multi-Agent based control to be able to deal with these more complicated settings and ii) to get insight in the way the system functions. We extend the model proposed in Douma et al. (2008) by introducing service-time profiles, which are an extension of the waiting profiles with respect to time dependent service times. We consider different experimental settings to evaluate (by means of simulation) the effect of increasing degrees of complexity on the performance of the Multi-Agent based controls and to compare the results (obtained by simulation) with an off-line benchmark, which is a central algorithm based on the assumption that information over the whole planning horizon is known a priori. This information concerns the arrival times of barges and sea vessels, the terminals each barge or sea vessel has to visit, the processing time of each barge and sea vessel, the terminals and their capacity (including opening times), the closing times of containers, and the sailing times in the network. In the off-line benchmark this information is known prior to the related planning horizon, whereas in the Multi-Agent based control, this information is revealed gradually during the planning horizon (except for opening times of terminals, and the arrival and processing times of sea vessels).

1.3 Assumptions

We make the following assumptions. We consider only barges and terminals operators as decision making actors. We assume that both actors are *opportunistic*, meaning that they exploit opportunities for their own benefit with no regard for the consequences for other players. Barges arrive over time with stochastic interarrival times. On arrival in the port the barge operator decides which rotation a barge is going to execute. Decisions of both barges and terminals have to be made in real-time and we assume that never two barge operators plan rotations simultaneously, but one after another. With respect to a terminal, we assume that it handles both barges and sea vessels. With respect to barges, terminals only have information about barges that arrived in the port. The service of a barge is not preempted for the service of another barge. Terminals have fixed capacity and can have restricted opening times, during which barges can be processed. Opening times of terminals are hard, i.e., no work is done in overtime. The time to handle a container, the mooring time, and the sailing time between terminals are deterministic. Sea vessels arrive with stochastic interarrival times at terminals and processing takes a stochastic amount of time. Arrival times of sea vessels are known to the terminal prior to the planning horizon. sea vessels have absolute priority over barges.

With respect to a barge we assume that, on arrival in the port, it needs to make a decision about the sequence in which it visits the terminals concerned. On arrival in the port, the barge has information about the terminals it has to visit, the number of container it has to (un)load at each terminal, and the mooring time at and the sailing time between terminals. It has no information about the state of the network, such as waiting times at terminals. We consider no capacity or stowage constraints for the barge. Barges visit terminals only once. We define closing of the terminal as *preemptive downtime*, and the processing of a sea-vessel as *non-preemptive downtime*. Preemptive downtime means that the

handling of a barge may start before the downtime and finish after it (Schutten, 1998). Non-preemptive downtime means that the handling of a barge might not be interrupted by a downtime. We assume that all terminals have the same objectives and the same holds for all barges. In the sequel we define these objectives, which are different in the various experiments we study. We do not consider disturbances and their effect on the operations of barges and terminals. These assumptions allow barges and terminal operators to make reliable appointments, since no unexpected delays occur. Introducing disturbances requires the introduction of a kind of penalty, e.g., penalty costs, if appointments are not met by one of the parties. This is part of future research.

1.4 Outline of the paper

The outline of the paper is as follows. In Section 2 we describe related problems and earlier studies to the barge rotation planning and quay scheduling problem. In Section 3 we introduce the practical extensions we study in this paper. Section 4 describes briefly the decentralized approach of the problem, the performance indicators, and the different levels of information exchange. In Section 5 we argue why waiting profiles need to be extended to service-time profiles. Sections 6 and 7 describe the barge operator agent and the terminal operator agent, respectively. Section 8 presents the off-line algorithm we use to benchmark the performance of the Multi-Agent based solution. Section 9 describes the experimental settings and Section 10 the results of the simulations and the off-line benchmark. We conclude in Section 11 and 12 with the conclusions and prospects on further research.

2 Related work

Except for a few earlier studies, the problem we consider has not been studied in the literature before. The most related fields are the berth allocation problem and the ship scheduling and routing problem, but both fields do not capture the characteristics of our problem. Also other related problems, like the attended home delivery problem and the hospital patient scheduling problem, show similarities but also important differences. We discuss these issues successively.

In 2003 a first study was performed into the barge handling and quay scheduling problem in the Port of Rotterdam to investigate the possibilities for a decentralized control structure (Connekt, 2003; Melis et al., 2003; Schut et al., 2004). The focus was on creating an off-line planning system, where barge rotations were planned one day in advance and not updated during execution. The main concern in the studies was to create feasible plans since this would mean a major improvement in practice. From the study it became clear that a decentralized control structure, besides the expected improvement, offers an acceptable solution for the parties involved (Moonen et al., 2007). In this paper we focus on a real time planning system and, besides aiming at feasible plans, we also aim at optimizing the operations of both barges and terminals.

A related problem is the berth allocation problem (BAP). It concerns the assignment of berths to ships (for an overview, see, e.g., Cordeau et al., 2005; Steenken et al., 2004; Stahlbock and Voss, 2008). However, there are two major reasons why the BAP is not applicable to our problem. First, in the static and dynamic BAP it is usually assumed that the arrival times of ships as well as the processing times are known at the time the berth plan is made (see, e.g., Imai et al., 2001; Cordeau et al., 2005). In our problem, however, arrival times of barges are uncertain and terminals have to plan their quays taking into account possible future barge arrivals. Second, the BAP considers the operations of a single terminal. In our problem we deal with multiple terminals which are mutually depending on each other

due to the barges that, in general, have to visit more than one terminal. This means that the arrival time of a barge at a terminal is also a result of decisions made at other terminals.

Another related field is the ship routing and scheduling problem (SR&SP). For a recent overview we refer to Christiansen et al. (2004). SR&SPs are considered separately from similar problems in other transportation modes (like vehicle routing problems), due to the specific conditions under which ships operate (Ronen, 1983). Although our problem can be considered as an SR&SP, there are some major differences with the existing literature. First, the routing of ships is mainly along ports instead of terminals within a port. Although this seems a similar problem at a different level, the level of freedom in the sequence ports are visited might be much more limited due to the geographical dispersion than this is the case within a port. Moreover, we need to take into account the availability of berths, whereas the majority of literature in the SR&SP only focuses on the route the ship sails and assumes that (when a port or terminal is not closed) quay space is available to process the ship. Second, nearly all papers that appeared in the field of SR&SP focus on the static problem, i.e., all information is known in advance. Moreover, these problems are solved using a single objective function, which cannot be done in the problem we consider. Within the literature of SR&SP, we like to mention a contribution of Christiansen and Fagerholt (2002), who introduce a service-time function to represent the expected service time of a ship during a certain time horizon. The service-time function is influenced by the availability of the port and used to calculate robust ship schedules. In our study we extend the concept of a service-time function and apply it for the construction of barge rotations.

A third related field is the vehicle routing problem and especially the attended home delivery problem (AHDP) (Campbell and Savelsbergh, 2006). In the AHDP, a carrier offers attended deliveries of packages at the homes of customers. An attended delivery means that a customer has to be present when the package is delivered. To optimize the route the carrier travels, Campbell and Savelsbergh (2006) consider incentive schemes to influence the preferred time windows of a customer. We consider a similar problem, although we have multiple carriers. We take explicitly into account the interest of the customer (the terminal), which might want to plan carriers (barges) close to each other in order to decrease the total time they need to be present ‘at home’. This makes our problem different from the AHDP.

A fourth related field is the hospital patient scheduling problem (HPSP). The HPSP is about the scheduling of patients which can have multiple appointments that have to be scheduled at different resources (see, e.g., Decker and Li, 2000). Especially in the diagnostic phase, the sequence in which tests need to be done is relatively free. This makes the HPSP comparable to the barge handling problem. However, there is a major difference with respect to the arrival time of barges and patients. In our problem, barges sail from the hinterland to the port and are not willing to delay their arrival time with days or weeks. In the HPSP, the arrival time of patients at the hospital is still variable and part of the decision to make the most convenient appointments from the patients’ perspective.

In this paper we approach the problem using Multi-Agent Systems (see for an introduction Wooldridge and Jennings, 1995). In an extensive survey on existing research on agent-based approaches in transport logistics, Davidsson et al. (2005) state that especially applications of agents in transportation via water are scarce and most papers have focused on the alignment of activities at a terminal. Contributions in this field are, e.g., Bürckert et al. (2000), Thurston and Hu (2002), and Franz et al. (2007). We agree with Davidsson et al. (2005) that most agent-based approaches (especially in the maritime industry) are not evaluated properly and comparisons with existing techniques are rare. Examples of two recent papers that apply a quantitative evaluation of their Multi-Agent model are Henesey et al. (2006) and Lokuge and Alahakoon (2007). Most papers stay, however, at the level of a conceptual agent model

and sometimes draw conclusions about the (expected) performance of the model without presenting experimental results. Among the literature on multi-agent based approaches in transport logistics, we found no application similar to the problem we consider (except for the papers already mentioned).

For the modeling of the Multi-Agent based controls and the off-line benchmark, we apply algorithms obtained from the literature on traveling salesman problems and scheduling problems. In the course of this paper, we refer more specifically to literature in this field.

We have chosen to apply (where possible) existing and proven methods, instead of developing them ourselves, to concentrate on the performance of the decentralized control. The reason is that if we develop new methods ourselves, we first have to test the performance of the parts before we can combine them and draw our conclusions. A poor performance of the decentralized control might then be caused by poor methods implemented in one of the parts.

The contribution of this study to the existing literature is the following. First, we extend the models presented in Douma et al. (2008) to deal with more realistic situations. Second, we provide insight in the performance of a Multi-Agent based approach. Third, we compare the results with an off-line benchmark to see how well a decentralized approach performs compared with a centralized approach.

3 Practical extensions

In addition to Douma et al. (2008) we make our experimental settings incrementally complex to study the effects of a specific factors on the performance of barges. We consider the following practical extensions successively: restricted opening times of terminals, unbalanced networks, sea vessels, and closing times of containers.

Restricted opening times of terminals mean that terminals are only opened during a certain time window every day or week. In the Port of Rotterdam a lot of terminals are closed during the night, especially the smaller terminals near the city. This means that the processing of a barge can be interrupted by closing of the terminal, which impacts the sojourn time at the terminal significantly. We assume that opening times of terminals are hard, i.e., even when a barge only needs to load or discharge one container at the moment the terminal closes, it has to wait until the terminal opens again. This is not realistic for all terminals in the port, but in general it is true that workers at the terminal hardly work overtime. In our model we consider closing of a terminal as preemptive downtime (see Section 1.3).

With unbalanced networks we mean that i) terminals are not identical, but have a different capacity and utilization degree and ii) can be distributed unequally over the port, resulting in busy and quiet regions. This is a realistic situation. In the Port of Rotterdam, e.g., about 80% of all container transshipment takes place in one region.

Sea vessels are handled in the Port of Rotterdam only at a few terminals that are able to receive these large vessels. For all these terminals it holds that sea vessels have absolute priority over barges, which means that the handling of a barge is stopped at the time a sea vessel arrives. We consider sea vessels as non-preemptive downtime (see Section 1.3). This choice is in accordance with the current practice in the port, although some terminals sometimes process barges in parallel. Sea vessels have long processing times compared to barges. Sea vessels therefore significantly impact the capacity available for barges. To determine the interarrival and processing time distributions of sea vessels, we analyzed data of sea

vessels that arrived from 2005 until 2007 at the major terminals in the Port of Rotterdam. See also Kuo et al. (2006) for a case study into arrival patterns of container vessels in international ports in Taiwan.

Closing times of containers represent specific times at which containers need to be at a terminal. The main reason for a closing time is the fact that the container has to be shipped with an ocean going vessel, which departs at a specific time (Sinclair and van Dyk, 1987). We therefore assume that only those containers that have to be transhipped at a terminal where also sea vessels are processed, can have a closing time.

4 Decentralized control

Let us first describe the Multi-Agent based approach to the barge rotation planning and quay scheduling problem as presented by Douma et al. (2008). All actors in the network are opportunistic. We describe the conceptual agent model, the performance indicators, and the alternative levels of information exchange.

4.1 Conceptual Multi-Agent model

We consider two types of agents: barge operator agents and terminal operator agents. All agents from the same type (like barge operator agents) have identical intelligence. Agents use a direct communication protocol (negotiation) to come to agreements. The reason is that direct communication mirrors daily practice and is probably easier accepted by barge and terminal operators. The barge operator agent plays a central role in the negotiation and we distinguish two phases in its decision process. In the first phase, the barge determines a convenient rotation based on information revealed by the terminals and information it already has about the network, such as sailing times. In the second phase, it makes appointments for the ‘best’ rotation found in the first phase. In the remainder of this paper, we discuss this in more detail. It is important to understand what we mean with an appointment.

An *appointment* made by a barge and a terminal is an agreement from two sides. The barge promises the terminal to be present at the terminal at a certain time, i.e., the latest arrival time. The terminal in turn guarantees the barge a latest departure time, if the barge keeps its promise. If the barge does not keep its promise and arrives later than the announced time, it has to make a new appointment. In making appointments, the barge uses the guaranteed latest departure times at preceding terminals to calculate the arrival time at the next terminal.

Note that barges in our model do not *estimate* the latest departure time (LDT), but use the *guaranteed* LDT to plan their rotation. Recall that no disturbances occur during operations and that sailing, mooring, and handling times are deterministic. This means that a barge is always able to arrive in time at a terminal, i.e., not later than the agreed latest arrival time. Penalty costs are therefore no issue. In a future study it might be interesting to let barges estimate the LDT instead of using the guaranteed LDT to plan their rotation. However, this probably requires the introduction of a penalty to force parties to meet agreements.

4.2 Performance indicators

In our model, terminals have fixed capacity, which means that the utilization degree of a terminal is determined by capacity requirements of the barges that visit this terminal during a certain period. To evaluate the system performance, we therefore focus on barge performance, which reflects the terminal performance as well. The more efficient terminals operate their quays, the smaller the service times at terminals and the shorter barges sojourn in the port. The key system performance indicators we use are:

1. *Fraction of barges leaving the port late.* Leaving the port late means that a barge leaves the port later than the time set in its sailing schedule. This indicator measures the fraction of barges that did not manage to leave the port in time.
2. *Average project tardiness.* The tardiness of a barge is equal to the maximum of zero and the *project lateness* of the barge. Project lateness is the difference between the actual time the barge leaves the port and the time set in the sailing schedule. Project lateness can be negative.
3. *Average activity tardiness.* If an activity (the loading or discharging of containers at a terminal) has a due date, we also measure the activity tardiness. The activity tardiness is the maximum of zero and the activity lateness. Activity lateness is the difference between the actual time an activity is completed and the due date of that activity. Activity lateness can be negative.
4. *Average project lateness.* To get an impression how ‘early’ or ‘late’ the barges leave the port on average, we measure also the average project lateness of all barges.

In this paper we consider both experimental settings in which activity tardiness is an issue and where it is not. For the off-line benchmark, we take a combination of indicators 2 and 3 as primary objective, and indicator 1 as secondary objective. In Section 8 we discuss this in more detail. The aim of the off-line benchmark is to get an impression of the solution quality of the Multi-Agent approach. Since it is hard to define one single objective for the whole system (given the different conflicting interests) we feel that the objectives of the off-line benchmark are reasonable, since they optimize the performance of all barges and terminals, regardless of the specific individual performance.

4.3 Levels of information exchange

Recall that actors in our problem are reluctant to share information, to prevent deterioration of their competitive position. To evaluate the value of sharing information, we consider different levels of information exchange in the first phase of the barge decision process. We define a *level of information exchange* as the extent to which a terminal gives insight to a barge in the occupation of the terminal during a certain horizon. The more insight a barge has in the occupation of terminals, the better it can determine a rotation minimizing its expected sojourn time in the port.

We consider three levels of information exchange:

1. *No information.* Terminals reveal no information about their occupation. The barge operator therefore determines its rotation by finding the shortest path along all terminals concerned.
2. *Yes/No.* A barge operator can ask terminals repeatedly whether a certain arrival time is convenient and the terminal replies only yes or no. To find a convenient rotation, the barge operator can delay terminal visits or visit terminals in different order. The basic idea is that barges can retrieve information about the occupation of terminals, but the information is very limited.
3. *Service-time profiles.* Terminals give barges information about the maximum service time given a certain arrival time. This information is provided for every possible arrival moment during a certain time horizon. After the barge operator has determined its ‘best’ rotation, it communicates its latest arrival times to the terminals.

The service-time profiles are an extension of the waiting profiles and are explained in more detail in Section 5. The concept of service-time profiles is similar to that of waiting profiles, with this difference,

that waiting profiles determine the maximum waiting time given a certain arrival time, whereas service-time profiles determine the maximum service time given a certain arrival time.

We like to mention again that the levels of information exchange relate to the first phase in the decision process of the barge (see Section 4.1), where a barge determines the ‘best’ rotation. The second phase in the decision process is independent of the level of information exchange used in the first phase. We implement the different levels of information exchange in the same way as Douma et al. (2008).

5 From waiting profiles to service-time profiles

In Douma et al. (2008) a waiting profile, issued by a terminal for a particular barge, denotes the maximum waiting time for a barge until the start of processing, for every possible arrival time during a certain time horizon. The basic assumption in generating waiting profiles is that the duration of the processing of a barge at the terminal is independent of the time at which the handling is started. This assumption holds for situations where terminals are never closed and only deal with barges. However, if the processing of a barge is interrupted by, e.g., the closing of the terminal, the processing time increases with the time the terminal is closed, since closing of the terminal is considered as preemptive downtime. This means that we are now confronted with *time dependent service times*.

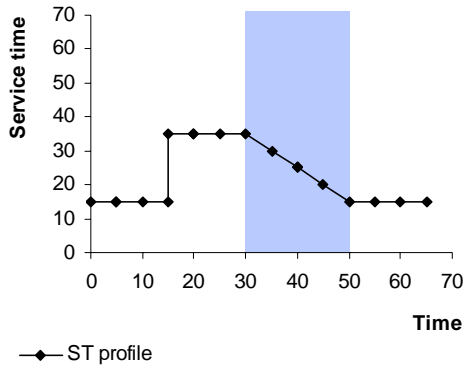


Figure 1: Example of a service-time profile for an incoming barge that needs 15 units of processing time. The terminal has planned no activity and is closed from $t=30$ to $t=50$.

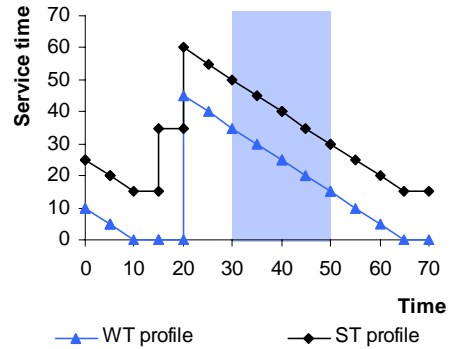


Figure 2: Example of a service-time profile for an incoming barge that needs 15 units of processing time. The terminal has planned already two barges (from $t=0$ to $t=10$ and from $t=55$ to $t=65$) and is closed from $t=30$ to $t=50$.

If the service time of a barge is time dependent, it is clear that waiting profiles alone do not offer enough information to barge operators. To be able to optimize a rotation, additional information about the service time needs to be revealed by the terminal. To keep the information exchange between terminals and barges as simple as possible, we propose to integrate both processing and waiting times in one profile, called a *service-time profile*. A service-time profile represents the guaranteed maximum time necessary to serve the barge, given a certain start time. A service-time profile or service-time function is also introduced by (Christiansen and Fagerholt, 2002) and can easily be generated if the terminal is not occupied. However, if the terminal has scheduled other barges as well, the service-time profile becomes more complex. We illustrate this with two examples.

In Figure 1 we give an example of a service-time profile for an incoming barge that needs 15 minutes of processing time. The terminal has planned no other activity and is closed from $t = 30$ to $t = 50$. In

Figure 2 we give an example, which is a bit more complex. It shows a service-time profile for an incoming barge that needs 15 minutes of processing time. The terminal has scheduled already two other barges, from $t = 0$ to $t = 10$ and from $t = 55$ to $t = 65$, and the terminal is closed from $t = 30$ to $t = 50$. Note that the service of the incoming barge cannot be preempted for the service of a yet scheduled barge. That is why the waiting time increases at $t = 20$.

6 The barge operator agent

In our model, the barge operator decides in which sequence a barge is going to visit the terminals. The algorithms introduced in Douma et al. (2008) have to be adapted slightly as a result of the change from waiting to service-time profiles and the introduction of closing times on containers. The objective of the barge operator agent in our experiments is a combination of activity tardiness and project tardiness. If we omit closing times on containers in our experiments, the objective of the barge becomes equal to the objective presented in Douma et al. (2008).

6.1 Model and notation

The barge operator agent has to make two decisions, namely i) in which sequence the barge visits the terminals and ii) the specific time each terminal is visited. We assume that every barge operator agent makes these decisions at the moment the concerning barge enters the port. We assume that the information of the terminals is reliable and does not change during the time the barge operator agent needs to make its decisions. It is important that barges make decisions in real time.

Let us consider one specific barge entering the port. The barge operator agent of this barge is assumed to have the following information. First, it knows the set \mathcal{N} of terminals that have to be visited. This set is a subset of all the terminals in the port. We assume that the agent knows the maximum service time θ_{it} , for every terminal $i \in \mathcal{N}$ and every arrival moment t . The maximum service time can be derived from the service-time profile and consists of waiting and handling time at the terminal. The handling time h_i consist of the time needed for loading and unloading of containers, as well as the mooring time on arrival and departure. Besides, the agent knows the sailing times s_{ij} between every pair of terminals (i, j) , where $i, j \in \mathcal{N}$. We assume that both the sailing and the handling time are deterministic. Let w be the planned port departure time set in the sailing schedule and c_i the earliest closing time of all containers that have to be (un)loaded at terminal i . Without loss of generality we assume $c_i \leq w$. Every barge enters and leaves the port via a *port entrance and exit point*.

The aim of the barge is slightly different from the objective in Douma et al. (2008). The barge operator has to weigh ‘sticking to the sailing schedule’ and ‘meeting the closing times of containers’, i.e., it has to weigh *project* tardiness against *activity* tardiness. The extent to which a barge operator agent can realize this objective depends both on the actual state of the port and the information it has about this. Optimizing its rotation, the barge operator assumes that the service time at the terminal will be equal to the time expressed in the service-time profile (which is actually a *maximum* service time). We can now model the barge operator agent by a *time dependent traveling salesman problem* (TDTSP). We define the time dependent travel time $\tau_{ij}(d_i) = s_{ij} + \theta_{j,a_j(d_i)}$, as the sum of the sailing and service time going from terminal i to j , where the barge departs at time d_i at terminal i . The aim of the barge is to find a rotation along all terminals it has to visit, such that the sum of the time dependent travel times is minimized. We assume that a barge visits a terminal only once.

6.2 Algorithms to solve the problem

To solve the TDTSP we use depth-first branch-and-bound for instances with at most seven terminals and the Dynamic Programming (DP)-heuristic of Malandraki and Dial (1996) for instances with more than seven terminals (see also Douma et al. (2008)). We adapt the objectives of both algorithms to be able to deal with activity and project tardiness. The change in objectives also require some changes in the DP-heuristic. We discuss the two topics in Section 6.2.1 and Section 6.2.2 respectively.

6.2.1 Objective

To deal with closing times of containers we have to adapt the objective function of the algorithms we use. Let ω_i^A denote the activity lateness at terminal i . If one or more containers at terminal i has a closing time, then $\omega_i^A = d_i - c_i$ and is zero otherwise. The project lateness is denoted by ω^P and is equal to the actual time the barge leaves the port minus the planned departure time w . To weigh project and activity tardiness we introduce $\gamma \geq 0$, the so-called project tardiness penalty. The primary and secondary objective function are then equal to

$$\begin{aligned} \text{primary objective:} & \quad \min \left[\gamma \cdot (\omega^P)^+ + \sum_{i \in \mathcal{N}} (\omega_i^A)^+ \right] \\ \text{secondary objective:} & \quad \min \omega^P \end{aligned}$$

So, we consider all possible scenarios and select the rotation that minimizes the weighed sum of the project and the total activity tardiness. If two or more rotations are equal to this respect, we minimize the project lateness, i.e., we prefer to let the barge leave the port as soon as possible.

6.2.2 Selection criterion in the DP-heuristic

The DP-heuristic of Malandraki and Dial (1996) constructs rotations by adding to each partial tour in the previous stage a (yet unvisited) node, which becomes the new end node of the partial tour. A state is a set of terminals that are in the partial tour, and an end node. The value of the state is the minimum time to travel from the depot, along all the terminals in the partial tour and ending in the end node. To limit the growth of the state space and to keep the computation time of the DP tractable, Malandraki and Dial (1996) retain in every stage the best H states, i.e., the states that most likely result in the optimal rotation. To select the best H states, the two objectives of Section 6.2.1 are not sufficient. Suppose we apply the two objective functions in the selection phase, then the algorithm will add terminals with a closing time at the end of the tour. Moreover, project tardiness is (most probably) zero at the start of the rotation construction, which means that in early stages of the DP all states have similar values. This is not desirable. We therefore use different selection criteria such that it is likely that a near-optimal solution is part of the H solutions retained in every stage.

The selection criterion we use is as follows. Suppose we are in a certain stage in the DP. Let \mathcal{N}^{inT} be the set of terminals that is part of the rotation at this stage and let \mathcal{N}^{NinT} be the terminals that still need to be added. Let the current duration of the rotation be denoted by t . The activity lateness ω_i^A at terminal $i \in \mathcal{N}^{inT}$ is calculated as we described above. The activity lateness $\hat{\omega}_i^A$ at terminal $i \in \mathcal{N}^{NinT}$ is equal to $t - c_i$ if one or more of the container that has to be transhipped at terminal i has a closing time, and is zero otherwise. The selection criterion we use is then given by:

$$\text{selection criterion: } \min \left[\gamma \cdot \omega^P + \sum_{i \in \mathcal{N}^{inT}} (\omega_i^A)^+ + \sum_{i \in \mathcal{N}^{NinT}} (\hat{\omega}_i^A)^+ \right]$$

By using this selection criterion we penalize postponing the visit of a terminal with a closing time to the end of the tour. In the selection phase we use project lateness instead of project tardiness, since this leads to a better distinction in the sojourn time of the port in early stages of the DP.

7 The terminal operator agent

In our model, the terminal operator decides about convenient times to process barges. In Douma et al. (2008) we argue that a terminal has two tasks, namely, making appointments and keeping appointments. However, the introduction of restricted opening times and sea vessels have a major impact on the way the terminal performs these tasks. In this section we briefly introduce the mathematical model and discuss adaptations in the algorithms used to plan the terminal quays and the construction of the service-time profiles.

7.1 Model and notation

Every terminal in the port has a terminal operator agent which has to negotiate with barges about convenient handling times. Every terminal has a set of, so-called, quays Q . A quay is a combination of resources that are all necessary to handle a barge, namely a berth, crane(s), and a team. A barge is assigned to one quay $q \in Q$ and every $q \in Q$ processes at most one barge at a time. Sea vessels, however, can be assigned to more than one quay. If a sea vessel needs more than one quay, the berths of the respective quays need to be adjacent. Without loss of generality, we assume that all quays are positioned in a line and are numbered in ascending order, starting from zero. This means that quay q_0 is only adjacent to q_1 , and quay $q_{|Q|}$ is only adjacent to $q_{|Q|-1}$. In general quay q_i is neighbored by q_{i-1} and q_{i+1} , where $0 < i < |Q|$. A terminal can be closed during certain periods of the day, which means that all quays $q \in Q$ are closed.

The terminal operator agent has two tasks. First, it has to make appointments and second, it has to keep its appointments. Making appointments concerns the construction of the service-time profile, indicating the preferred handling times of the barge. Keeping appointments is about scheduling the barges with which already appointments are made, such that no appointment is violated.

An appointment is an agreement between the barge and the terminal operator (see Section 4.1). By making an appointment, the barge promises the terminal to be present before a certain time, namely the *latest arrival time (LAT)*. Based on the service-time profile, issued in the first phase of the barge's decision process, the terminal derives the *maximum service time (MST)* for the LAT and guarantees the barge a *latest departure time (LDT)*. It holds that $LDT=LAT+MST$. After the appointment is made, the terminal schedules the barge tentatively such that the appointment with the barge is not violated, nor the appointments made with other barges. In the schedule (which is hidden for the barge) every barge has a *planned starting time (PST)* and an *expected departure time (EDT)*. The EDT is calculated based on the planned starting time, the processing time (PT), and the opening times of the terminal. The PT is equal to the handling time (h_i) of the barge, as introduced in Section 6.1 and is revealed by the barge in the first phase of its decision process.

To give an illustration, consider the agreements made with barge B1 in Table 1. Barge B1 has promised to arrive not later than $t = 5$ and it has a guarantee that its processing will be completed no later than time $t = 25$ (if it arrives in time). Hidden for the barge is the schedule of the terminal, denoting the PST, PT and EDT, respectively 5, 15 and 20 for this barge. Barge B2 has made similar appointments as shown in Table 1. In the example, the slack in the appointments is 5 to 10 minutes.

Barge	<i>Agreements with the barges</i>		<i>Schedule (hidden for the barges)</i>		
	Latest arrival time (LAT)	Latest departure time (LDT)	Planned starting time (PST)	Processing time (PT)	Expected departure time (EDT)
B1	5	25	5	15	20
B2	55	75	50	10	60

Table 1: Example of a quay schedule and the corresponding agreements

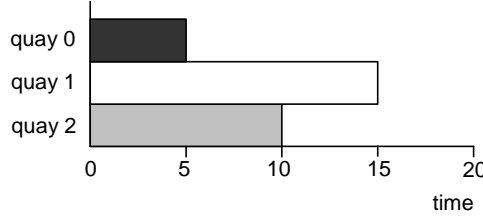


Figure 3: Example of a quay schedule of a terminal. The bars represent the scheduled time for sea vessels at each quay.

7.2 Keeping appointments

The keeping appointments task is the scheduling of *expected barges* on the quays such that no appointment is violated. Expected barges are barges that already made an appointment with the terminal and still need to be processed (see Table 1 for an example). From time to time, it might be beneficial for a terminal to reschedule barges on the quays, e.g., when a barge arrives earlier than its latest arrival time. This is possible when a barge does not need to wait the maximum waiting time at previous terminal(s).

To schedule the expected barges, the terminal has to decide how the barges are assigned to the available quays, such that all appointments are met, i.e., the handling of no barge is finished later than its LDT. Finishing the handling of a barge later than its LDT is not permitted in our model. By scheduling the barges, the terminal has to take into account the *sea vessels* that arrive in the near future. For every sea vessel the terminal knows the number of quays it needs and the amount of containers it needs to load and discharge. We assume that sea vessels are known a priori to the planning horizon, or at least so early that that no conflict arises with yet scheduled barges. We assume that sea vessels are scheduled in the order of arrival.

We schedule sea vessels in the following way. Let τ_j be the the departure time of the last scheduled sea vessel on quay q_j , k the number of quays required for the sea vessel that is to be scheduled, and t^a the arrival time of the sea vessel. A sea vessel is then assigned to a group of quays q_i, \dots, q_{i+k-1} for which the most time has elapsed after the last scheduled sea vessel, i.e., $\max_{i=0, \dots, |Q|-k} \min_{j=i, \dots, i+k-1} [t^a - \tau_j]$. Suppose we have to schedule a sea vessel arriving at time $t^a = 20$, requiring two quays, and that we have already a schedule as given in Figure 3. The sea vessel is now scheduled on quays q_0 and q_1 .

The primary objective for the terminal for the scheduling of the barges is to minimize the maximum terminal lateness and the secondary objective is to minimize the average *terminal lateness* of all expected barges. We define the terminal lateness of a barge as the expected departure time of this barge minus its latest departure time. Remark that the terminal lateness always needs to be less than or equal to zero, since all appointments with barges need to be met. The algorithm we apply to obtain a schedule is list-scheduling in combination with depth-first branch-and-bound (see Schutten, 1996). The algorithm is adapted to deal with sea vessels and restricted opening times, since these events impact the start and service time of the barge. The basic idea of the list scheduling algorithm is to make an ordered list of

barges and to assign these barges successively to the first available resource. A good initial solution for the branch-and-bound algorithm is usually to sort barges based on latest departure time (LDT), which results in a strong bound and makes the algorithm fast.

7.3 Making appointments: how to construct the service-time profile

To construct a service-time profile we need to determine i) the possible start times of the barge and ii) the service time given a certain start time. For the former we use the concept of waiting profiles as introduced in Douma et al. (2008), and for the latter we create a service-time function. The waiting profile and the service-time function are then merged into a service-time profile. We explain the construction of the waiting profile, the service-time function, and the service-time profile in this section.

In this section we use the following example. Assume that we have a schedule as given in Table 1 and that we have to make a service-time profile for barge b with processing time $PT = 15$. In this example the terminal is closed from $t = 30$ until $t = 50$.

7.3.1 Construction of the waiting profile

The construction of the waiting profile is only depending on the non-preemptive events, which in our model are barges and sea vessels. We have to extend the procedure in Douma et al. (2008) to construct a waiting profile that is compatible with sea vessels and restricted opening times of terminals. We proceed as follows. First we determine all *start intervals*, by considering all possible insertion points i in the quay schedule. Insertion point i corresponds with insertion after the i^{th} ship (barge or sea vessel) in the schedule ($i = 0$ means insertion before all scheduled ships). We define a start interval as a time interval in which the handling of the concerning barge can be started such that no appointment with other barges or sea vessels is violated when the processing of the barge is completed. Based on the start intervals we create a quay waiting profile analogous to the procedure in Douma et al. (2008), and based on all the quay waiting profiles we construct a terminal waiting profile. It is important that in this stage no slack is added to the waiting profile. We describe the way the start intervals are determined in more detail.

To determine all the start intervals we take two steps. The first step is similar to the start interval procedure described in Douma et al. (2008). For all insertion points i we determine a start interval in the following way. We schedule all barges and sea vessels before insertion point i as early as possible, and all barges and sea vessels after insertion point i as late as possible. Let m_i and n_i be the respective start and end of start interval i . Then m_i becomes equal to the EDT of ship i (the last scheduled ship (barge or sea vessel) before insertion point i), and n_i is equal to the $PST_{i+1} - PT$. If $n_i < m_i$ then the start interval is empty and we do not consider it anymore. If $m_i \leq n_i$ then we remain the start interval.

Once we have considered all possible insertion points i in the quay schedule, we proceed to step two. In this step we evaluate whether the processing of a barge is interrupted by the closing of the terminal when we start the handling of a barge during start interval i . This could mean that we have to adjust start interval $[m_i, n_i]$ resulting in a new start interval $[m'_i, n'_i] \subset [m_i, n_i]$. If $n'_i < m'_i$ then the adjusted start interval is empty and we do not consider it anymore. Otherwise we maintain the start interval.

In Figure 4 we give an illustration of a start interval $[m'_i, n'_i]$ that has to be adapted because of closing of the terminal. In Figure 4 we denote the interval in which the terminal is closed with $[e^s, e^c]$. We see that in the example $m_i < e^s < n_i < e^c$. Assuming that we derived the start interval $[m_i, n_i]$ in the way we describe above, then $n_i = PST_{i+1} - PT$. However, if we start barge b in interval $[e^s, n_i]$ the completion time of barge b will be greater than PST_{i+1} , the planned (and latest) starting time of barge

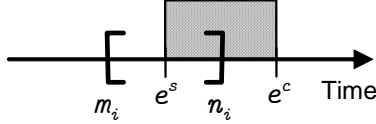


Figure 4: An example how a start interval can be related to closing of the terminal.

$i + 1$. This means that we have to decrease n_i such that $n'_i + PT + e^c - e^s \leq PST_{i+1}$. Rewriting the expression gives $n'_i = n_i + e^s - e^c$. The example is just one illustration of how a start interval has to be adapted. In practice all other kind of (nested) situations can exist.

To give an illustration, consider the example in Table 1 and our barge b which requires 15 minutes processing time. After the first step in our procedure we find the start intervals as given by Table 2. A start interval i corresponds with insertion point i .

Start interval	Start time	End time	Insertion point
1	20	50	1
2	65	∞	2

Table 2: The start intervals in the example

In the example it is clear that start interval one overlaps with closing of the terminal, which means that we possibly have to adjust the interval. However, in this situation the start interval can be maintained, since the barge handling can be completed without any problem before the next scheduled barge. During the interval the service time will vary. The terminal in our example has one quay and the resulting terminal waiting profile is depicted in Figure 5.

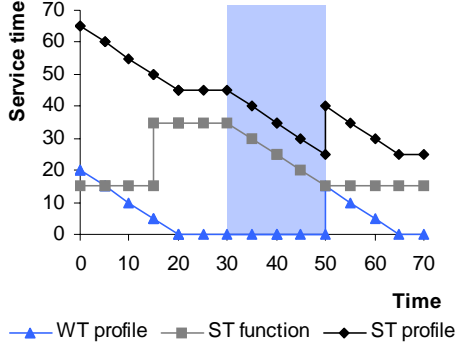
7.3.2 Construction of the service-time function

The construction of the service-time function is only depending on the preemptive events, which in our model is only the closing of the terminal. Since most terminals close in repetitive patterns, the service-time function can easily be constructed based on the opening times of the terminal. We apply the following procedure to construct the service-time function. Suppose the closing period of the terminal is repeated every T time units, called the *cycle length*, e.g., a day. Then for a sufficiently long horizon (nT , $n \in \mathbb{N}$) we add the following points to the service-time function $s(t)$. For opening of the terminal at time t^O we add $s(t^O) = PT$, with slope 0. Closing of the terminal results in two points, namely closing of the terminal at time t^C with corresponding $s(t^C) = PT + t^O - t^C$ and slope -1, and a point $t^C - PT$ with corresponding $s(t^C - PT) = s(t^C)$ and slope 0.

7.3.3 Construction of the service-time profile

To determine the service-time profile we evaluate, starting from the current time t_0 , for all possible arrival times t in the time interval $[t_0, nT]$ the corresponding maximum service time. The maximum service time $\tilde{S}(t)$ at time t can be calculated by $\tilde{S}(t) = w(t) + s(t + w(t)) + s$, with $w(t)$ the waiting time at time t given by the waiting profile, $s(t)$ the service time, given by the service-time function, and s the amount of slack that is added to the maximum service time to increase scheduling flexibility. The reason this function holds, is that at time $t + w(t)$ a barge can be started by definition, since $w(t)$ is the

time till opening of the next start interval, and a start interval has the property that the barge handling be started without violating any other appointments. Besides the maximum service time we also have to determine the slope of the function. The slope of the function is -1 if $w(t) > 0$, and is equal to the slope of the service-time function otherwise.



Time	Max service time	Slope
0	65	-1
15	50	-1
20	45	0
30	45	-1
50	40	-1
65	25	0

Figure 5: Example of a waiting (WT) profile, service-time (ST) function, and the (ST) profile

Table 3: Data representation of the service-time profile of Figure 5

The service-time profile of the example presented at the start of this section is given in Figure 5. The corresponding data representation is presented in Table 3. In the service-time profile we included 10 time units slack.

8 Centralized control

To get an impression of the solution quality of the multi agent approach, we compare it with the situation in which we know all information in advance, i.e., we know a priori the arrival times of barges, the terminals each barge has to visit, the number of containers a barge has to (un)load, the arrival times of sea vessels and their processing time, the terminals and their capacity, and the sailing times in the network. We call this situation the off-line problem. In this section we describe the model and the solution methods we apply.

8.1 Basic model

We model the off-line problem as a Resource Constrained Project Scheduling Problem (RCPSP; see, for example, Demeulemeester and Herroelen, 1992), as a base model. In the classical RCPSP, a single project has to be scheduled, which consists of a number of activities. These activities have to be processed on a number (not necessarily all) of the available resources, consisting of a number of parallel processors. If an activity is processed at a resource, it can use one or more processors during its processing. Between activities precedence relations exist. Precedence relations determine the sequence in which activities have to be processed, i.e., an activity cannot start before all its predecessors have been finished.

We represent the terminals by resources in the RCPSP. The number of processors for each resource is equal to the number of quays of the associated terminal. In this way we model that a terminal can handle two barges at the same time if it has two quays. Each terminal visit of a barge is represented as an activity. The processing time of an activity is equal the handling time of the associated terminal visit.

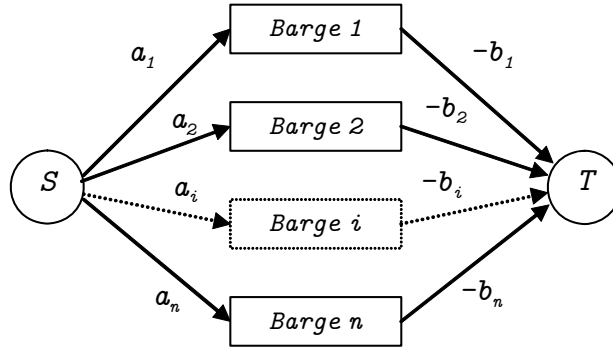


Figure 6: Barges are denoted as parallel projects in one super project that start with dummy node S and end in dummy node T .

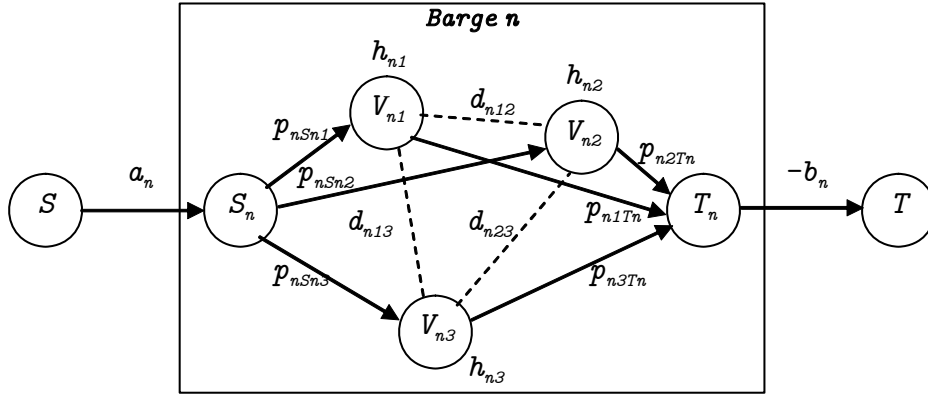


Figure 7: A barge is represented as a project consisting of different activities, starting in dummy node S and ending in dummy node T .

Barges are represented as independent projects within one super-project (see Figure 6). The super project start node precedes all activities of each barge and the super project end node succeeds all activities of each barge. From the start node an arrow goes to all barges (all projects) with an associated weight a_i , denoting the arrival time of barge i in the port. From every project an arrow goes to the super project end node with an associated weight $-b_i$, with b_i the due date of barge i . By modeling it this way, the length of a path from S to T (the dummy source and sink node, respectively) is equal to the lateness of the corresponding path. This enables us to compare the time a barge actually leaves the port to the time the barge has to leave the port according to its sailing schedule. We can use this to solve the problem by, e.g., *minimizing the maximum lateness* or *minimizing the total tardiness* of all paths from S to T , i.e., for all barges (projects).

If we focus on one barge (one project) we get a picture analogous to Figure 7. Every barge project starts in dummy node S_n , which precedes all activities of barge n . An activity is denoted by V_{ni} , i.e., the processing of barge n at terminal i . The associated handling time of activity V_{ni} is h_{ni} . From dummy node S_n an arrow goes to all activities of barge n , with weight p_{nS_ni} denoting the sailing time from the start node S_n (denoting the entrance point of the port) to terminal i . Similar arrows go from all activities

to the dummy end node T_n (denoting the port exit point of the barge). The weights on these arrows are p_{njT_n} , i.e., the sailing time from terminal j to the port exit point T_n . Between activities a dashed lined is depicted, with associated cost d_{nij} , denoting the sailing time for barge n to go from terminal i to j . We assume that the sailing time from $i \rightarrow j$ is equal to the sailing time from $j \rightarrow i$. During the scheduling of activities the dashed lines have to be given a direction to indicate the sequence in which the activities are performed. The reason is that the sequence in which activities are performed is not determined beforehand, but is part of the decision. To model this, we introduce for each barge a dummy resource that has capacity one and is required by each activity associated with this barge.

We represent opening times of terminals (resources) by so-called resource profiles. Since we assume that the arrival and processing times of sea vessels are deterministic, we choose to model the sea vessels in the resource profiles as well. To deal with closing times of containers we have to register both the activity tardiness and the project tardiness. As objective function for the RCPSP we take a combination of project and activity tardiness.

In the basic RCPSP, the objective is to minimize the makespan, i.e., the time to complete all activities in the project. For our problem, the primary objective is minimizing the *total project tardiness* plus *the total activity tardiness*. As secondary objective we minimize the fraction of barges leaving the port late. We weigh the project tardiness with a project tardiness penalty $\gamma \geq 0$ (see Section 6.2.1). Remark that if no container has a closing time and when $\gamma = 1$, we get the same objective function as in Douma et al. (2008).

8.2 Solution method and implementation

To solve the resulting problems, we use a randomized construction algorithm that is based on the adaptive search algorithm by Kolisch and Drexel (1996). In this algorithm, a (large) number of schedules is generated using a randomized priority rule scheduling heuristic. For the basic RCPSP, this heuristic finds schedules that are close to optimal very fast. We use a parallel generation scheme (Kolisch, 1996) to construct 34.000 schedules (iterations). After every 2.000 iterations we decrease the value of parameter alpha. The construction of a schedule is stopped if the value of the objective function of the (partial) schedule is not better than the best found objective value so far. We use the following values for alpha, $\alpha \in \{20, 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1, 0\}$. We start the random sampling procedure with the highest alpha.

Experiments revealed that our solution method appears not to be satisfying anymore if we deal with non-preemptive downtime, e.g., caused by sea vessels. We have tried several priority rules, but were not able to find a rule that weighs in a proper way the scheduling of an activity in the current stage, and the possible consequences of scheduling the activity in a later stage. The effect of a poor decision can be quite significant, especially when containers have a closing time. We therefore apply, after the running adaptive search algorithm, a 1 and 2-opt procedure by moving and swapping activities.

9 Experimental settings

To evaluate the Multi-Agent System for different levels of information exchange, we consider different scenarios and compare the results with the off-line scheduling algorithm. To obtain insight in the functioning of the Multi-Agent System, we consider also other port configurations besides the Port of Rotterdam. In this section, we describe the scenarios, the variables, and the parameters. Recall that all handling and sailing times are deterministic. As time unit we use minutes in our experiments.

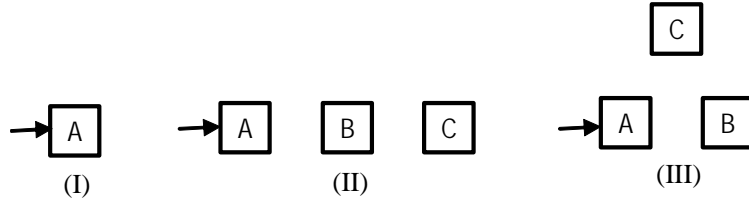


Figure 8: Three network layouts; one region, three regions in line, and three regions in a triangle. The arrows represent the port entrance and exit point.

9.1 Introduction

We consider different experimental settings, each divided in different scenarios, to evaluate the performance of the different levels of information exchange. We consider the effect of restricted opening times of terminals and unbalanced network settings in separate experimental settings. We do not consider the effect of introducing sea vessels separately, since we expect that the effect will be similar to introducing restricted opening times. Instead, we consider an experimental setting with restricted opening times of terminals, unbalanced network settings, sea vessels, and closing times of containers. In future research one might consider also other experimental settings, like the effect of introducing sea vessels.

In the remaining sections we discuss the experimental settings in more detail. There are however a number of settings which are similar for all experimental settings, like the network layout (including sailing times between terminals), several parameter settings and distributions, and the time window of a barge. These common settings are similar to the experimental settings in Douma et al. (2008), unless specified otherwise in the description of the experimental settings. We describe briefly two of the common settings, namely the network layout and the time window of a barge. For more details we refer to Appendix A.

We consider three different network layouts (see Figure 8), which are inspired by the geographical structure of large ports around the world. Layout I is added to evaluate the effect of regions on the performance of the system. The time window of a barge is the time a barge is supposed to be in the port according to its sailing schedule. We consider *fixed* and *variable* time windows. Fixed time windows mean that all barges have the same time window, irrespective of the length of their rotation. The length of the time window is based on the expected sailing and handling time of an average barge increased with a certain amount of slack. Variable time windows mean that the length of the time window of a barge is depending on the sailing and handling time of a specific barge, increased with a fixed amount of slack (y) and a variable amount of slack per terminal (x).

9.2 Setting 1: Restricted opening times of terminals

For Experimental Setting 1 we generate 36 scenarios, varying along the following dimensions: the number of terminals per region (either four or nine terminals), three different network layouts (see Figure 8), three different utilization degrees of the network (50, 75, and 90%), and variable and fixed time windows for the barges. In all experimental settings, half of the terminals in a region (rounded down to the nearest integer) are closed during the night (6 pm - 6 am) and the rest is opened 24h a day. To realize the desired utilization degree, the number of barges is reduced for the terminals that close during the night. The utilization degree of a terminal is calculated based only on the time the terminal is open. The slack factor used for the fixed time window is 1.8. The fixed slack y (slack per rotation) for the variable time

window is equal to 0.5, 1.0, and 1.5 for the different utilization degrees of 50, 75, and 90% respectively, and the variable factor x (slack per terminal) is equal to 10% for all utilization degrees. The call size at a terminal is drawn from a normal distribution with mean 30 containers and a standard deviation of 10 containers. We evaluate all the scenarios for all levels of information exchange (Section 4.3). For every scenario, we performed 10 replications and every replication has a length of 100 days.

Besides we perform a sensitivity analysis to analyze the impact of i) the distribution of terminals that are closed during the night over the different regions and ii) the time the terminals are closed. Aim of the experiment is to get insight in the impact of closing times on the performance of barges and terminals.

9.3 Setting 2: Unbalanced network settings

For Experimental Setting 2 we generate six scenarios, varying along the following dimensions: three different network variants (see Figure 8), and a variable and fixed time window for the barge. Terminals have a different number of quays, different utilization degrees, and different processing times. We distinguish four terminal types (see Table 4).

Terminal type	#Quays	Utiliz. degree	Mean call size	Stdev call size
Alpha	1	50%	15	5
Beta	2	50%	15	5
Gamma	3	75%	40	20
Delta	6	90%	60	30

Table 4: Different types of terminals, with corresponding number of quays, utilization degree, and average and standard deviation of the number of containers handled for every barge

The call sizes are drawn from a normal distribution. We simulate different network layouts with specific configurations of terminal types, as given by Table 5. The reason we have chosen this distribution of terminals over the regions is that it is comparable with the Port of Rotterdam. We have two relatively quiet regions and one very busy region (region C) in the network layouts with three regions.

Network layout - region/ terminal type	I	II and III		
	A	A	B	C
Alpha	6	6	8	3
Beta	2	1	1	3
Gamma	1	2	0	2
Delta	0	0	0	1

Table 5: The number of different terminal types per network layout

Terminals are opened 24h a day in all experiments. The slack factor used for the fixed time window is 0.75. The fixed slack y (slack per rotation) for the variable time window is equal to 0.5 and the variable slack x (slack per terminal) equal to 0.03. All experiments are replicated 10 times and each replication has a length of 75 days.

To gain additional insight we also simulate the scenarios using service-time profiles with varying amounts of slack per terminal. We consider three options (1, 2, 3) and vary the slack depending on the type of terminal as denoted in Table 6. The choice for the slack values is based on previous experiments (see Douma et al. (2008)).

Aim of the experiments is to get insight in the effect of an unbalanced network setting on the performance of barges and terminals and the performance of different controls in different settings.

Terminal type	Slack option		
	1	2	3
Alpha	0	0	0
Beta	0	0	0
Gamma	30	30	30
Delta	30	60	90

Table 6: The amount of slack for different terminal types

9.4 Setting 3: Unbalanced network settings, restricted opening times terminals, sea vessels and closing times containers

Experimental Setting 3 is an extension of Experimental Setting 2, with respect to the introduction of restricted opening times, sea vessels, and closing times of containers. The network configurations are similar to Table 5. However, the terminal types have some additional properties. The alpha terminals are closed from 6pm - 6am and do not handle sea vessels. The other terminal types are opened 24h a day and do handle sea vessels. The fraction of time they work on sea vessels per day is 40%. sea vessels are handled at one quay, like barges.

The processing time of sea vessels is based on historical data of 23.000 sea vessels that visited the Port of Rotterdam at the major terminals in the period 2005-2007. Based on this data we decided to apply a beta distribution with parameters β_1 and β_2 , equal to 1.14 and 8.3, respectively. The corresponding mean and standard deviation is equal to 770 and 645 minutes. Based on these settings, we can calculate the number of sea vessels that have to arrive during a certain time period in order to realize that terminals work 40% of their time on sea vessels.

Based on the same historical data, we assume that sea vessels arrive with exponentially distributed interarrival times in the port. This is to be expected as liner shipping alliances determine their sailing schedules independently. Moreover, the assumption is in accordance with results found by (Kuo et al., 2006).

Sea vessels are generated per terminal. The mean interarrival time depends on the number of sea vessels the terminal has to process to realize the utilization degree we aim for. If a sea vessel is about to arrive during processing of another vessel and a terminal has no capacity available to process the vessel, we delay the arrival of the sea vessel until the handling of the first sea vessel is completed. The arrival time of the next sea vessel, however, is calculated using the initial arrival time of the previous vessel.

Closing times of containers are assigned uniformly to 20% of the activities of a barge that are handled at terminals where also sea vessels are handled. A closing time is drawn uniformly distributed from the time window of the barge (both for fixed and variable time windows).

A full-factorial analysis (varying all parameters in all possible combinations) is out of the scope of our study, since the simulation effort is computationally intractable. Considering the experimental results in previous experiments we have chosen the following settings. First, we use network layout II, since the results of the other network layouts generally follow similar patterns. We consider all levels of information exchange considered in Experimental Setting 1, plus ‘Slack Option 2’ (see Table 6) introduced in Experimental Setting 2. We vary the penalty for the project tardiness $\gamma \in \{0, 10\}$. We simulate 30 replications for every scenario, and every replication has a length of 100 days.

10 Results

In this section, we present the results and insight obtained after simulating all the experimental settings described in Section 9. We describe the results of Experimental Settings 1, 2, and 3 respectively.

10.1 Setting 1 - Restricted opening times of terminals

The performance of different levels of information exchange in case terminals have restricted opening times is comparable to the results presented in Douma et al. (2008). We therefore depict only the comparison with the off-line benchmark in terms of average tardiness and average lateness (see Figure 9 and 10).

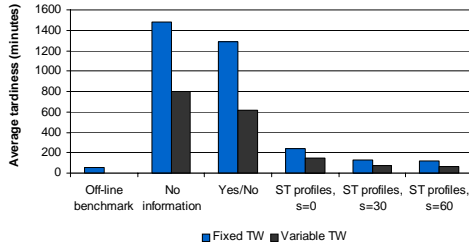


Figure 9: The average project tardiness per barge averaged over all considered experiments, compared with the off-line benchmark

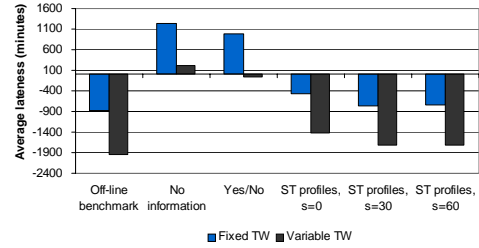


Figure 10: The average project lateness per barge averaged over all considered experiments, compared with the off-line benchmark

If we look at the way barges deal with restricted opening times for different levels of information exchange, we see that in case of service-time profiles, barges avoid to visit terminals with restricted opening times at the end of the day, i.e., right before the terminal closes. This holds especially for lower utilization degrees. The reason is that barges limit the risk to stay overnight at a terminal which would lead to a significant longer sojourn time in the port. In Figure 11 and 12, we depict the average fraction of nights barges were present at a terminal (with restricted opening times), averaged over all terminals and all scenarios, and specified for a utilization degree of 50 and 90%. Besides we denoted the fraction of times a terminal started processing a barge at opening of the terminal. It is clear that with service-time profiles communication barges arrive during the night. This means in practice that terminal with restricted opening times will see a reduced occupation of their quays at the end of the day.

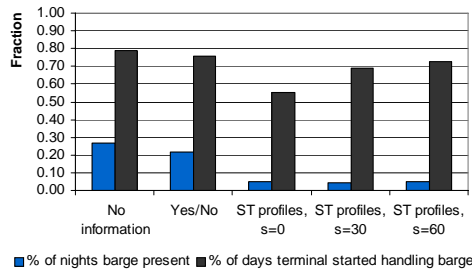


Figure 11: Fraction of barges during the night present at the terminal and fraction of barges for which the processing has been started at the start of the day. Results for different communication protocols, averaged over all scenarios with a 50% utilization degree

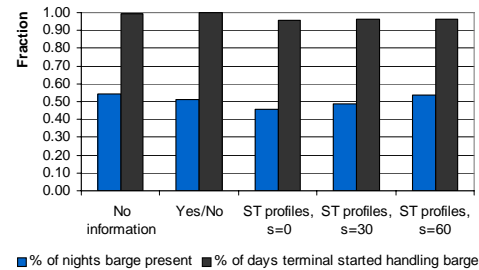


Figure 12: Fraction of barges during the night present at the terminal and fraction of barges for which the processing has been started at the start of the day. Results for different communication protocols, averaged over all scenarios with a 90% utilization degree

The sensitivity analysis reveals that an increase of the duration of the closing of the terminals leads to an increase of the average sojourn time of barges in the port. The extent to which the sojourn time increases depends on the way the terminals (with restricted opening times) are distributed over the different regions, and on the utilization degree of the network. If terminals (with restricted opening times) are equally distributed over the regions or the network utilization is low, then the sojourn time increases linearly in the duration of the closing. If terminals are divided unequally over the regions and the network utilization is high, then we see that, the longer the terminal closes, the greater the impact is on the sojourn of barges. The reason for the increase in sojourn time is probably an increase in variability in the system. From the factors determining the cycle time of a product, namely the variability, the utilization degree and the processing time (Hopp and Spearman, 2000), only the variability is affected.

10.2 Setting 2 - Unbalanced network settings

We now look at the results for Experimental Setting 2, regarding the unbalanced network settings. In Figures 13 and 14 we depicted the performance of the different communication protocols on the simulated scenarios. We omit the comparison with the off-line benchmark, since we could only calculate a limited time horizon (about eight days) with the off-line benchmark which is not realistic anymore if we also have to apply a warm-up and cool-down period. From the results it is clear that waiting profiles communication outperforms the other levels of information exchange.

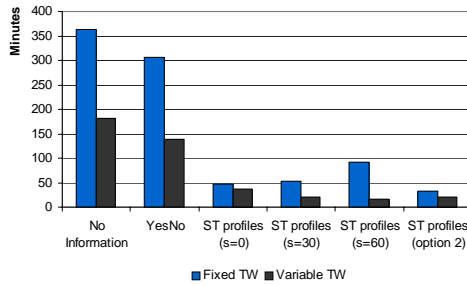


Figure 13: Average project tardiness per barges, averaged over all scenarios and specified for fixed and variable time windows.

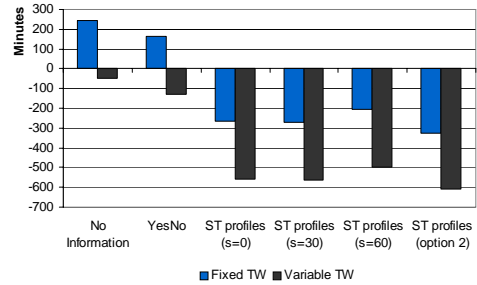


Figure 14: Average project lateness per barges, averaged over all scenarios and specified for fixed and variable time windows.

Besides the regular amount of slack added to the service-time profiles (0, 30, and 60 minutes), we also considered options in which terminals add a different amount of slack (see Table 6). In Figures 13 and 14 we depicted also the results of *slack option 2*, which performs not always better in terms of project tardiness, but does lead to lower project lateness. However, the performance of all three options of Table 6 are very similar and (in case of fixed time windows) clearly better than when we let every terminal add the same amount of slack to their service-time profile. We observe that the average waiting time at a terminal can decrease significantly by introducing a certain amount of slack. Note that waiting time at a terminal is not depending on fixed or variable time windows, since barge do not take these time windows into account in their decision, except for the yes/no information exchange. In Table 7 we show an example of the average waiting time per terminal type for the alternative amounts of slack in a line network layout. The 0, 30, and 60 minutes slack mean that all terminals add this amount of slack to the service-time profile. Slack options 1, 2, and 3 are described in Table 6.

In general it would be interesting to know which amount of slack results in the best performance of the system. Based on the simulated data sets it is hard to draw a conclusion about that, however, we expect

Terminal type	slack in min.			slack option			No information
	0	30	60	1	2	3	
alpha	19	19	24	15	14	15	63
beta	11	15	21	10	10	11	28
gamma	38	41	60	32	30	31	164
delta	141	155	183	129	117	118	519

Table 7: Average waiting time per terminal type for alternative slack policies. The letters A, B, and C refer to the options mentioned in Table 6

that there is a relation between average waiting time at the terminal and the best amount of slack to add to the service-time profile.

10.3 Setting 3 - Unbalanced network settings, restricted opening times terminals, sea vessels and closing times containers

Experimental Setting 3 is not easy to analyze, since it includes the extensions we considered in previous experimental settings. If we look at the average performance of the different levels of information exchange, we find the same pattern as seen before. We therefore focus on the comparison with the off-line benchmark. In Figure 15 we depict the performance of different controls, in terms the average activity tardiness, for a project tardiness penalty of zero. We only depicted *slack option 2* as slack policy for the waiting profiles, since the other slack policies lead to similar (though slightly worse) results. From the picture it is clear that the off-line benchmark outperforms the other controls. Figure 16 is similar to Figure 15, but now we depict the average project tardiness for a project tardiness penalty of ten. Also here we see that the off-line benchmark outperforms the other controls, especially for variable time windows.

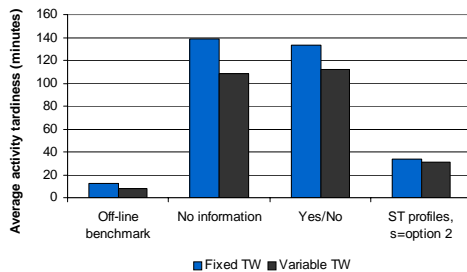


Figure 15: Average activity tardiness for different levels of information exchange, compared with the off-line benchmark. Results for project tardiness penalty equal to zero.

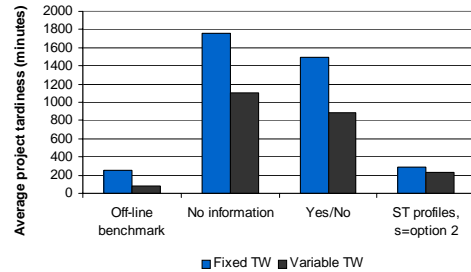


Figure 16: Average project tardiness for different levels of information exchange, compared with the off-line benchmark. Results for project tardiness penalty equal to ten.

Analysis of the results reveals that raising the project tardiness penalty from zero to ten does not significantly affect the average project tardiness, but reduces the average activity tardiness slightly with about 5%. The explanation for the fact that project tardiness is not significantly affected is that some terminals function as bottleneck, which means that the sojourn time of a barge is mainly determined by the waiting time at this terminal. In Figure 17 and 18 we depicted the waiting time announced by terminals at time t_0 (the time a barge plans its rotation), related to the position of this terminal in the rotation of the barge. We plotted this for different rotation lengths (1, 5, 10, and 15). Besides we have plotted the average duration of different rotation lengths. We conclude the following. First, it is clear that terminals with high waiting times at time t_0 are usually visited at the end of the rotation. Second,

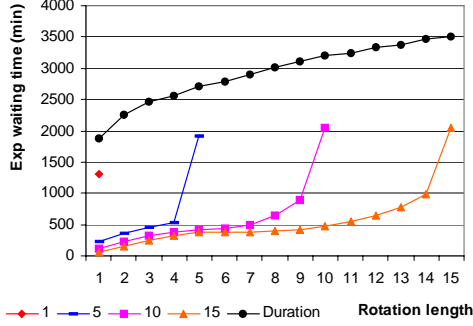


Figure 17: Average waiting time at t_0 related to the position of the terminal in the rotation. For comparison we also plotted the average duration for different rotation lengths (1, 5, 10, and 15) and a project tardiness penalty of zero

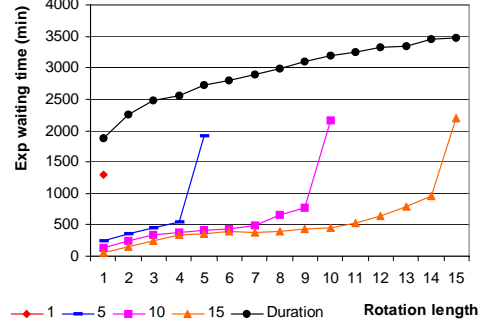


Figure 18: Average waiting time at t_0 related to the position of the terminal in the rotation. For comparison we also plotted the average duration for different rotation lengths (1, 5, 10, and 15) and a project tardiness penalty of ten

the waiting time at this terminal determines to a large extent the total sojourn time of the barge in the port. Third, we see that most terminals on average have some waiting time at time t_0 of about 400 minutes. This is probably due to the restricted opening times and sea vessels that block capacity for a significant amount of time. The fact that activity tardiness is only reduced by about 5% is probably caused by the fact that barges plan rotations successively and have very limited possibilities to reduce the activity tardiness, unless it is possible to swap activities of two barges at the same terminal. Moreover, terminals do not take into account the due date of a barge in their scheduling procedure. If terminals would give the barges with a due date a higher priority, the average tardiness might be reduced.

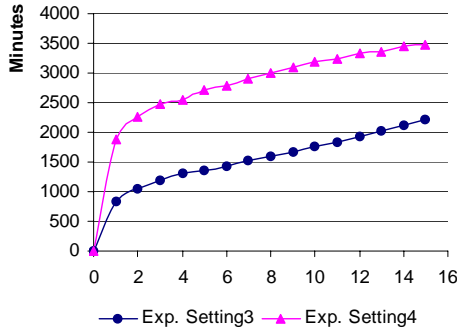


Figure 19: Average sojourn time of barges in Experimental Setting 3 and 4, related to the length of the rotation.

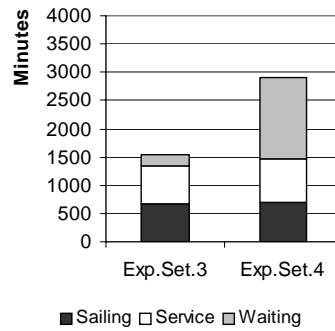


Figure 20: Average service, sailing, and waiting time for a rotation with 7 terminal visits. Results for Experimental Setting 3 and 4.

Figure 19 depicts the average sojourn time of barges for different rotation lengths in Experimental Setting 2 and 3. It is clear the sojourn of time of barges increases significantly as a results of sea vessels, restricted opening times, and closing times of containers. The reason for this increase is mainly an increase in waiting time at terminals, as indicated by Figure 20. Figure 20 shows the average amount of sailing, service, and waiting time in an average rotation (seven terminals) in Experimental Settings 2 and 3.

11 Conclusions

In this paper we look at the barge rotation planning and quay scheduling problem. This problem is about the alignment of barge rotations (sequence of terminal visits) and terminal quay capacity. In line with Douma et al. (2008) we use a Multi-Agent based approach for the problem, since a Multi-Agent system can mirror to a large extent the way the business network is currently organized and can provide a solution that is acceptable for each of the actors involved. The results presented in Douma et al. (2008) indicate that a communication protocol based on waiting profiles is promising.

In this study we consider practical extensions to make the model suitable for a realistic situation. These extensions are restricted opening times of terminals, unbalanced networks, sea vessels, and closing times of containers. It turns out that waiting profile communication is not sufficient in these situations. The reason is that, besides waiting time, also the service time of a barge becomes time-dependent, e.g., when the barge handling cannot be finished before closing of the terminal. We therefore propose service-time profiles, which are an extension of the waiting profiles and include time-dependent service times.

The change from waiting to service-time profiles, as well as the other practical extensions, require some adaptations of the barge and terminal operator agent algorithms. We explain, e.g., how a service-time profile can be constructed and how barge operator agents can deal with closing times of containers. We perform different experiments to gain insight in the effect of a certain extension, such as restricted opening times, on the performance of barges and terminals. We compare the results with an off-line benchmark.

The experiments reveal that also in a more realistic setting service-time profiles in general perform well. If we look more specifically at the results, we see that if terminals have restricted opening times, barges prefer not to visit these terminals when the terminals are about to close. The reason is that they want to reduce the risk to stay overnight at the terminal. If the utilization degree of the terminals increases, however, then more barges tend to stay at a terminal during the night. We find that the longer the terminals are closed, the longer the sojourn time of barges in the port. In case of unbalanced networks, we see that an increase in total capacity of the network reduces the average sojourn time of barges. Moreover, the average waiting time at a terminal can be reduced further when terminals add different amounts of slack to their service-time profile. If we introduce also sea vessels and closing times of containers, then we find that the Multi-Agent based approach based on service-time profiles performs well compared to other levels of information exchange. However, the approach has some difficulty to minimize the activity tardiness. The reason is that barges plan rotations successively and have no opportunity to swap activities with other barges. Moreover, terminals do not take into account the closing times of activities during their scheduling. The latter two issues are interesting extensions of our model. Finally, we find that the introduction of sea vessels and closing times of containers lead to significant sojourn times of barges in the port, which is mainly caused by increased waiting time at terminals.

Based on the experimental results we believe that a communication protocol based on service-time profiles is promising in practice. Through the service-time profiles terminals can reveal to barges when the terminal can be visited, without giving too much insight in the operations of the terminal. We expect that the Multi-Agent based solution using service-time profiles may lead to a significant improvement of barge and terminal operations in the current situation.

12 Further research

This research has raised many questions in need of further investigation. We mention some of these. First, it would be interesting to apply the Multi-Agent system in practice and to see how the system performs. We already aim to do a simulation study based on realistic data of the Port of Rotterdam. Second, it is interesting to introduce disturbances during operation which might lead to a need for, e.g., replanning and robust scheduling. Third, we aim to further develop the algorithms of barge and terminal operators to align the agent intelligence with the way specific barge and terminal operators make decisions. Fourth, we aim to evaluate our model with practitioners in a game-like setting to see how they use information of service-time profiles and if this way of communication is acceptable for the actors involved. Fifth, we aim to investigate the relationship between the best amount of slack in a service-time profile of a terminal and the performance of the whole system in general.

Acknowledgement *This research is performed within a TRANSUMO program. TRANSUMO is a Dutch foundation which supports research to realize a transition to sustainable mobility. We are grateful to TRANSUMO for financing this research.*

References

- Bürckert, H., Fund, P. and Vierke, G. (2000). Holonic transport scheduling with Teletruck, *Applied Artificial Intelligence* **14**(7): 697–725.
- Campbell, A. M. and Savelsbergh, M. (2006). Incentive schemes for attended home deliveries, *Transportation Science* **40**(3): 327–341.
- Christiansen, M., Fagerholt, F. and Ronen, D. (2004). Ship routing and scheduling: Status and perspectives, *Transportation Science* **38**(1): 1–18.
- Christiansen, M. and Fagerholt, K. (2002). Robust ship scheduling with multiple time windows, *Naval Research Logistics* **49**(6): 611–625.
- Connekt (2003). Eindrapport approach., *Technical report*, Connekt, Delft. Report is in Dutch.
- Cordeau, J., Laporte, G., Legato, P. and Moccia, L. (2005). Models and tabu search heuristics for the berth-allocation problem, *Transportation Science* **39**(4): 526–538.
- Davidsson, P., Henesey, L., Ramstedt, L., Törnquist, J. and Wernstedt, F. (2005). An analysis of agent-based approaches to transport logistics, *Transportation Research Part C* **13**(4): 255–271.
- Decker, K. and Li, J. (2000). Coordinating mutually exclusive resources using gpgp, *Autonomous Agents and Multi-Agent Systems* **3**(2): 133–157.
- Demeulemeester, E. and Herroelen, W. (1992). A branch-and-bound procedure for the multiple resource-constrained project scheduling problem, *Management Science* **38**(12): 1803–1818.
- Douma, A., Schutten, J. and Schuur, P. (2008). Waiting profiles: an efficient protocol for enabling distributed planning of container barge rotations along terminals in the Port of Rotterdam, *To appear in Transportation Research Part C*.
- Franz, T., Voss, S. and Rölke, H. (2007). Market-mechanisms for integrated container terminal management, *Proceedings of the 6th international conference on computer and IT applications in the maritime industry*, pp. 234–248.

- Henesey, L., Davidsson, P. and Persson, J. (2006). *Agent based simulation architecture for evaluating operational policies in transshipping containers*, Lecture Notes in Computer Science, pp. 73–85.
- Hopp, W. and Spearman, M. (2000). *Factory Physics*, 3 edn, McGraw Hill, New York, NY.
- Imai, A., Nishimura, E. and Papadimitriou, S. (2001). The dynamic berth allocation problem for a container port, *Transportation research part B* **35**(4): 401–417.
- Kolisch, R. (1996). Serial and parallel resource-constrained project scheduling methods revisited: Theory and computation, *European Journal of Operational Research* **90**: 320–333.
- Kolisch, R. and Drexel, A. (1996). Adaptive search for solving hard project scheduling problems, *Naval Research Logistics* **43**(1): 23–40.
- Kuo, T., Huang, W., Wu, S. and Cheng, P. (2006). A case study of inter-arrival time distributions of containers ships, *Journal of Marine Science and Technology* **14**(3): 155–164.
- Lokuge, P. and Alahakoon, D. (2007). Improving the adaptability in automated vessel scheduling in container ports using intelligent software agents, *European Journal of Operational Research* **177**: 1985–2015.
- Malandraki, C. and Dial, R. (1996). A restricted dynamic programming heuristic algorithm for the time dependent traveling salesman problem, *European Journal of Operational Research* **90**: 45–55.
- Melis, M., Miller, I., Kentrop, M., van Eck, B., Leenaarts, M., Schut, M. and Treur, J. (2003). *Intelligent Logistics Concepts*, Eburon Publishers, chapter Distributed Rotation Planning for Container Barges in the Port of Rotterdam, pp. 101–116.
- Moonen, H., van de Rakt, B., Miller, I., van Nunen, J. and van Hillegersberg, J. (2007). *Managing Supply Chains - Challenges and Opportunities*, number ISBN: 8763001780, Copenhagen Business School Press, Copenhagen, chapter Agent Technology supports Inter-Organizational Planning in the Port.
- Ronen, D. (1983). Cargo ships routing and scheduling: Survey of models and problems, *European Journal of Operational Research* **12**: 119–126.
- Schut, M., Kentrop, M., Leenaarts, M., Melis, M. and Miller, I. (2004). Approach: Decentralised rotation planning for container, *Proceedings of the 16th European Conference on Artificial Intelligence*, Prestigious Applications Intelligent Systems, Valencia.
- Schutten, J. (1996). List scheduling revisited, *Operations Research Letters* **18**: 167–170.
- Schutten, J. (1998). Practical job shop scheduling, *Annals of Operations Research* **83**: 161–178.
- Sinclair, M. and van Dyk, E. (1987). Combined routing and scheduling for the transportation of containerized cargo, *The Journal of the Operational Research Society* **38**(6): 487–498.
- Stahlbock, R. and Voss, S. (2008). Operations research at container terminals: a literature update, *OR Spectrum* **30**: 1–52.
- Steenken, D., Voß, S. and Stahlbock, R. (2004). Container terminal operation and operators research - a classification and literature review, *OR Spectrum* **26**: 3–49.
- Thurston, T. and Hu, H. (2002). Distributed agent architecture for port automation, *Proceedings of the 26th Annual International Computer Software and Applications Conference*.

Appendix A Common experimental settings

The scenarios we consider have common settings, namely the network layout, some parameters and distributions used to generate randomized data, and finally the calculation of the time window of the barge. These settings are similar to the settings presented in Douma et al. (2008).

A.1 Sailing times

In the experimental settings we vary the number of terminals per region. The sailing time between two terminals only depends on the regions each of the terminals belongs to, not on the Euclidian distance. In the port it is not possible to sail straight from one terminal to another, since there are only a few connecting water ways. We therefore assume that sailing time within a region is always equal to 20 minutes. Sailing through a region (without visiting a terminal) takes 40 minutes. Sailing times between terminals are given by Table 8 on a regional level. So, from Table 8 we can see that traveling from a terminal in region A to a terminal in region C takes 240 minutes in a line network configuration.

From/to a terminal in region	<i>Line</i>			<i>Triangular</i>		
	A	B	C	A	B	C
Port entrance and exit point	20	140	260	20	140	140
Region A	20	120	240	20	120	120
Region B	120	20	120	120	20	120
Region C	240	120	20	120	120	20

Table 8: Sailing times (in minutes) between terminals belonging to specific regions, specified for different network layouts

A.2 Parameter settings and distributions

The number of barges that visit the port within the planning horizon is derived from the number of terminals in the network, the capacity available for processing barges, and the average number of terminal visits in a rotation. The interarrival time between barges is exponentially distributed.

Parameter	Value
Time to load or discharge a container	3 min.
Mooring time on arrival and departure	10 min.
Maximum number of terminal visits per rotation	15

Table 9: Parameter settings

The call size (sum of the containers to load and discharge) at a terminal is drawn from a normal distribution. The mean and standard deviation varies per experimental setting. We discretize the normal distribution by rounding to the nearest integer with a minimum value of one. The number of terminal visits (calls) in a rotation is triangularly distributed with a minimum a , maximum b , and mode c . The mode denotes the most frequent value in the distribution. The minimum a is equal to one. The maximum b is equal to the maximum number of calls in a rotation or the number of terminals in the network. Mode c is equal to $(a + b) / 2$. Other parameters are given in Table 9. The distributions used for the call size and the number of terminals in a rotation are chosen in the absence of real data.

A.3 Time window of the barge

Most barges, sailing the river Rhine to Rotterdam, sail according to sailing schedules that are determined once a year. This means that the time a barge is supposed to be in the port is fixed, irrespective of the number of calls in the port, i.e., the time window of all barges have equal length. On the other hand, one might argue that the sojourn times of barges in the port depend also on the number of terminal visits in the rotation. We therefore choose to consider both fixed and variable time windows.

Fixed time windows are determined as follows. We assume an average number of calls and an average call size per call. We assume that an average barge visits all regions in the network. Based on that, we can calculate the expected handling time (including mooring time) and sailing time. This is an estimation of the minimum time an average barge needs in the port to finish all its activities. To add slack for waiting at terminals, we multiply the sum of the handling and sailing time with some slack factor which varies per experimental setting. This factor is chosen such that a reasonable number of barges can leave the port within their time window. The exact value of the factor is not very important in our experiments, more important is the fact that the time windows of all barges have the same length (in one single experiment). The size of the time window does not depend on the chosen average utilization degree.

The *variable time windows* are calculated as follows. For every barge, we calculate the sum of the handling time (including mooring time) and the expected sailing time. The result is increased with some fixed percentage of slack and a variable percentage depending on the number of terminals in the rotation. The slack per terminal (x) and the slack per rotation (y) depend on the utilization of the network and differs per experimental settings we consider. To give an illustration, assume that we have a network with a utilization degree of 50% and corresponding slack factors $x = 3\%$ and $y = 10\%$. The time window of a barge with, e.g., eight terminals is then equal to $(1 + 10\% + 8 \cdot 3\%)$ times the total handling and sailing time in the rotation.