

Hartmann, Sönke

Working Paper — Digitized Version

Generating scenarios for simulation and optimization of container terminal logistics

Manuskripte aus den Instituten für Betriebswirtschaftslehre der Universität Kiel, No. 564

Provided in Cooperation with:

Christian-Albrechts-University of Kiel, Institute of Business Administration

Suggested Citation: Hartmann, Sönke (2002) : Generating scenarios for simulation and optimization of container terminal logistics, Manuskripte aus den Instituten für Betriebswirtschaftslehre der Universität Kiel, No. 564, Universität Kiel, Institut für Betriebswirtschaftslehre, Kiel

This Version is available at:

<https://hdl.handle.net/10419/147631>

Standard-Nutzungsbedingungen:

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

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

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

Terms of use:

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

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

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

Manuskripte
aus den
Instituten für Betriebswirtschaftslehre
der Universität Kiel

No. 564

**Generating Scenarios for Simulation and
Optimization of Container Terminal Logistics**

Sönke Hartmann



Abstract

This paper introduces an approach for generating scenarios of sea port container terminals. A scenario consists of arrivals of deep sea vessels, feeder ships, trains, and trucks together with lists of containers to be loaded and unloaded. Moreover, container attributes such as size, empty, reefer, weight, and destination are included. The generator is based on a large number of parameters that allow the user to produce realistic scenarios of any size. The purpose of this paper is to outline the parameters that are important to produce realistic scenarios of high practical relevance and to propose an algorithm that computes scenarios on the basis of these parameters. The scenarios can be used as input data for simulation models. Furthermore, they can be employed as test data for algorithms to solve optimization problems in container terminal logistics such as berth planning and crane scheduling. The generator discussed here has been developed within the simulation project at the HHLA Container-Terminal Altenwerder in Hamburg, Germany. Nevertheless, its structure is general enough to be applied to various other terminals as well.

Keywords: Container logistics, container terminal, scenario generator, simulation.

1 Introduction

Since the 1960s, the container has gained an enormous importance in worldwide trade and transportation of goods. This is due to both the increasing containerization (which means that the number of goods transported in containers has steadily grown) and increasing world trade. As a consequence, new container terminals are being built and existing ones are extended in order to cope with the growing number of containers. In addition, the container terminals face the challenge of turning around not only more but also larger ships in the shortest possible time. To meet these objectives, container terminals employ innovative (and often automated) equipment and optimize their logistic processes. Given the increasing importance of efficient container terminal logistics, it is no surprise that practitioners as well as researchers continuously develop new optimization approaches as well as simulation models.

Optimization problems include the allocation of berths to arriving vessels (see Imai et al. [11, 12], Lim [18]) as well as scheduling the loading and unloading operations of quay cranes (see Daganzo [5], Peterkofsky and Daganzo [20]). Internal transportation of containers, particularly between the quay and the stack, has been studied for straddle carriers (see Böse et al. [4], Kim and Kim [15], Steenken et al. [22]) and for automated guided vehicles (see Bae and Kim [2], Evers and Koppers [7]). Moreover, the problem of allocating and scheduling stacking cranes has been addressed (see Zhang et al. [26]). Finally, strategies for locating containers in the yard have been discussed for two different cases, namely import containers (see de Castilho and Daganzo [6], Kim and Kim [13]) and export containers (see Kim et al. [14], Taleb-Ibrahimi et al. [23]). An approach covering both cases has been proposed as well (see Zhang et al. [25]). A more comprehensive overview of literature on optimization issues is given by Meersmans and Dekker [19]. While these papers consider sea port container terminals, an optimization approach for an inland container terminal (i.e., without vessels or ships) has been presented by Alicke [1].

Simulation models are developed to evaluate the dynamic processes on container terminals. This allows to generate and analyze statistics such as average productivity and average waiting time (e.g., of a quay crane waiting for a straddle carrier). This way, potential bottlenecks can be identified. Usually, detailed simulation models cover both the physical resources (particularly

the equipment such as cranes and vehicles) and the components for control and strategies (hence, simulation models provide a testing environment for optimization algorithms). Simulation projects can be carried out when building a new terminal (see Schütt and Hartmann [21]) or when analyzing or modifying an existing one (see Legato and Mazza [17], Yun and Choi [24]). In either case, plans can be tested by means of simulation before they are actually implemented. Moreover, simulation models can be employed as a decision support system for the terminal management (see Gambardella et al. [8, 9]). For example, one could simulate the next shift on the basis of decisions to be made (e.g., on the assignment of resources) such that the effects of the decisions can be evaluated in advance.

This paper introduces an approach to generate scenarios for container terminals. On the basis of parameter settings, the generator computes arrivals for deep-sea vessels, feeder ships, trains, and trucks including lists of containers to be unloaded and loaded. By specifying the parameters appropriately, one can generate realistic scenarios for terminals of different sizes. It is possible to adjust the distributions among the modes of transport, the ship sizes, the arrival distributions over time, and the distributions of various container properties. The generator has been developed within the simulation project of the new HHLA Container-Terminal Altenwerder in Hamburg, Germany (for details on that terminal see Baker [3]). Nevertheless, it is general enough to cover other seaport container terminals as well. In fact, the parameters keep the generator as general and flexible as possible such that it is able to generate scenarios for almost all sea port container terminals. Since it was developed for a real-world container terminal, special emphasis was laid on an easy-to-use generator to produce realistic scenarios. The parameters allow to employ real-world data such as planning information and statistics from the past to allow for a realistic scenario. This paper outlines the parameters that have been identified as being important to produce realistic scenarios for container terminal logistics, and it defines a procedure to construct scenarios on the basis of these parameters.

Generating experimental data is an important field in operations research (see, e.g., Hall and Posner [10], Kolisch et al. [16]). The main purpose of the generator presented in this paper is to produce scenarios that can be used as input data for simulation models. Such scenarios are needed if no real world data from the past are available that could be employed. This is the case for new terminals to be built as well as for terminals to be expanded. Moreover, the generator allows to produce scenarios for future developments such as more ship arrivals or larger vessels. In addition to simulation, the scenarios can also be used as test data for various optimization problems arising in container terminal logistics such as berth planning, quai crane scheduling, and container storage.

2 General Concept

In this paper, a scenario for a container terminal is defined as the data concerning all ship, train, and truck arrivals within a specified horizon including the information related to the containers being delivered or picked up. That is, for each day within the horizon, the scenario includes the arrival time of any ship, train, and truck together with the number of containers delivered and picked up. For each of these containers, detailed information on size (20' oder 40'), type (reefer, oversized etc.), and destination are given.

2.1 Modes of Transportation

The generator distinguishes four different types of transport modes which are depicted in Figure 1. On the seaside we have large vessels for world-wide service as well as (typically smaller)

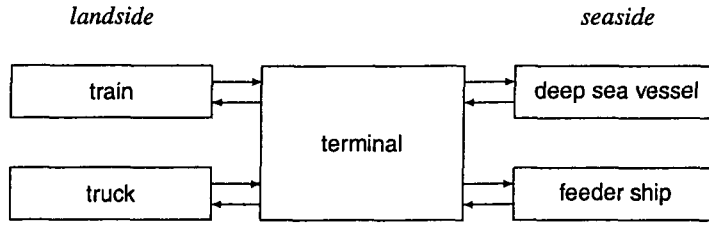


Figure 1: Modes of transport

feeder ships for regional service. The landside is made up by trains and trucks. With these four predefined transport modes, it is possible to adjust realistic parameters of the individual transport modes (usually, a vessel carries substantially more containers than a train). Equally important, the distribution of containers among the four transport modes can be specified (for example, one can determine how many of the containers delivered by truck will be picked up by vessel and by feeder ship). Hence, the generator allows for container flows between any of the four modes. Considering the port of Hamburg (and hence also the Container-Terminal Altenwerder), the container flow between large vessels and feeder ships is of particular importance. Therefore, we distinguish these two categories instead of having just a single seaside mode “ship.” Nevertheless, it is possible to adjust the parameters in a way that only one type of ship is considered.

Note that our definition of transport modes is more general than approaches in the literature (see, e.g., Zhang et al. [26]) which often consider only one seaside and one landside mode and container flows either from landside to seaside (often referred to as “export”) or from seaside to landside (“import”).

2.2 Interdependencies

The interdependencies within a real world container terminal scenario are considerably complex. This includes the distribution of containers among the transport modes on delivery and pick-up as mentioned above. At the same time, the distribution of arrival times and sizes of the means of transport must be considered, as must be the dwell time distribution of containers. In other words, all containers must be assigned to a transport mode for delivery and to a transport mode for pick-up such that the distributions on transport modes, transport mode sizes, transport mode arrival times, and container dwell times are matched simultaneously. With regard to these interdependencies, the goal of the generator concept is twofold: First, it should provide parameters that allow to control the distributions mentioned above. Second, it should provide an algorithm that observes these distributions simultaneously in order to reflect the interdependencies.

2.3 Initial State

When dealing with container terminal logistics on the operational level, one usually has to consider the current state of the terminal. For example, in berth planning, one not only has to consider the next ship arrivals but also those ships that are currently berthed at the quai. If one wants to run a simulation model, one has to consider the current state as well, that is, the means of transport that are currently at the terminal and the containers that are currently stacked in the yard.

At first glance, this seems to imply that generating a scenario would have to include not only the subsequent arrivals but also the initial state of the terminal. However, such an approach bears

two important drawbacks. First, generating an initial state would require additional parameters. Probably, a large number of parameters would have to be specified, and it would be difficult to define parameters that capture a terminal's state in a realistic way. In particular, if we want to obtain a general generator for all seaport terminals, the parameters would have to capture all possible terminal layouts (yard and block configurations) which appears to be rather difficult. Second, the initial state depends on the strategies used in terminal logistics. This becomes clear for simulations of strategies for storing containers in the yard. If the initial state is produced by the generator and not by the strategies tested, it must be seen as a bias in the evaluation of the strategies.

With these points in mind, we chose not to consider an explicit initial state in the generator. Hence, the generator produces only arrivals of means of transport and containers. This is to say that we implicitly assume an empty terminal as initial state. At time zero of a scenario, there are no ships at the quai, no trains or trucks waiting, and no containers in the yard. The generator produces arrivals of means of transport which bring and pick up containers. As we start with an empty terminal, the means of transport on the first few days will only bring containers but they cannot pick up containers. This way, the stack successively fills with containers. At the same time, containers become available for pick-up. After a few days, realistic numbers of containers are picked up by ships, trains, and trucks. Then the stack level oscillates around a constant number of containers because delivery and pick-up numbers are roughly equal (see also the computational example in Section 5.2).

As discussed above, the approach of an empty initial state reduces the number of parameters and avoids any bias when evaluating strategies. When applying a simulation model to such a scenario, one should not use the simulated first days when collecting statistics for evaluation because these days are not yet realistic. In a simulation study on strategies for stacking containers, we used the following rule of thumb: We started collecting statistics after the maximum container dwell time because at this time, the yard state was considered to be realistic. Nevertheless, we had to simulate the first days as well because we needed to fill the stacks with containers (in particular, it was important the stacks fill according to the tested strategies). If for some application a filled yard is not relevant, one might as well omit the first days of the scenario.

2.4 File Concept

The generator works with several files. In order to make the parameter adjustment as easy as possible for the user, all parameters are simply entered into a text file in a predetermined order. This file is read by the generator. After the computation has finished, the generator writes two types of files: First, the scenario is written into text files which can then serve as input for simulation models or optimization algorithms. For each day in the horizon and each of the four transport modes, a separate file is produced in order to avoid huge output files. Second, a file with statistics is generated which contains an overview of the scenario including several statistics. The format of this file is HTML so it can easily be viewed with a browser. The general file concept is visualized in Figure 2.

While the parameter file is described in Section 3, Section 4 summarizes the procedure which is used to compute a scenario. The output files containing the scenario itself and the overview are briefly outlined in Section 5.

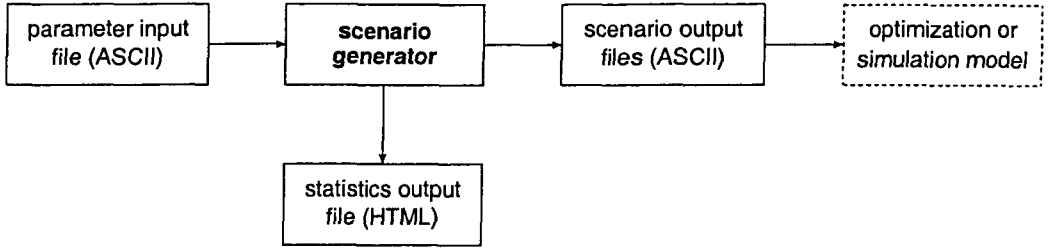


Figure 2: File concept of the scenario generator

T	horizon (number of days)
T_1	first day, $T_1 \in \{\text{Monday}, \dots, \text{Sunday}\}$
C_i	number of containers delivered by mode i within the horizon T
c_{ij}	fraction of containers picked up by mode j among all containers arriving by mode i , $i, j \in \{\text{vessel}, \text{feeder}, \text{train}, \text{truck}\}$
τ	maximum dwell time (days)
δ_t	fraction of containers with a dwell time of t days, $t \in \{0, \dots, \tau\}$

Table 1: General parameters

3 Scenario Parameters

The parameters allow the user to specify the scenario to be generated. They were selected in such a way that planning data and/or statistics can be employed in order to adjust the parameters. This should allow to produce scenarios that are as realistic as possible. A scenario is generated randomly while it is attempted to observe the parameter settings as closely as possible.

The following subsections describe the parameters which are classified into four groups. These are general parameters like the horizon and the number of containers arriving within that horizon, parameters concerning the sizes of the modes of transport, parameters dealing with arrival frequencies, and parameters reflecting container properties like size and weight distribution.

3.1 General Parameters

A summary of the general parameters is given in Table 1. In what follows, they are discussed in more detail. The first of the general parameters are related to the horizon of the scenario to be generated. The length T of the horizon (in days) has to be specified along with the first day T_1 of the horizon which can be a Monday, Tuesday and so on.

Next, we consider the number of containers which arrive at the terminal within the horizon. We define parameters for the numbers C_{vessel} , C_{feeder} , C_{train} , and C_{truck} of containers that arrive at the terminal by deep sea vessel, feeder ship, train, and truck, respectively, during the horizon of T days. Clearly, this parameter set allows to produce scenarios for container terminals of arbitrary size.

The following parameters take care of the distribution of the containers among the four modes

of transport when picked up at the terminal. Let us consider two modes of transport $i, j \in \{\text{vessel, feeder, train, truck}\}$ (for the sake of readability, we refer to the transport modes by symbolic constants). We denote as c_{ij} the fraction of those containers that are picked up by mode j among those that arrive by mode i . In other words, of the C_i containers arriving by mode i , $C_i \cdot c_{ij}$ containers are picked up by mode j . Note that this definition leads to the condition $\sum_j c_{ij} = 1$ which must be fulfilled for each mode i . The number of containers picked up by train, for example, can be computed as

$$C_{\text{vessel}} \cdot c_{\text{vessel,train}} + C_{\text{feeder}} \cdot c_{\text{feeder,train}} + C_{\text{train}} \cdot c_{\text{train,train}} + C_{\text{truck}} \cdot c_{\text{truck,train}}.$$

If none of the containers arriving by train will be picked up by train, one simply has to set $c_{\text{train,train}} = 0$. By specifying all C_i and c_{ij} , the user has completely defined the number of containers and their distribution among the four modes of transport on arrival and pick-up.

The last set of the general parameters determines the dwell time, that is, the number of days a container spends on the terminal before it is picked up. The average dwell time is important because it has a significant impact on the level to which the terminal stack capacity is filled. Moreover, the dwell time of a container (if known in advance) can play a role in strategies to find an appropriate yard location for an arriving container. First, the maximum dwell time τ (in days) has to be specified. Then for each possible dwell time $t = 0, \dots, \tau$ the fraction d_t of containers with that dwell time has to be given. For example, $d_0 = 0.05$ means that 5 % of the containers are picked up on the day of arrival, $d_1 = 0.15$ implies that 15 % of the containers are picked up after one day in the yard and so on.

3.2 Parameters for Means of Transport

The second group of parameters determines the modes of transport. An overview is given in Table 2. Let us consider the deep sea vessels first. The user can define several classes of vessels, that is, different vessel sizes. After specifying the number $n_{\text{vessel}}^{\text{class}}$ of classes, each class $k \in \{1, \dots, n_{\text{vessel}}^{\text{class}}\}$ is described by the following parameters. Each class is associated with a range of the containers that a vessel of this class will unload but not for those to be loaded. Note that a range is given only for containers to be unloaded. It will become clear below that the number of containers to be loaded is determined by the generator without a parameter range. Next, ranges for the length and the number of bays of the vessel class have to be given. Finally, the fraction (i.e., relative occurrence) of vessels of each class has to be specified.

So far, we have described the parameters for deep sea vessels. The parameters for feeder ships are analogous. For trains, the same parameters apply with the exception that length and, of course, number of bays are not considered. Finally, for trucks, different classes are not considered. For the sake of simplicity, we assume that one truck either delivers a single container or picks up a single container (in the real world application this generator was originally designed for, no data on the distribution of trucks carrying different numbers of containers were available).

The last parameter F limits the relation of the number of containers unloaded and loaded by the same vessel, feeder, or train. If the number of containers to be unloaded and loaded are denoted as U and L , respectively, this parameter implies that the condition $\frac{L}{U} \leq F$ must be fulfilled. The idea behind this is that the classes defined above only determine the number of containers to be unloaded. In order to keep the number of containers to be loaded realistic, it can be limited by this parameter. Note that the parameter F is the same for vessels, feeder ships, and trains. Since trucks carry only a single container by definition, it does not apply to trucks.

n_i^{class}	number of classes of mode of transport $i \in \{\text{vessel, feeder, train}\}$
s_{ik}	fraction of class k for mode $i \in \{\text{vessel, feeder, train}\}$
\underline{S}_{ik}	minimum number of delivered containers in class k of mode $i \in \{\text{vessel, feeder, train}\}$
\overline{S}_{ik}	maximum number of delivered containers in class k of mode $i \in \{\text{vessel, feeder, train}\}$
\underline{L}_{ik}	minimum ship length in class k of mode $i \in \{\text{vessel, feeder}\}$
\overline{L}_{ik}	maximum ship length in class k of mode $i \in \{\text{vessel, feeder}\}$
\underline{B}_{ik}	minimum number of bays in class k of mode $i \in \{\text{vessel, feeder}\}$
\overline{B}_{ik}	maximum number of bays in class k of mode $i \in \{\text{vessel, feeder}\}$
F	factor for the maximal relation of containers unloaded and loaded by a vessel, feeder, or train

Table 2: Parameters for means of transport

D_{id}	fraction of mode $i \in \{\text{vessel, feeder, train, truck}\}$ arriving on day $d \in \{\text{Monday}, \dots, \text{Sunday}\}$
n_i^{timeslot}	number of time slots of arrivals of mode $i \in \{\text{vessel, feeder, train, truck}\}$
h_{ik}	fraction of arrivals in timeslot k for mode $i \in \{\text{vessel, feeder, train, truck}\}$
\underline{H}_{ik}	first hour of time slot k for mode $i \in \{\text{vessel, feeder, train, truck}\}$
\overline{H}_{ik}	last hour of time slot k for mode $i \in \{\text{vessel, feeder, train, truck}\}$

Table 3: Parameters for arrival frequencies

3.3 Parameters for Arrival Frequencies

The third group of parameters determines the distribution of arrivals over time (for a summary see Table 3). For each of the four transport modes, the arrival frequencies can be influenced separately. This is necessary because statistics indicate that the four modes clearly show different arrival distributions over time.

First, for each day $d \in \{\text{Monday}, \dots, \text{Sunday}\}$ and each transport mode i , the fraction D_{id} arriving on that day has to be given. Over the seven days of the week, these fractions must sum up to 1, that is, we require $\sum_{d=\text{Monday}}^{\text{Sunday}} D_{id} = 1$ for each mode i . Next, one day can be partitioned into time slots which allow to capture the arrival rates within a day. Note that the arrival characteristics of a transport mode within a day are assumed to be the same on each day. A time slot is given by the start and end hour (ranging between 0 and 24) and by the fraction arriving within that time slot. Hence, a time slot can reflect a single hour or blocks of several hours.

Considering the notation of Table 3, we show how to reflect the arrival behavior of trucks as an example. Due to German law, trucks are not allowed to travel on Sundays which leads to a fraction of $D_{\text{truck}, \text{Sunday}} = 0$ for container terminals in Germany. Within a day, trucks arrive rather in the daytime than during the night, and their arrivals peak at early afternoon. In a rather rough approach, we could define $n_{\text{truck}}^{\text{timeslot}} = 4$ time slots of 6 hours each (e.g., the second time slot would cover the hours between $\underline{H}_{\text{truck}, 2} = 6$ and $\overline{H}_{\text{truck}, 2} = 12$). The fractions of the four time slots could be $h_{\text{truck}, 1} = 0.1$, $h_{\text{truck}, 2} = 0.3$, $h_{\text{truck}, 3} = 0.4$, and $h_{\text{truck}, 4} = 0.2$.

$p^{20'}$	fraction of 20' containers ($1 - p^{20'}$ is the fraction of 40' containers)
p^{empty}	fraction of empty containers
p^{reefer}	fraction of reefer containers
p^{IMO}	fraction of IMO containers, i.e., containers with dangerous goods
n^{IMO}	number of different IMO container types
p_k^{IMO}	fraction of containers of IMO type k among all IMO containers
p^{OOG}	fraction of OOG containers, i.e., oversized containers
n^{OOG}	number of different OOG container types
p_k^{OOG}	fraction of containers of OOG type k among all OOG containers
n^{dest}	number of container destinations
p_{ik}^{dest}	fraction of containers with destination k among all containers picked up by mode $i \in \{\text{vessel, feeder, train, truck}\}$
$n^{\text{weight}, 20'}$	number of weight groups for 20' containers
$p_k^{\text{weight}, 20'}$	fraction of 20' containers with weight group k
$\underline{w}_k^{20'}$	minimum weight of weight group k for 20' containers
$\overline{w}_k^{20'}$	maximum weight of weight group k for 20' containers
$w_{\text{empty}}^{20'}$	weight of empty 20' container
$n^{\text{weight}, 40'}$	number of weight groups for 40' containers
$p_k^{\text{weight}, 40'}$	fraction of 40' containers with weight group k
$\underline{w}_k^{40'}$	minimum weight of weight group k for 40' containers
$\overline{w}_k^{40'}$	maximum weight of weight group k for 40' containers
$w_{\text{empty}}^{40'}$	weight of empty 40' container

Table 4: Parameters for container properties

3.4 Parameters for Container Properties

The properties of a container (such as size, weight, and destination port) are important for vessel stowage as well as for determining a location for the container in the yard (cf. Kim et al. [14]). In what follows, we summarize the parameters that allow to specify the distribution of various container attributes. An survey is given in Table 4.

The first parameter defines the container size. We distinguish 20' and 40' containers. The user has to enter the fraction $p^{20'}$ of 20' containers. Of course, the $1 - p^{20'}$ is the fraction of 40' containers. Next, the fraction p^{empty} of empty containers has to be given.

The following parameters are related to special container types for which specific constraints for yard locations must be observed. Reefer containers require a connection (usually to electricity) to keep their content cool or frozen. These connections are available only in designated areas of the yard. Other containers may contain dangerous goods such as specific chemicals. These containers are often called IMO containers after the International Maritime Association (IMO). Here, complex restrictions have to be considered. For example, a minimum distance between the yard locations of two containers with certain different chemicals must be observed in order to avoid the danger of a chemical reaction. Finally, OOG (out of gauge) containers are oversized containers which require more space than normal 20' or 40' containers. For each of these three categories, the fraction has to be given (for example, p^{OOG} denotes the fraction of OOG containers

among all containers).

The IMO and OOG containers may be further classified into different groups. In both cases, the number of groups has to be specified, along with the fraction of each group. For illustration, consider the following example of $n^{\text{OOG}} = 2$ different OOG container groups. Let us assume that 30 % of the OOG containers are open top containers for which the load exceeds the container height such that no other container may be put on top of them. Moreover, 70 % of the OOG containers are open on one side such that the width of the container is exceeded by the load. These two OOG groups can then be captured by defining the fractions $p_1^{\text{OOG}} = 0.3$ and $p_2^{\text{OOG}} = 0.7$. If, however, separate groups of OOG containers are not needed, one would simply define a single group by setting $n^{\text{OOG}} = 1$ and $p_1^{\text{OOG}} = 1$. Considering the IMO containers, the parameters allow to apply the standard categorization defined by the International Maritime Association.

The next set of parameters specifies the container weight. Starting with 20' containers, the number of weight groups has to be specified. Then for each weight group the fraction of that group together with a lower and an upper weight limit are entered (the latter two can be interpreted as, e.g., metric tons). Subsequently, weight groups for 40' containers have to be defined. Finally, the weights of empty 20' and empty 40' containers are required.

The last set of parameters for container attributes is concerned with the destination of a container. First, the user specifies the number n^{dest} of destinations. Considering container terminals in Hamburg as an example, the destinations would subsume destination ports for large vessels such as Hong Kong and Singapore, destination ports for feeder ships such as Gothenborg, and destinations for trains such as Munich. As the destination of a container may be unknown (especially if a container will be picked up by a truck), it is useful to define an additional destination representing an unknown destination. Obviously, the destination is related to the means of transport that picks up the container. Therefore, for each of the four means of transport, a separate distribution among the destinations can be given. For mode of transport $i \in \{\text{vessel}, \text{feeder}, \text{train}, \text{truck}\}$ and destination $k \in \{1, \dots, n^{\text{dest}}\}$, p_{ik}^{dest} represents the fraction of containers with destination k among all containers picked up by transport mode i . Clearly, this implies $\sum_{k=1}^{n^{\text{dest}}} p_{ik}^{\text{dest}} = 1$ for each mode $i \in \{\text{vessel}, \text{feeder}, \text{train}, \text{truck}\}$. For example, let us assume that destination $k = 3$ refers to an Asian port like Hong Kong. If 10 % of the container picked up by large vessels go to Hong Kong but none of the containers picked up by feeder ships, one will set $p_{\text{vessel},3}^{\text{dest}} = 0.1$ and $p_{\text{feeder},3}^{\text{dest}} = 0$.

Finally, let us remark that for several applications, specifying parameters for container properties is not necessary. If only berth planning or container transportation is under consideration, details on containers are usually not needed. However, if scenarios for problems like location of containers in the yard or for overall simulation models are generated, container properties will typically be relevant. Even if the destination property is relevant, it might not be necessary to model "real" destinations and ports exactly. Often, a number of abstract destinations will be sufficient.

4 Procedure for Generating a Scenario

In this section, we summarize the algorithm that constructs a scenario. The goal of the algorithm is to compute a scenario in a way that the parameter settings are observed as closely as possible. The following three subsections describe the three stages of the algorithm.

4.1 Generating Means of Transport and Containers to be Unloaded

In the first stage of the generator, individual means of transport (i.e., individual vessels, feeder ships, trains, trucks) are computed. In what follows, we describe the computation of deep sea

vessels in more detail. The procedure for the remaining three modes of transport is analogous.

Basically, we proceed as follows. We begin with a distribution of the number of containers arriving by vessel onto the days within the horizon. Then we distribute the containers assigned to a day onto different vessels arriving on that day. The last step is to compute the arrival times of the vessels.

Recall that $D_{\text{vessel},d}$ denotes the fraction of containers arriving on day $d \in \{\text{Monday}, \dots, \text{Sunday}\}$. For each day $t = 1, \dots, T$ of the horizon, we set $D'_{\text{vessel},t} := D_{\text{vessel},d}$ if the t -th day of the horizon corresponds to day $d \in \{\text{Monday}, \dots, \text{Sunday}\}$ with respect to the first day T_1 . Now we are ready to split the containers arriving by vessel during the entire horizon onto the days of the horizon. The number of containers arriving by vessel on day $t = 1, \dots, T$ is defined as

$$C_{\text{vessel},t} = C_{\text{vessel}} \cdot \frac{D'_{\text{vessel},t}}{\sum_{q=1}^T D'_{\text{vessel},q}}.$$

Having determined the number of containers $C_{\text{vessel},t}$ delivered by vessel on each day t , we now split this number into loads of individual vessels. This is done for each day by repeatedly applying the following steps: First, we randomly select a vessel class k according to the fractions defined as parameters. Then we draw the number of containers to be unloaded from the interval $\{\underline{S}_{ik}, \dots, \bar{S}_{ik}\}$. The number of containers assigned to that vessel is then subtracted from the remaining containers on that day. This is repeated until one of the following two cases applies:

- If less containers are left than the number that would be required for the smallest vessel type, then the remaining containers are randomly added to the vessels that have already been defined for that day.
- If less containers are left than the number that would be required for the largest vessel type, then another vessel is constructed which is assigned to all remaining containers. To do so, an appropriate vessel type with respect to minimum and maximum number of containers is selected (ties are broken arbitrarily).

Whenever a new vessel is defined, the length and the number of bays are selected randomly within the respective parameter intervals of the vessels' class.

We now have to determine the exact time at which each of the vessels arrives on the day under consideration. For each vessel, we first draw a time slot k for arrival according to the fractions $h_{\text{vessel},k}$ which are interpreted as probabilities. Then we draw the hour of arrival from $\{\underline{H}_{\text{vessel},k}, \dots, \bar{H}_{\text{vessel},k} - 1\}$. Finally, we draw the minutes past the hour from the interval $\{1, \dots, 59\}$.

For each day, the process described above leads to a number of vessel arrivals and to the number of containers to be unloaded from each vessel. This is successively done for all days of the horizon. Then this procedure is applied to generate individual means of transport for the remaining three modes as well.

In the process described above, each generated vessel, feeder ship, train, and truck is assigned a unique number (ID). Accordingly, each generated container is assigned a unique ID. These IDs connect the means of transport with the individual containers they *bring* to the terminal. In the next stage of the generator, the IDs will be used when defining the connection that has not yet been considered, that is, the connection between the means of transport and the individual containers they *pick up*.

4.2 Generating Container Pick-up

So far, we have generated lists of arriving means of transport together with arrival date and time as well as the lists of containers delivered. While we already know which container is delivered by which vessel, feeder ship, train, or truck, we now determine for each container which vessel, feeder ship, train, or truck will pick it up. For all containers, the following three steps are executed. Let us consider a container for which we want to compute the pick-up information.

In the first step, we determine the transport mode that will pick up the container. The distribution of containers among the transport modes is prescribed by parameter c_{ij} (see Table 1). If the container is delivered by transport mode i , then we use c_{ij} as the probability that it will be picked up by mode j ($i, j \in \{\text{vessel, feeder, train, truck}\}$). Let $j^* \in \{\text{vessel, feeder, train, truck}\}$ denote the transport mode selected to pick up the container.

The second step is to determine the day on which the container will leave the terminal. With t_1 we denote the day on which the container arrives at the terminal. Now we determine the dwell time of the container. Using the dwell time distribution, dwell time $\Delta \in \{0, \dots, \tau\}$ is drawn with probability δ_Δ . Let Δ^* be the selected dwell time. The day on which the container is picked up is then determined as $t_2 = t_1 + \Delta^*$.

Finally, the third step is to find a mean of transport mode j^* that arrives on day t_2 . Among the means of transport that fulfill these requirements, some might not be eligible for pick-up because they have already reached their maximal load. Given a mean of transport with U containers to be unloaded that has already been assigned L to load, it can only receive more containers to pick up if we have $L < F \cdot U$ (for the definition of parameter F see again Table 2). Let M be the set of the eligible means of transport of mode j^* arriving on day t_2 . When we select one of them to pick up the container under consideration, we want the probability of larger means of transport (e.g., larger vessels) to have a higher probability to be selected. To do so, we reflect the size of the means of transport $m \in M$ by the number U_m of containers to be unloaded (recall that the number of containers to be unloaded has already been fixed). More precisely, the probability to chose $\mu \in M$ is defined as

$$p(\mu) = \frac{U_\mu}{\sum_{m \in M} U_m}.$$

Three special cases that may occur in the third step should briefly be mentioned:

- If no eligible means of transport exist on that day (either because none of mode j^* arrive or because those that arrive have already reached their maximal load), then we start again with the first step.
- If day t_2 is not in the horizon (i.e., $t_2 > T$), then a mean of transport is not determined.
- If the transport mode j^* is by truck, we define a new truck to pick up the container given that truck arrival on that day is possible (recall that we assumed trucks to deliver or pick up a single container, thus the selection mechanism of existing means of transport as described above cannot be applied).

Observe that at the end of this stage we know for each vessel, feeder ship, train, and truck which containers it unloads and which containers it loads. Of course, we also know for each container which vessel, feeder ship, train, or truck delivers it and which one picks it up. Hence, we have completed the connection between means of transport and individual containers. Thereby, we have taken into account the parameters concerning the classes (or sizes) of the means of transport, the arrival time distributions over the days and within a day, and the dwell time distribution.

4.3 Generating Container Properties

At this point, we have individual containers, each of which is associated with individual means of transport for arrival and pick-up. The last stage of the generator consists of the assignment of properties to the containers. The procedure is based on the parameters of Section 3.4 and is rather straightforward.

Using the fractions $p^{20'}$, p^{empty} , p^{reefer} , p^{IMO} , and p^{OOG} as probabilities, the related properties are fixed for each container. Next, for the containers that have been assigned the IMO attribute, a specific IMO type is selected. This is done using the fractions of the different IMO types as probabilities. Analogously, OOG containers are assigned an OOG type.

The destination of a container is determined in a similar way. However, the choice of the destination depends on the transport mode that picks up the container. That is, if a container is picked up by mode $i \in \{\text{vessel}, \text{feeder}, \text{train}, \text{truck}\}$, then the probability to select destination $k \in \{1, \dots, n^{\text{dest}}\}$ is given by p_{ik}^{dest} .

Finally, a container is assigned a weight. Here, the weight parameters related to the container size (20' or 40') are applied. If the container has been defined as empty, the related parameter for the weight of empty containers is used. Otherwise, a weight group k is drawn according to the related fractions which are employed as probabilities. The actual weight w is randomly chosen from the interval determined by the minimal and maximal weight of selected weight group k , that is, we select $w \in \{\underline{w}_k^{20'}, \dots, \bar{w}_k^{20'}\}$ or $w \in \{\underline{w}_k^{40'}, \dots, \bar{w}_k^{40'}\}$, respectively.

5 Output of the Generator

So far, we have described the parameter input and the algorithm to compute a scenario. In this section, we turn to the output of the generator. As already pointed out in Section 2.4, two kinds of output files are produced, namely the detailed scenario data itself and a statistics file.

5.1 Scenario Data

The scenario data contains all information that is needed to perform a detailed simulation. The structure of the output files can be summarized as follows. The means of transport are listed in the order of arrival. Each vessel, feeder ship, train, and truck is described by a line that contains the

- transport mode,
- unique ID,
- arrival day and time,
- number of containers to be unloaded,
- number of containers to be loaded.

For vessels and feeder ships, also the length and the number of bays are given. Then a line for each container to be loaded and unloaded follows which contains all container attributes. These attributes include information about the pick-up of the container and its properties, that is,

- unique container ID,
- mode and ID of the vessel, feeder ship, train, or truck that picks up the container,

- arrival day of the vessel, feeder ship, train, or truck that picks up the container,
- size (20' or 40'),
- empty (true or false),
- reefer (true or false),
- IMO (true or false and, if true, IMO type),
- OOG (true or false and, if true, OOG type),
- destination,
- weight.

Various computational tests on the basis of realistic parameter settings have been performed. For example, the generator produces about 6.5 MB of output for a scenario of 21 days and a total of 57,500 containers arriving within the horizon. This corresponds to a medium sized terminal with one million containers per year. On a Pentium III computer with 1 GHz, this computation took approximately 7.1 seconds (we used the lcc compiler under Windows XP). In order to keep the individual files small, a separate file is generated for each transport mode and each day (in this example, this leads to 84 files).

It should be noted that the generator produces scenarios that are complete in the sense that all information is available. In reality, however, one always has to deal with missing or incorrect data. For example, truck arrivals are usually not known in advance (only estimates can be made), and the pick-up information of a container is often unknown at the time of arrival (particularly if a container will not be loaded by a deep sea vessel). For a realistic simulation, the model should include explicit assumptions on which information may be exploited at what time during the simulation.

As already mentioned, the output files are written in the ASCII format. This is advantageous because it allows the user to view and edit the generated scenario data with any editor. Editing can be useful to adjust the scenario. E.g., one might want to modify the arrival times of the vessels. Nevertheless, due to the interdependencies within a scenario, any editing should be done carefully. Otherwise, one might produce inconsistent data such as containers that should be picked up before they have actually arrived (or even containers that are never picked up at all). Inconsistent data can lead to errors when used in optimization or simulation models.

5.2 Statistics Output

The purpose of the statistics file is twofold: First, it illustrates the generated scenario by providing some overview information. Second, it provides a possibility to check the scenario with respect to plausibility.

The statistics file contains various illustrations of the scenario. Perhaps most important, the generated schedules for deep sea vessels, feeder ships, and trains are summarized in tables. These schedules contain arrival date and time together with the numbers of containers that are delivered and picked up. For trucks, no schedules are given because usually there are far too many of them (recall that a truck carries only a single container). Moreover, the average dwell time of containers in the yard is displayed. Finally, some kind of rough simulation concerning the stack is provided. For each day within the horizon, the numbers of containers arriving and being picked up on that

day		morning	arrival	departure	evening
1	Thursday	0	3395	0	3395
2	Friday	3395	4024	825	6594
3	Saturday	6594	1862	1390	7066
4	Sunday	7066	1058	1224	6900
5	Monday	6900	2932	1756	8076
6	Tuesday	8076	3770	2226	9620
7	Wednesday	9620	2353	2781	9192
8	Thursday	9192	3828	2894	10126
9	Friday	10126	3777	3325	10578
10	Saturday	10578	1973	2509	10042
11	Sunday	10042	848	1274	9616
12	Monday	9616	2671	2651	9636
13	Tuesday	9636	2931	2935	9632
14	Wednesday	9632	4200	3002	10830
15	Thursday	10830	3225	3478	10577
16	Friday	10577	3491	3638	10430
17	Saturday	10430	2014	2141	10303
18	Sunday	10303	920	1383	9840
19	Monday	9840	3174	2821	10193
20	Tuesday	10193	3022	3285	9930
21	Wednesday	9930	2032	2457	9505

Table 5: Development of the number of containers in the stack (part of example statistics file)

day are given, together with the resulting number of containers in the stack at the end of the day. The computation is based on the simplifying assumption that all containers are put in the stack on the day the related mode of transport arrives at the terminal (a more accurate computation could only be achieved by a “real” simulation model). An example for the stack illustration over three weeks is displayed in Table 5. Observe that the scenario starts with an empty stack which is filled successively until the number of containers in the stack oscillates around a constant level (see also Section 2.3).

The plausibility check of a scenario is provided by listing most of the parameters together with related statistical information from the generated data. For example, the selected parameter for the fraction of 20’ containers is given along with the actual fraction. For the more complex parameters which describe distributions over time, also the correlation between the selected distribution and the actual one is provided. In our computational tests, we observed that most correlations were very close to 1.

6 Conclusions

In this paper, we presented an approach to generate scenarios for container terminals. A scenario contains data on arrivals of vessels, feeder ships, trains, and trucks together with lists of the containers to be delivered and picked up and the container attributes (size, weight etc.). The generation of a scenario is controlled by means of various parameters. The goal was to develop an easy-to-use generator that produces realistic scenarios. Therefore, we selected parameters that allowed to use statistics and planning information that are typically available in practice. By employing real-world information for the parameters, the artificially generated scenarios become realistic.

The generator is currently applied at the HHLA Container-Terminal Altenwerder within a simulation study of strategies for yard block and stack selection for arriving containers. As that terminal was still in the planning phase, no appropriate real-world scenarios were available. Therefore, the generator was needed to produce scenarios. The parameters were adjusted such that they capture the specific planning assumptions of the Container-Terminal Altenwerder (e.g., the expected number of containers turned over per year). Further parameter settings (e.g., on the arrival distributions of the transport modes over time) were done on the basis of statistics from the HHLA Container-Terminal Burchardkai, another container terminal in the port of Hamburg. The simulation study has already helped to improve the concept on stack selection strategies. The improved strategies make use of the block capacities more efficiently. It was extremely important to conduct this study in the planning phase of the terminal. Otherwise, the shortcomings of the original concept would probably have caused problems during operation. It should be emphasized that the generator is not restricted to the characteristics of the Container-Terminal Altenwerder. The parameters are general enough to be adjusted to other sea port container terminals as well.

The parameters cover the entire scenario as would be needed for an overall simulation model. However, we have pointed out that the generator can also be used to produce test data for optimization problems that occur in container terminal logistics such as berth planning and quai crane scheduling. Of course, some of the parameters are not important when dealing with a specific subproblem. For example, the properties of the individual containers (such as size, weight, and destination) are not relevant if one wants to generate a scenario for berth planning. In such a case, the related parameters simply need not be filled in. Thus, the proposed generator is a flexible tool that can be applied for various simulation and optimization purposes.

Although the generator is already considerably general, extensions of the generator might be promising tasks in future research. A possible extension of the generator could be to compute the positions of the individual containers on the vessels and feeder ships. These positions are relevant when stowage plans are part of the problem considered. This is of particular importance if the process of loading vessels is examined in detail. Often, each position on the vessel is associated with a specific container. This imposes a partial order on the containers to be loaded. Alternatively, clusters of cells may be associated with container attributes rather than individual containers. In the latter case, there are more degrees of freedom concerning the loading order, and specific strategies are needed to exploit this setting. Hence, the design of an extended generator (and particularly that of the additional parameters) would depend on the application of the stowage plans.

Acknowledgements. This research has been carried out for and sponsored by the HHLA Container-Terminal Altenwerder in Hamburg, Germany.

References

- [1] K. Alicke. Modeling and optimization of the intermodel terminal Mega Hub. *OR Spectrum*, 24:1–17, 2002.
- [2] J. W. Bae and K. H. Kim. A pooled dispatching strategy for automated guided vehicles in port container terminals. *International Journal of Management Science*, 6:47–70, 2000.
- [3] C. Baker. Altenwerder – the details. *Port Development International*, (07/08):24–25, 1999.
- [4] J. Böse, T. Reinert, D. Steenken, and S. Voß. Vehicle dispatching at seaport container terminals using evolutionary algorithms. In R. H. Sprague, editor, *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences*, pages 377–388. IEEE, Piscataway, 2000.

- [5] C. F. Daganzo. The crane scheduling problem. *Transportation Research B*, 23:159–175, 1989.
- [6] B. de Castilho and C. F. Daganzo. Handling strategies for import containers at marine terminals. *Transportation Research B*, 27:151–166, 1993.
- [7] J. J. M. Evers and S. A. J. Koppers. Automated guided vehicle traffic control at a container terminal. *Transportation Research A*, 30:21–34, 1996.
- [8] L. M. Gambardella, G. Bontempi, E. Taillard, D. Romanengo, G. Raso, and P. Piermari. Simulation and forecasting of an intermodal container terminal. In A. G. Bruzzone and E. J. H. Kerckhoffs, editors, *Simulation in Industry – Proceedings of the 8th European Simulation Symposium*, pages 626–630. SCS, Ghent, Belgium, 1996.
- [9] L. M. Gambardella, A. E. Rizzoli, and M. Zaffalon. Simulation and planning of an intermodal container terminal. *Simulation*, 71:107–116, 1998.
- [10] N. G. Hall and M. E. Posner. Generating experimental data for computational testing with machine scheduling applications. *Operations Research*, 49:854–865, 2001.
- [11] A. Imai, K. Nagaiwa, and C. W. Tat. Efficient planning of berth allocation for container terminals in Asia. *Journal of Advanced Transportation*, 31:75–94, 1997.
- [12] A. Imai, E. Nishimura, and S. Papadimitriou. The dynamic berth allocation problem for a container port. *Transportation Research B*, 35:401–417, 2001.
- [13] K. H. Kim and H. B. Kim. Segregating space allocation models for container inventories in port container terminals. *International Journal of Production Economics*, 59:415–423, 1999.
- [14] K. H. Kim, Y. M. Park, and K.-R. Ryu. Deriving decision rules to locate export containers in container yards. *European Journal of Operational Research*, 124:89–101, 2000.
- [15] K. Y. Kim and K. H. Kim. A routing algorithm for a single straddle carrier to load export containers onto a container ship. *International Journal of Production Economics*, 59:425–433, 1999.
- [16] R. Kolisch, A. Sprecher, and A. Drexl. Characterization and generation of a general class of resource-constrained project scheduling problems. *Management Science*, 41:1693–1703, 1995.
- [17] P. Legato and R. M. Mazza. Berth planning and resources optimisation at a container terminal via discrete event simulation. *European Journal of Operational Research*, 133:537–547, 2001.
- [18] A. Lim. The berth planning problem. *Operations Research Letters*, 22:105–110, 1998.
- [19] P. J. M. Meersmans and R. Dekker. Operations research supports container handling. Technical Report EI 2001-22, Econometric Institute, Erasmus University Rotterdam, 2001.
- [20] R. I. Peterkofsky and C. F. Daganzo. A branch and bound solution method for the crane scheduling problem. *Transportation Research B*, 24:159–172, 1990.
- [21] H. Schütt and S. Hartmann. Simulation in Planung, Realisierung und Betrieb am Beispiel des Container-Terminals Altenwerder. In D. P. F. Möller, editor, *Frontiers in Simulation — Simulationstechnik, 14. Symposium in Hamburg*, pages 425–430. SCS, Ghent, Belgium, 2000.
- [22] D. Steenken, A. Henning, S. Freigang, and S. Voß. Routing of straddle carriers at a container terminal with the special aspect of internal moves. *OR Spectrum*, 15:167–172, 1993.
- [23] M. Taleb-Ibrahimi, B. de Castilho, and C. F. Daganzo. Storage space vs. handling work in container terminals. *Transportation Research B*, 27:13–32, 1993.
- [24] W. Y. Yun and Y. S. Choi. A simulation model for container-terminal operation analysis using an object-oriented approach. *International Journal of Production Economics*, 59:221–230, 1999.
- [25] C. Zhang, J. Liu, Y.-W. Wan, K. G. Murty, and R. J. Linn. Storage space allocation in container terminals. Technical report, Hong Kong University of Science and Technology, 2001.
- [26] C. Zhang, Y.-W. Wan, J. Liu, and R. J. Linn. Dynamic crane deployment in container storage yards. *Transportation Research B*, 36:537–555, 2002.