Jena Research Papers in Business and Economics

Determining fixed Crane Areas in Rail-Rail Transshipment Yards

Nils Boysen, Malte Fliedner, Michael Kellner

07/2009

Jenaer Schriften zur Wirtschaftswissenschaft

Working and Discussion Paper Series
School of Economics and Business Administration
Friedrich-Schiller-University Jena

ISSN 1864-3108

Publisher:

Wirtschaftswissenschaftliche Fakultät
Friedrich-Schiller-Universität Jena
Carl-Zeiß-Str. 3, D-07743 Jena

www.jbe.uni-jena.de

Editor:

Prof. Dr. Hans-Walter Lorenz
h.w.lorenz@wiwi.uni-jena.de

Prof. Dr. Armin Scholl
armin.scholl@wiwi.uni-jena.de

www.jbe.uni-jena.de

Determining fixed Crane Areas in Rail-Rail Transshipment Yards

Nils Boysen, Malte Fliedner, Michael Kellner

Friedrich-Schiller-Universität Jena, Lehrstuhl für Operations Management, Carl-Zeiß-Straße 3, D-07743 Jena, Germany,
{nils.boysen,malte.fliedner,michael.kellner}@uni-jena.de

Abstract
Rail-rail transshipment yards are an emerging and more efficient alternative to traditional shunting yards. Instead of a time-consuming reshuffling of railcars via shunting hills, in these modern yards, huge gantry cranes tranship containers between different freight trains. As multiple cranes process the trains in parallel, it is an important operational task to avoid interferences among cranes. To ensure smooth crane operations, it is a widespread real-world policy to assign each crane to an exclusive yard area. In such a setting, each crane's workload heavily depends on the length and location of its respective yard area. The paper on hand provides an exact Dynamic Programming procedure with polynomial runtime, which determines crane areas, so that the resulting workload is evenly spread among cranes and, thus, train processing is accelerated. Furthermore, in a straightforward simulation of transshipment yard operations, the effect of optimal crane areas versus equally sized areas is studied, the latter being a common real-world policy. The results indicate a remarkable speed-up of train processing if optimal crane areas are applied.

Keywords: Railway systems; Transshipment yards; Container handling; Crane scheduling

1 Introduction

In spite of extraordinary support programs introduced by the European Union (EU) and other national authorities, railway systems still face considerable disadvantages when compared to freight traffic by truck. Within the last 25 years the fraction of the overall freight traffic moved by train fell from 20% (1970) to 10% (2005) (EU, 2007). In addition to reduced flexibility and reliability, e.g., caused by the general right of way of passenger traffic in many European countries, a major handicap of train traffic are the high fixed cost. These cost make container transport only profitable if large trains are moved over long distances. However, with traditional point-to-point traffic such high demands for freight transport merely exist between large cities, so that areas of lower population density are excluded from freight transport by rail. Only recently, Hub-and-Spoke systems (as they are applied in road and air traffic since many years) have been identified as a possibility to profitably employ smaller freight trains (see Ballis and Golias, 2004). With such a system, the freight of multiple smaller trains can be consolidated to few large trains in a hub terminal, which are then moved on long-haul routes to realize economies of transportation.

Traditional shunting (or classification) yards have a notable share in the competitive disadvantage of rail transport, since a reshuffling of railcars via shunting hills is very time-consuming. Instead, modern rail-rail transshipment yards are required, where huge gantry cranes, which span over all railway tracks, transship containers between different freight trains more efficiently. Some of these hub yards have already been constructed in the EU (e.g., Port-Bou at the border between France and Spain, see Martinez et al., 2004) and others are under construction (e.g., the German "Mega Hub" in Hannover-Lehrte, see Alicke, 2002; Rotter, 2004). Figure 1 gives a schematic representation of such a rail-rail transshipment yard.

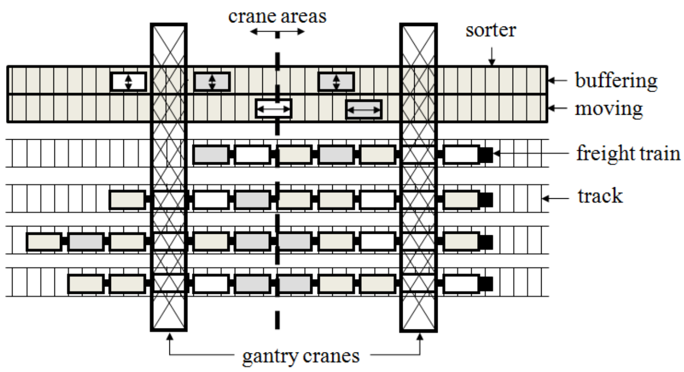

Figure 1: Schematic representation of a rail-rail transshipment yard

A modern rail-rail transshipment yard consists of multiple parallel railway tracks, where successive pulses (one train per track) of freight trains are processed. Container transshipment among trains is conducted by multiple rail-borne gantry cranes, which span over the railway tracks. Due to their size and the fact that they run on the same tracks for horizontal movement, cranes cannot pass by one another, so that interferences occur, whenever a crane A seeks to transport a container to a distant location which is currently blocked by another crane B operating in between. In practice, such interferences are often ruled out by applying a simple organizational policy (see Boysen and Fliedner, 2009): the total yard space is partitioned into distinct areas, to each of which a single crane is assigned. As cranes are not allowed to overstep the left and right-hand borders of their areas, they can never interfere with each other. To enable distant container moves across multiple crane areas, such yards further make us of a sorting system, where automated guided vehicles (Bostel and Dejax, 1998) or some rail-mounted vehicles (Alicke, 2002) take up the container from the emitting area and move it alongside the yard to its dedicated area. There, it is picked up by the respective crane and finally loaded on the train. The sorting system typically comprises one or more moving lanes and buffer lanes (see Alicke, 2002), so that containers can overtake each other in the sorting system.

An important decision problem in this context is to dimension the operating area of each crane in such a way that the workload is evenly spread among cranes. As all cranes process the same pulse of trains in parallel, cranes which executed all moves in their area have to wait idle until the last container is processed. A yard partition which minimizes the maximum workload of cranes also minimizes the total makespan of train processing. In a rail-rail transshipment yard, the actual partition of areas directly effects the total workload required to process all containers. Whenever the start and the target position of a container lie in the same crane area, the container can directly be transshipped by the respective crane (direct move). If, however, start and target positions fall into different areas, two cranes (and the sorter) are required to process the move. Such a double handling of containers, referred to as split move (see Boysen and Pesch, 2008), considerably extends train processing as the time-consuming pick and drop operations need to be executed twice for each split move. Especially, locating the spreader is precision work so that on average merely 20-25 moves per crane and hour can be processed (see Rotter, 2004). Thus, split moves and their impact on the cranes' workload are to be considered while determining crane areas.

The paper on hand presents a Dynamic Programming procedure for the aforementioned problem with polynomial runtime. Furthermore, we compare optimally sized yard areas with equally sized ones, the latter being a widespread policy in real-world transshipment yards (see Boysen and Fliedner, 2009). The results of a straightforward simulation of transshipment yard operations reveal a tremendous potential for processing time reduction by applying optimal crane areas.

The remainder of the paper is structured as follows. Section 2 provides a literature review, whereas Section 3 gives a detailed description of the yard partition problem. An exact Dynamic Programming procedure is presented in Section 4. Then, optimally and equally sized crane areas are compared in a yard simulation (Section 5). Finally, Section 6 concludes the paper.

2 Literature Review

Although there is a lot of attention paid to railway optimization (see, e.g., Cordeau et al., 1998) and intermodal transportation (see Crainic and Kim, 2007) in general, literature on rail-rail transshipment yards is scarce. This is astounding, as these yards are an emerging technology in railway systems (see the surveys by Bontekoning et al., 2004 as well as Macharis and Bontekoning, 2004).

In-depth descriptions of structural properties and different operational policies employed in transshipment yards are provided by Ballis and Golias (2002) as well as Rotter

(2004). Meyer (1998), Abacoumkin and Ballis (2004), Ballis and Golias (2004) as well as Wiegmans et al. (2006) specifically address the design process of an optimal terminal layout. However, only very few research papers deal with the scheduling problems perpetually arising during the daily operations of a transshipment yards. As the overall scheduling task seems far too complicated to allow a simultaneous solution, a hierarchical decomposition of the overall problem is recommendable (see Boysen and Pesch, 2008):

(i) Schedule the service slots of trains by assigning them to pulses.

(ii) Decide on the containers' positions on trains.

(iii) Assign each train to a railway track.

(iv) Assign container moves to cranes.

(v) Decide on the sequence of container moves per crane.

Problem (i) is treated by Boysen and Pesch (2008). Here, a given set of trains to be processed is to be assigned to different pulses. In such a setting, revisits of trains in a later period to receive remaining containers not delivered up to the train's first stay in the yard and double handling of containers are to be avoided. Bostel and Dejax (1998) as well as Corry and Kozan (2006, 2008) treat problem (ii) and provide scheduling procedures to determine the optimal positions of containers on freight trains so that crane moves are minimized. Additionally, multiple restriction, e.g., dangerous goods separations, maximum gross wagon mass and train height, need to be considered while determining container positions. Problem (iii), the assignment of trains to tracks, can be solved as a quadratic assignment problem, which is shown by Alicke and Arnold (1998), if the schedule of trains (problem (i)) and container positions (problem (ii)) are already determined.

Container moves to cranes (problem (iv)) can either be assigned in a *static* or *dynamic* way. Under a static assignment, borders of crane areas are fixes and a crane exclusively processes containers within its area. A dynamic assignment policy allows cranes to move freely alongside the yard, however, crane movements need to be coordinated in real-time, so that interferences are minimized. Obviously, a dynamic assignment leaves more degrees of freedom since cranes can support each other and, thus, promises a more efficient train processing. On the other hand, an information system containing complex scheduling procedures is required. Such a centralized online control of cranes becomes superfluous with exclusive areas. Here, container moves inside each area can simply be scheduled in a decentralized manner by the crane operators, who can flexibly adjust operations to unforseen events (delayed truck or train arrivals, prolonged container moves etc.). Thus, a static assignment of crane areas is a widespread policy in real-world transshipment yards (see Boysen and Fliedner, 2009).

Existing literature mainly investigates a dynamic crane assignment. Alicke (2002) provides a scheduling procedure to jointly cover problems (iv) and (v). Based on constraint programming Alicke dynamically assigns container moves to cranes and decides on the sequence of moves per crane. Related problems also occur in the hinterland of seaport container terminals (see, e.g., Ng, 2005; Zhu and Lim, 2006; Moccia et al., 2006; Lim et al., 2007; Sammarra et al., 2007). Up to now, a static assignment with fixed crane areas is only considered by Boysen and Fliedner (2009) for a conventional rail-truck transshipment yard. However, in such a yard container moves across areas do not occur, so that split moves do not need to be considered. Thus, the paper on hand is the first to exclusively treats problem (iv) with fixed crane areas in rail-rail transshipment yards. Note that our solution procedure (for the static case) can also be helpful for a dynamic container assignment, e.g., to determine a first feasible start solution.

3 Detailed Problem Description

A rail-rail transshipment yard is typically operated in distinct so-called pulses (Bostel and Dejax, 1998) or bundles (Alicke, 2002; Rotter, 2004) of trains, which means that T trains (one per track) are simultaneously served and jointly leave the system not before all container moves are processed, which are required for the respective bundle of trains. Container moves are processed in parallel by the given number n of gantry cranes available on the yard. Typically, at the point in time a train arrives it is already specified which containers to move (problem (ii) in Section 2), so that some containers are declared to be transhipped between given railcars, whereas others – already dedicated to the train's next destination – remain untouched. Thus, we presuppose that a pulse of trains, for which all container moves are specified, is already located on the tracks of a yard. Note that this train set might be an actual pulse of trains already waiting or being expected. In this case, crane areas are newly determined for any pulse of trains, which can easily be executed in a real-world terminal by loading only container moves of the respective area in the actual schedule list on a crane operator's computer monitor inside the steeple cap. However, a pulse can also be a representative average train set, if crane areas are to be fixed over a mid-term horizon.

To ease orientation for crane operators a transshipment yard is subdivided into smaller line segments labeled "slots", whose numbers and borders are drawn on the ground along the horizontal spread of the yard. These slots are adjusted to the length of standardized railcars. For instance in German transshipment yards, the typical slot length is 14 meters to exactly cover a railcar labeled "Lgs580" which carries one forty feet (FEU) or two twenty feet containers (TEU). A typical yard length is 700 meters, so that such a representative transshipment yard would be subdivided into $S = 50$ slots. We presuppose that freight trains are parked and adjusted so that each container exactly falls into one slot.

With these input parameters, it is the aim of the yard partition problem to divide the slots $s = 1, \ldots, S$ of a transshipment yard into n disjunct and consecutive crane areas, so that the maximum workload over all cranes $i = 1, \ldots, n$ induced by respective container moves falling into each crane's area is minimized. Cranes process a given pulse of trains in parallel, so that all cranes have to wait idle until the last crane finished container processing. Thus, the min-max objective balances the workload among cranes and minimizes the makespan of train processing.

In order to estimate the resulting workload of a specific crane area, split moves need to be considered. If start and target position of a container move fall into the same area, the move can directly be processed by the respective crane, so that a single pick and drop operation is required. Such a *direct move* is depicted on the left-hand-side of Figure 2. However, if start and target positions are separated by at least one area border (represented by the dashed vertical line) a *split move* occurs. The crane which covers the start position has to move the container onto some vehicle waiting in the same slot on the moving lane of the sorter. Then, the container is moved to the slot of its target position, where the second crane (covering the target position) finally moves the container from the sorter to the container's final destination on train. The right-hand-side of Figure 2 depicts the double handling required for such a split move.

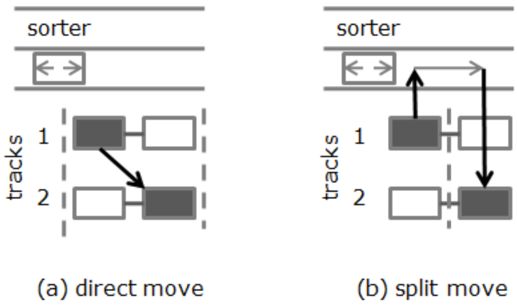

Figure 2: Direct move vs. split move

For each container move $c \in C$ startup slot s_c^s and track t_c^s as well as target slot s_c^t and track t_c^s are known. Thus, for given left l and right r borders of an area, the sets of associated direct moves $D(l,r)$, split moves into the sorter $IN(l,r)$ and and split moves out of the sorter $OUT(l,r)$ can be determined as follows:

$$\begin{array}{l} D(l,r) = \left\{c \in C | l \leq s_c^s \leq r \wedge l \leq s_c^t \leq r\right\} \\ IN(l,r) = \left\{c \in C | l \leq s_c^s \leq r \wedge \left(l > s_c^t \vee r < s_c^t\right)\right\} \\ OUT(l,r) = \left\{c \in C | \left(l > s_c^s \vee r < s_c^s\right) \wedge l \leq s_c^t \leq r\right\} \end{array} \tag{1}$$

A container c can be transshipped by a direct move ($c \in D(l,r)$) only if both start s_c^s and target s_c^t slot are within an area's left l and right r border, whereas a split move into ($c \in IN(l,r)$) and out of ($c \in OUT(l,r)$) the sorter occurs if only one position is located in the respective area. As the actual route of a container through the yard depends on whether it is transshipped by a direct or a split move, we need to consider container specific weights to represent the resulting workload for cranes. If container move c is a direct move, the resulting workload for the respective crane is represented by weight w_c^D, whereas split moves into and out of the sorter are weighted with w_c^{IN} and w_c^{OUT}, respectively. There exist multiple alternatives of how to calculate these weights. In the most basic version each weight can be fixed to one, so that merely the number of moves per crane are leveled. However, more sophisticated weights incorporate the resulting distances of container moves (see the following Example) or approximate exact processing times (see Section 5).

With these weights on hand the workload $W(l,r)$ of a crane area ranging from slot l to r can be determined by summing weights of associated direct and split moves as follows:

$$W(l,r) = \sum_{c \in D(l,r)} w_c^D + \sum_{c \in IN(l,r)} w_c^{IN} + \sum_{c \in OUT(l,r)} w_c^{OUT} \tag{2}$$

Example: In the example of Figure 3 a transshipment yard is subdivided into six slots. On four parallel tracks a pulse of trains is parked, where arcs symbolize container moves $c \in C$ to be processed between the respective railcars. White boxes represent containers which remain on the train, e.g., because they are dedicated to another destination. Arc weights represent the weights w_c^D of direct moves. As gantry cranes are able to simultaneously move in vertical and horizontal direction using two different engines, these weights approximating the distances of container moves are simply chosen as the maximum of the number of tracks or slots covered. Note that this simple approximation presupposes equal vertical and horizontal distances between tracks and slots which are covered with identical crane velocity. Thus, the move from the fourth track of slot five to the second track of slot four passes two tracks and one slot, so that $w_c^D = \max\{2;1\} = 2$. Weights for split moves into and out of the sorter are not depicted, but simply amount to the track number of the respective container position. If the aforementioned container move is executed as a split move (which occurs if an area border is located between slots four and five), then the split move into (out of) the sorter has to cover a distance of four (two) tracks, so that $w_c^{IN} = 4$ ($w_c^{OUT} = 2$).

On the right-hand-side of Figure 3 two alternative solutions for $n = 3$ cranes are depicted, where the numbers indicate the realized weights of the respective crane area. Solution A has equally sized crane areas, which results in a maximum workload of ten faced by crane three. In Solution B border areas lie after slots three and five and the maximum workload reduces to four.

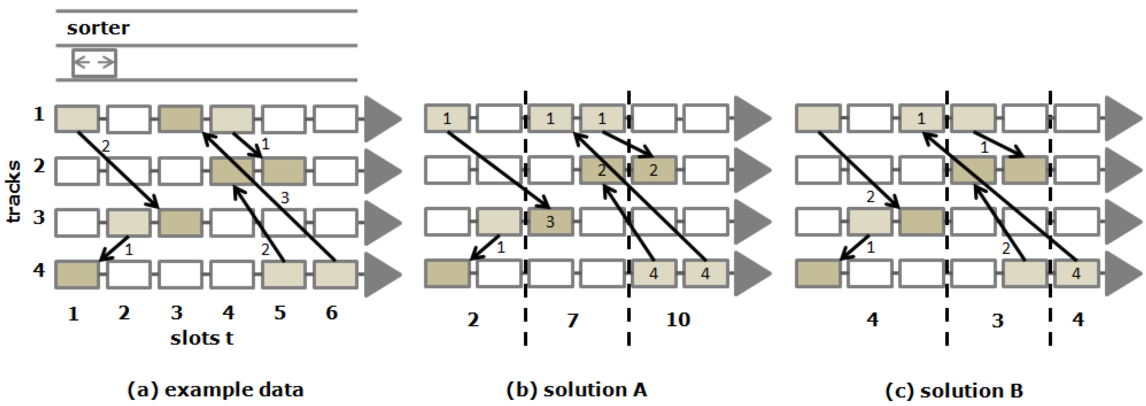

Figure 3: Example data and two solutions

Note that the aforementioned weights for calculating each crane's workload only cover "loaded" moves where cranes actually carry a container, whereas "empty" moves bridging

n	number of gantry cranes (index i)
T	number of tracks (index t)
S	number of slots (index s)
C	set of container moves (index c)
s_c^s, s_c^t	start and target slot of container move c, respectively
t_c^s, t_c^t	start and target track of container move c, respectively
$D(l,r)$	set of direct moves in a crane area ranging from slot l to r
$IN(l,r)$	set of split moves into sorter in a crane area ranging from slot l to r
$OUT(l,r)$	set of split moves out of sorter in a crane area ranging from slot l to r
w_c^D, w_c^{IN}, w_c^{OUT}	weights of move c if executed as a direct move, split move into sorter and out of sorter, respectively
$W(l,r)$	workload in an area ranging from slot l to r
X	solution vector of area borders

Table 1: Notation

the distance between two loaded moves are not considered. However, determining each crane's actual workload (resulting from a complete crane tour with loaded and empty moves) would require integrating a detailed crane scheduling. This results in a complex optimization problem, since already scheduling a single crane takes the form of an asymmetric traveling salesman problem, which is known to be NP-hard in the strong sense (see Garey and Johnson, 1979). With multiple cranes connected via split moves and the sorting system, the problem becomes even more difficult to handle. To avoid such a detailed crane scheduling within the yard partition problem only loaded moves are considered. We, thus, assume that minimizing the workload resulting from loaded moves is strongly positively correlated to minimizing the overall operating time, which includes empty moves. If this assumption holds, minimizing the workload due to loaded moves would be a suitable surrogate objective, which considerably eases the solution process. It is part of our simulation study in Section 5 to evaluate this assumption.

With the help of the notation summarized in Table 1 the yard partition problem for rail-rail transshipment yards (YPP^{rr}) can be formalized as follows.

Model formulation: Within YPP^{rr} a vector $X = \{0, x_1, x_2, \ldots, x_{n-1}, T\} \longrightarrow \{1, 2, \ldots, T-1\}$ of area borders is to be determined which minimizes objective function (3) subject to constraints (4):

$$(\text{YPP}^{rr}) \text{ minimize } F(X) = \max_{i=1}^{n} \{W(x_{i-1}+1, x_i)\} \tag{3}$$

subject to

$$x_i \geq x_{i-1} + 1 \qquad \forall\, i = 1, \ldots, n \tag{4}$$

In objective function (3) the maximum workload over all cranes $i = 1, \ldots, n$ is minimized, where each crane i's workload $W(x_{i-1}+1, x_i)$ is determined by adding all weights of direct and split moves, which fall into the respective area between left border $(x_{i-1}+1)$ up to the right border (x_i). Constraints (4) ensure a minimum area length of one slot between two consecutive trains. Note that by fixing the first and the last entry of solution vector X it is ensured that the area of the first (last) crane starts (ends) in the first (last) slot.

4 Dynamic Programming

As an area's workload exclusively depends on its left and right border (and not on the detailed partition of preceding or succeeding areas) all possible partitions of the yard can be evaluated with the help of a Dynamic Programming (DP) procedure. For this purpose, the decision process is subdivided into n stages, where each stage $i = 1, \ldots, n$ represents a crane. Any stage i contains a set of states (i, s), where each state represents a possible right-hand border (slot s) of the respective crane area i. As the minimum area width is a single slot, each stage $i = 1, \ldots, n-1$ exactly contains $S-n+1$ states. Only, final stage n contains merely a single state (n, S), because in any case last crane n has to cover the remaining yard length up to final slot S.

The DP procedure operates with a forward recursion, so that first stage 1 (representing crane 1) has to be initialized by defining partial objective values f_{1s} for each possible right-hand slot s (see (1) and (2)):

$$f_{1s} = W(1, s) \quad \forall s = 1, \ldots, S-n+1 \tag{5}$$

Then, (partial) objective values f_{is} assigned with the states of remaining stages can be calculated with the basic recursion as follows:

$$f_{is} = \min_{i-1 \leq j < s-1} \{\max\{f_{i-1j};\, W(j+1, s)\}\} \qquad \begin{array}{l} \forall i = 2, \ldots, n-1,\; s = i, \ldots, i+S-n \\ \text{and } \; i = n,\; s = S \end{array} \tag{6}$$

Partial objective value f_{is} of state (i, s) is calculated by considering all feasible predecessor states $(i-1, j)$ of preceding stage $i-1$, which are all those with a smaller right-hand slot: $j = i-1, \ldots, s-1$. For any of these possible predecessor states, workload amounts to the maximum of partial objective value f_{i-1j} of predecessor state $(i-1, j)$ (first term of maximum function) and additional workload $W(j+1, s)$ of the actual yard area of crane i ranging from slot $j+1$ to s (second term of maximum function). Lastly, final state (n, S) is reached and f_{nS} is the maximum workload of the optimal yard partition. Area borders for the optimal partition can be determined by a simple backward recursion along the states part of the optimal policy.

Example (cont.): For the input data of Figure 3 the resulting DP graph is shown in Figure 4. The bold sketched optimal solution value for three cranes amounts to a min-max workload of four. The solution equals Solution B of Figure 3, so that optimal yard areas range up to slots 3, 5 and 6, respectively.

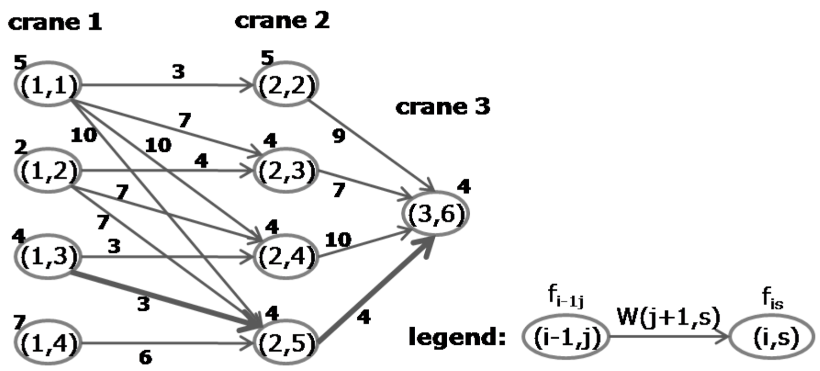

Figure 4: DP graph of the example

The DP graph consists of n stages, where each stage $i = 1, \dots, n$ contains no more than S states. Since only states of consecutive stages $i-1$ and i are connected, the total number of arcs linking all states of a stage is bounded by S^2. For each state transition at maximum all container moves $c \in C$ are inspected to check whether a direct or a split move is counted for the respective area. As the number of cranes n is bounded from above by the number S of slots the overall runtime complexity of DP is bounded by $O(|C| \cdot S^3)$. Thus, DP has polynomial runtime complexity, so that optimal yard partitions can efficiently be determined even for larger instances of real-world size.

5 Yard Simulation

In this section, two research questions are to be answered by a simulation of real-world transshipment yard operations. First, it is investigated whether the surrogate objective of minimizing the workload of loaded moves is indeed a suited objective to reduce the overall workload (consisting of loaded and empty moves). Recall that the surrogate objective is applied to avoid the solution of a complex crane scheduling problem. Furthermore, we aim at a comparison between optimal yard areas and equally sized areas, with the latter being a widespread policy in real-world transshipment yards (see Boysen and Fliedner, 2009). Before describing the results with regard to both research questions (Section 5.2), we first elaborate on the setup of our computational study (Section 5.1).

5.1 Setup of simulation study

To derive test instances for simulating yard operations some assumptions on the yard layout, the container moves to be processed, the parking policy of trains, technical parameters of gantry cranes and the sequencing of crane movements are required. All assumptions are described in detail in the following.

Yard Layout: With regard to the yard layout, we base the study on the typical setting of a German transshipment yard. A typical yard length is 700 meters and slots are adjusted to accommodate standard railcars with a total length of 14 meters. Thus, we assume a yard length of $T = 50$ slots, with a horizontal distance of $d^h = 14$ meters between any two adjacent slots. Furthermore, we assume a vertical distance of $d^v = 7$ meters between neighboring tracks and the sorter, which is located above track one. The number T of parallel tracks and the number n of gantry cranes are varied as follows: $T \in \{2, 3, 4, 5\}$ and $n \in \{2, 3, 4, 5\}$, so that differently sized transshipment yards are investigated.

Container moves: The train length (in slots) is assumed to follow a truncated normal distribution with expected train length $\mu = 43$ (adjusted to an average train length of 600 meters, see Ballis and Golias, 2002) and a standard deviation, which is varied as follows: $\sigma \in \{2, 4, 6, 8\}$. If a train length of more than 50 (less than zero) is drawn, the train length is reduced to the maximum (increased to the minimum) yard length of 50 (zero) slots. The number $|C|$ of container moves is determined by applying parameter $frac \in \{0.2, 0.4, 0.6, 0.8\}$, which is multiplied with the overall number of containers resulting from the different train lengths and divided by 2. Then, for each container move c unique start and target positions (start and target slots: s_c^s and s_c^t as well as start and target tracks: t_c^s and t_c^t with $t_c^s \neq t_c^t$) are randomly drawn out of a uniform distribution.

Parking Policy: In real-world transshipment yards, it is a widespread policy to park locomotives at the beginning of the yard, so that we assume each train's first railcar (may it or not carry a container to be processed) is positioned at slot 1 (see Boysen and Fliedner, 2009). The assignment of trains to tracks is randomly determined, which reflects a first-come-first-serve policy often applied in real-world yards.

Technical crane parameters: The gantry cranes move in horizontal and vertical direction simultaneously propelled by independent engines. In horizontal direction the whole crane moves on special rail tracks, whereas vertically merely the steeple cab carrying the spreader is moved. Thus, the maximum time span for executing the vertical and horizontal movement determines the processing time of a container move. We assume a velocity of crane and steeple cab of $v^e = 3$ meters per second, if the crane moves empty, whereas the velocity reduces to $v^l = 2$ meters per second, if a container is carried. Once positioned, picking and dropping of containers requires additional processing time. Especially, locating the spreader is precision work, so that we assume a typical time span of $t^d = 45$ seconds for picking or dropping a container. See Alicke (2002) and Martinez et al. (2004) for comparable parameters.

Crane Movement: If all container moves are fixed and assigned to gantry cranes, sequencing moves per crane remains a complex optimization problem. In our case, for each crane an asymmetric traveling salesman problem would need to be solved while considering the interdependencies among cranes resulting from split moves. However, the sequence of container moves is typically not optimized by a scheduling procedure but locally determined by the respective crane operator. Thus, to simulate a human decision rule we apply a simple nearest neighbor heuristic. Each crane's starting position is the left hand border of its area, while the steeple cab is positioned over the sorter. From there it consecutively executes the container move closest to its current position. Split moves, e.g., from yard area A to B, are considered by updating crane B's list of unprocessed container moves, not before the respective container arrived in the sorter-segment of crane B. Thus, the list is updated just after crane A processed the first part of the split move (from train into the sorter) and a vehicle (with sorter velocity $v^s = 3$) moved the container into yard area B. With regard to the sorting system it is assumed that vehicles are no bottleneck and congestions do not occur.

The aforementioned parameters of instance generation are summarized in Table 2. All parameters are combined in a full-factorial design and in each parameter constellation instance generation is repeated 100 times, so that $4 \cdot 4 \cdot 4 \cdot 4 \cdot 100 = 25,600$ different instances were obtained.

symbol	description	values
T	number of tracks	2, 3, 4, 5
n	number of gantry cranes	2, 3, 4, 5
S	length of transshipment yard (in slots)	50
d^h	horizontal distance between two adjacent slots (in meters)	14
d^v	vertical distance between two adjacent tracks (in meters)	7
μ	expected train length (in slots)	43
σ	standard deviation of train length	2, 4, 6, 8
$frac$	fraction of all containers being part of a container move	0.2, 0.4, 0.6, 0.8
t^p	processing time for a crane to pick (or drop) a container (in seconds)	45
v^e	velocity of an empty crane carrying no container (in meters per second)	3
v^l	velocity of a loaded crane carrying a container (in meters per second)	2
v^s	velocity of sorter carrying a container (in meters per second)	3

Table 2: Parameters for instance generation

For any of these instances we apply two different policies for partitioning the transshipment yard. First, we determine optimal crane areas, which rely on a balancing of the cranes' workload as described in Section 4. For this purpose during a preprocessing the weights for direct (w_c^D) and slit moves (w_c^{IN} and w_c^{OUT}) need to be determined. To directly represent the resulting processing times weights are calculated as follows:

$$w_c^D = 2 \cdot t^p + \max\left\{\frac{|s_c^s - s_c^t| \cdot d^h}{v^l}; \frac{|t_c^s - t_c^t| \cdot d^v}{v^l}\right\} \quad \forall\, c \in C \tag{7}$$

If processed as a direct move, container move c requires a pick and a drop operation $(2 \cdot t^p)$. Additionally, the time for the actual move is to be added, which amounts to the maximum of the crane's horizontal and vertical distance each weighted with velocity v^l (for a loaded move).

Split moves into (with weight w_c^{IN}) and out of (with weight w_c^{OUT}) the sorter also require a pick and drop operation and the actual movement time, which is required for the vertical movement between rail track and sorter:

$$w_c^{IN} = 2 \cdot t^p + \frac{t_c^s \cdot d^v}{v^l} \quad \forall\, c \in C \tag{8}$$

$$w_c^{OUT} = 2 \cdot t^p + \frac{t_c^t \cdot d^v}{v^l} \quad \forall\, c \in C \tag{9}$$

Note that varying distances between tracks and/or slots (see Alicke, 2002) can easily be integrated into weight calculation. With these weights, optimal crane areas are calculated (with the DP procedure of Section 4) and passed over to the yard simulation, where the resulting overall processing time (makespan) of the actual pulse of trains is determined by simulating each crane (in its respective area) according to the nearest neighbor heuristic. The second policy investigated is to equally size crane areas. Again, the makespan of train processing is determined by the yard simulation. This way, the gap of train processing times between both policies can be computed. The results are summarized in the following section.

5.2 Results

First, we compare optimal results of the surrogate objective (SURR) generated by the DP approach of Section 4 with the actual makespan of the yard simulation (ACT), which takes the yard areas of the DP solution. While the DP approach minimizes the cranes' workload merely on the basis of loaded moves, the yard simulation provides the actual workload consisting of loaded and empty crane moves. On average over all 25,600 instances, the workload of SURR already ranges at 84.79 % compared to that of ACT, which results from the overproportional influence of tedious pick and drop operations. Moreover, the coefficient of correlation between both approaches amounts to a remarkable 0.9973. Thus, the conclusion can be drawn, that our surrogate objective of merely considering loaded moves is a suited simplification. On the one hand, the actual objective of reducing the overall workload is strongly supported and, on the other hand, the solution process is considerably alleviated by excluding a detailed crane scheduling.

Furthermore, we aim at investigating the question whether optimal crane areas enable a considerable reduction of train processing time compared to equally sized areas. For this purpose, we report the average absolute deviation (labeled "avg abs") between both policies with regard to the makespan. Avg abs denominates the acceleration of train processing if optimal crane areas are applied instead of equally sized areas in minutes averaged over all instances of the respective parameter constellation. Furthermore, the average relative deviation (labeled "avg rel") of both policies in percent is reported, where the deviation is measured by $\frac{F(EQU)-F(OPT)}{F(OPT)} \cdot 100$ with $F(EQU)$ and $F(OPT)$ being the makespan when crane areas are equally sized or optimally partitioned, respectively. Table 3 lists both performance measures in dependency of the parameters: number T of tracks and number n of cranes, which together reflect the size of a transshipment yard.

	n				
T	2	3	4	5	total
2	4.1/13.1	3.4/16.0	3.2/18.7	2.9/19.8	3.4/16.9
3	6.3/12.5	5.3/16.2	4.8/19.0	4.6/21.5	5.2/17.3
4	8.7/13.0	7.1/16.1	6.2/18.5	6.0/21.7	7.0/17.3
5	10.9/12.9	8,7/15.5	7.9/18.3	7.3/21.0	8.7/16.9
total	7.5/12.9	6.1/15.9	5.5/18.6	5.2/21.0	6.1/17.1

legend: avg abs [minutes]/avg rel [%]

Table 3: Absolute and relative speed-up of train processing depending on yard size

The results reveal a remarkable potential for accelerating train processing. Depending on the size of the yard, possible absolute accelerations (avg abs) deviate between 10.9 minutes with five tracks (high overall workload) and two cranes (low division of labor) and 2.9 minutes with two tracks (low overall workload) and five cranes (high division of labor). Interestingly, the relative acceleration (avg rel) of train processing performs somewhat contrarily. This is explained by the fact, that with a high division of labor the average makespan tends to be lower in value, because a pulse of trains is processed much faster. As a consequence a comparable absolute reduction in makespan leads to a higher relative reduction. In relative terms, train processing is accelerated by between 12,5 % and 21,7 % whenever crane areas are dimensioned optimally.

Further conclusions (in terms of a sensitivity analysis) can be drawn if the speed-up of optimal crane areas is related to the parameters of instance generation. Therefore, Figure 5 displays the average relative deviation (avg rel in %) and the average absolute deviation (avg abs in minutes) in dependency of the parameters: number T of tracks, number n of cranes, standard deviation σ of train lengths and fraction $frac$ of containers to be processed, respectively.

While σ seems to have negligible influence on the results, the following conclusions can be drawn for the other parameters:

- With an increasing number T of tracks the overall workload is increased and, thus, the absolute speed-up (avg abs) of optimal crane areas also rises. This effect is counterbalanced by the increasing level of workload, so that the relative speed-up (avg rel) remains on a constant level.
- The higher the division of labor (more cranes n), the lower absolute (avg abs) and the higher relative (avg rel) speed-ups of optimal crane areas. The relative gap is

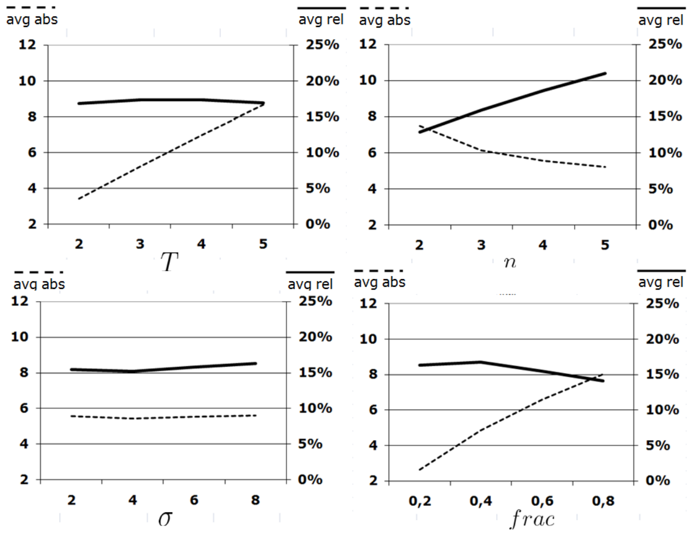

Figure 5: Absolute and relative speed-up in dependency of parameters of instance generation

also influenced by the fact that with an increasing number of cranes, yard areas become smaller. With fairly short crane areas the probability of unbalanced slots being leveled by others decreases.

- If the fraction ($frac$) of railcars which carry containers increases, then the overall workload rises. Thus, the absolute speed-up (avg abs) by optimal crane areas increases, as well. Again this effect is leveled by the general increase in workload, so that avg rel remains nearly unaffected.

It can be concluded that a more leveled workload evenly spread over the yard reduces the disadvantage of equally size crane areas. However, as the results over all instances indicate an average absolute speed-up of 6.1 minutes (or 17.1%) optimal yard areas can be recommended independent of the respective real-world transshipment yard setting.

6 Conclusion

In this work a partition problem is introduced which aims at evenly balancing the workload among gantry cranes of rail-rail transshipment yards. In these yards, the partitioning needs to especially consider time-consuming split moves, where a single container move is to be processed by two cranes, which are connected by a sorting system. The problem is formalized and an exact Dynamic Programming procedure with polynomial runtime is developed. In a comprehensive computational study possible accelerations of train processing are evaluated by a comparison between optimally and equally sized crane areas, the latter being a widespread policy in practice. The results reveal that an optimal repartitioning of crane areas can result to significant speed-ups in train processing.

There are several ways in order to build up on this study in future research. On the one hand, partitioning the transshipment yard into crane areas is heavily interdependent with determining the horizontal (parking position along the yard) and vertical (track assignment) positions of trains. Thus, further improvements of train processing might be gained by additionally optimizing the parking positions of trains. On the other hand, it would be valuable to quantify the disadvantages (with regard to train processing times) of fixed yard areas compared to a dynamic assignment of container moves to cranes (see Section 2). This could provide valuable decision support for assessing dynamic assignment policies, so that yard managers could evaluate the trade-off between the acceleration of train processing and the investment cost required for a sophisticated online-control system.

References

[1] Abacoumkin, C., Ballis, A., 2004. Development of an expert system for the evaluation of conventional and innovative technologies in the intermodal transport area, European Journal of Operational Research 152, 410–419.

[2] Alicke, K., 2002. Modeling and optimization of the intermodal terminal Mega Hub, OR Spectrum 24, 1–17.

[3] Alicke, K., Arnold, D., 1998. Optimierung von mehrstufigen Umschlagsystemen, Fördern und Heben 8, 769–772.

[4] Ballis, A., Golias, J., 2002. Comparative evaluation of existing and innovative rail-road freight transport terminals, Transportation Research Part A 26, 593–611.

[5] Ballis, A., Golias, J., 2004. Towards the improvement of a combined transport chain performance, European Journal of Operational Research 152, 420–436.

[6] Bontekoning, Y.M., Macharis, C, Trip, J.J., 2004. Is a new applied transportation research flied emerging? A review of intermodal rail-truck freight transport literature, Transportation Research Part A 38, 1–24.

[7] Bostel, N., Dejax, P., 1998. Models and algorithms for container allocation problems on trains in a rapid transshipment shunting yard, Transportation Science 32, 370–379.

[8] Boysen, N., Fliedner, M., 2009. Determining crane areas in intermodal transshipment yards: The Yard Partition problem, Jena Research Papers in Business and Economics (JBE) 4/2009, FSU Jena Germany.

[9] Boysen, N., Pesch, E., 2008. Scheduling freight trains in rail-rail transshipment yards, Jena Research Papers in Business and Economics (JBE) 11/2008, FSU Jena Germany.

[10] Cordeau, J.-F., Toth, P., Vigo, D., 1998. A survey of optimization models for train routing and scheduling, Transportation Science 32, 380–404.

[11] Corry, P., Kozan, E., 2006. An assignment model for dynamic load planning of intermodal trains, Computers & Operations Research 33, 1–17.

[12] Corry, P., Kozan, E., 2008. Optimised loading patterns for intermodal trains, OR Spectrum 30, 721–750.

[13] Crainic, T.G., Kim, K.H., 2007. Intermodal transport, In: Barnhart, C., Laporte, G. (eds.) Transportaion, Handbooks in Operations Research and Management Science 14, 467–538.

[14] EU, 2007. Mitteilung der Kommission an den Rat und das europäische Parlament: Aufbau eines vorrangig für den Güterverkehr bestimmten Schienennetzes, Brüssel 2007.

[15] Garey, M.R., Johnson, D.D., 1979. Computers and Intractability: A Guide to the Theory of NP-Completeness, W. H. Freeman, San Francisco.

[16] Lim, A., Rodrigues, B., Xu, Z., 2007. A m-parallel crane scheduling problem with a non-crossing constraint, Naval Research Logistics 54, 115–235.

[17] Macharis, C., Bontekoning, Y.M., 2004. Opportunities for OR in intermodal freight transport research: A review, European Journal of Operational Research 153, 400–416.

[18] Martinez, F.M., Gutierrez, I.G., Oliveira, A.O., Bedia, L.M.A., 2004. Gantry crane operations to transfer containers between trains: A simulation study of a Spanish terminal, Transportation Planning and Technology 27, 261–284.

[19] Meyer, P., 1998. Entwicklung eines Simulationsprogramms für Umschlagterminals des Kombinierten Verkehrs, Ph.D. thesis, Universität Hannover.

[20] Moccia, L., Cordeau, J.F., Gaudioso, M., Laporte, G., 2006. A branch-and-cut algorithm for the quay crane scheduling problem in a container terminal, Naval Research Logistics 53, 45–59.

[21] Ng, W.C., 2005. Crane scheduling in container yards with inter-crane interference, European Journal of Operational Research 164, 64–78.

[22] Rotter, H., 2004. New operating concepts for intermodal transport: The Mega Hub in Hanover/Lehrte in Germany, Transportation Planning and Technology 27, 347–365.

[23] Sammarra, M., Cordeau, J.F., Laporte, G., Monaco, M.F., 2007. A tabu search heuristic for the quay crane scheduling problem, Journal of Scheduling 10, 327–336.

[24] Wiegmans, B.W., Stekelenburg, D.T., Versteegt, C., Bontekoning, Y.M., 2006. Modeling rail-rail exchange operations: An analysis of conventional and new-generation terminals, Transportation Journal 2006(3), 5–20.

[25] Zhu, Y., Lim, A., 2006. Crane scheduling with non-crossing constraints, Journal of Operational Research Society 57, 1472–1481.