Friedrich-Schiller-Universität Jena
seit 1558

Jena Research Papers in Business and Economics

Determining Crane Areas in Intermodal Transshipment Yards: The Yard Partition Problem

Nils Boysen, Malte Fliedner

04/2009

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 Crane Areas in Intermodal Transshipment Yards: The Yard Partition Problem

Nils Boysen, Malte Fliedner

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

Abstract
At rail-road transshipment yards, gantry cranes move containers from freight trains to trucks and vice versa. They constitute important entities in today's intermodal transportation systems. Real-world yards are often partitioned into several disjunct crane areas, so that crane interferences during container transshipment are avoided. In practice, the lengths of such crane areas are typically determined by simple rules of thumb, i.e., each crane receives an equally sized area, which might result in an unleveled division of labor among cranes and, thus, prolong train processing times. This paper provides an exact solution procedure which determines disjunct yard areas of varying size for multiple gantry cranes in polynomial runtime, so that the workload for a given pulse of trains is equally distributed among cranes. Furthermore, we investigate the potential acceleration of train processing as compared to equally sized areas in a yard simulation.

Keywords: Intermodal transport; Transshipment yard; Container handling; Crane scheduling

1 Introduction

Intermodal transport can be defined as the successive use of "various modes of transportation (road, rail, air and water) without any handling of the goods themselves during transfers between modes" (European Conference of Ministers of Transport, 1986). Due to the steadily increasing volume of transported goods, it has become a key concept for executing global transportation in an economically and ecologically efficient manner. In rail-road intermodal systems, freight trains move massive quantities of containers over long distances (long-haul transportation), whereas trucks are employed for short-distance pick-up and delivery activities. To enable the container transfer from road to rail and vice versa, modern transshipment yards are required, where huge gantry cranes move containers between both modes of transportation. Figure 1 gives a schematic representation of a rail-road transshipment yard.

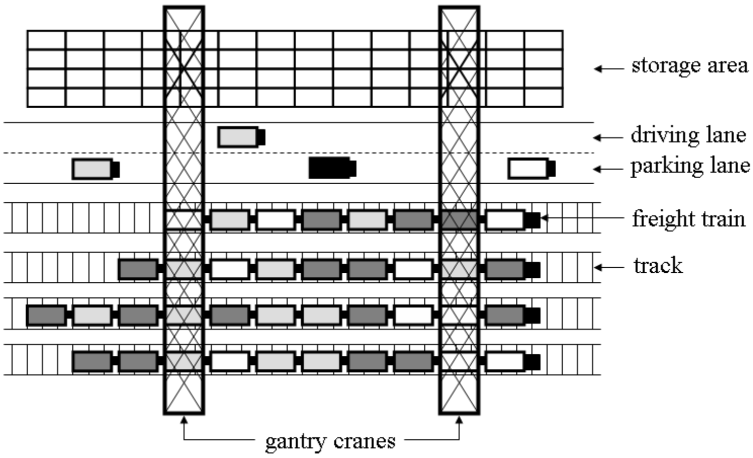

Figure 1: Schematic representation of a transshipment yard

A transshipment yard consists of a given number of parallel railway tracks, a storage area to intermediately stock containers and additional truck lanes (typically one lane for parking and another one for driving, see Ballis and Golias, 2002). Multiple gantry cranes with a cantilever on both sides, which span over tracks, storage area and truck lanes, transfer containers between trucks and railcars. The exact number of railway tracks and gantry cranes differs with regard to the size of the yard. The largest German transshipment yard in Köln-Eifeltor, for instance, consists of nine parallel tracks and six successively arranged gantry cranes.

Typically, gantry cranes move horizontally on the same specialized crane tracks which run along the sides of the yard. As a consequence gantry cranes cannot pass each other and interferences occur, whenever a move of crane A is blocked by another crane B operating in between the actual and the target position of crane A. A common policy to avoid interferences is to assign a dedicated crane area to each crane (referred to as static assignment), so that all container moves within an area are exclusively processed by the respective gantry crane. The main disadvantage of this strategy is that cranes are restricted to stay within the borders of their area, so that they cannot assist with container moves of adjacent cranes. Nevertheless, a dynamic assignment of container moves to cranes would need to be coordinated in real-time, e.g., by information systems employing complex online scheduling procedures. Such a centralized online control 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.) without worrying about the positions of other cranes. Thus, static crane areas are the typical choice in real-world rail-road transshipment yards.

As part of the daily operations, all gantry cranes process parked trains in parallel until all containers are transshipped and another set of trains (labeled pulse, see Bostel and Dejax, 1998; Rotter, 2004) enters the yard. Thus, all cranes have to wait - thereby incurring idle time - until the last crane has finished container processing. Consequently, it is an important objective to equally share the workload among cranes, i.e., to minimize the cranes' maximum processing time. However, for a given pulse of trains (with given positions of containers to be processed) each crane's workload heavily depends on its assigned yard area. The paper on hand presents an exact Dynamic Programming approach with polynomial runtime complexity which partitions the yard into disjunct crane areas, so that the resulting workload is equally distributed among cranes. This problem is denominated as Yard Partition problem. Additionally, we question a common real-world policy which assigns equally sized areas to each crane. In a straightforward yard simulation, we compare optimal crane areas with equally sized ones and reveal a significant potential for processing time reduction.

For this purpose the remainder of the paper is structured as follows. Section 2 provides a literature review on yard scheduling and structures the associated decision problems. Section 3 gives a detailed description of the Yard Partition problem for which an exact Dynamic Programming procedure is developed in Section 4. Then, optimally and equally sized crane areas are compared in a simulation of transshipment yard operations. Section 5 describes the design of the simulation study whose results are presented in Section 6. Finally, Section 7 concludes the paper with an outline of future research.

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, specific literature on transshipment yards is scarce. This is astounding since transshipment yards are an emerging technology in railway systems (see the surveys by Bontekoning et al., 2004 as well as Macharis and Bontekoning, 2004) and critical to enable intermodal rail-road transport.

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 complex to allow a simultaneous solution, a hierarchical decomposition of the overall problem is recommended (see Boysen and Pesch, 2008):

(i) Schedule the service slots of trains.

(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), who introduce a train scheduling problem for rail-rail transshipment yards. This special kind of yard is dedicated to transship containers exclusively between different freight trains, in order to consolidate container transport in a hub-and-spoke railway network. In this setting, train pulses are to be determined in such a way, that double handling due to intermediate storage of freight or revisits of already processed trains is minimized. 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 at the yard are minimized. 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. With a static assignment, fixed crane areas are applied and containers are exclusively processed by the crane assigned to their respective yard area. On the other hand, a dynamic assignment abstains from fixed areas and dynamically coordinates cranes so that crane interferences need to be continuously controlled for. Existing literature mainly investigates the latter policy. 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 within container terminals in the hinterland of seaports (see, e.g., Ng, 2005; Zhu and Lim, 2006; Moccia et al., 2006; Lim et al., 2007; Sammarra et al., 2007). The paper on hand is the first to treat problem (iv) with fixed crane areas within a rail-road transshipment yard and provides an exact solution procedure. Note that our solution procedure can also be helpful for a dynamic crane assignment, i.e., to determine a first feasible start solution.

3 Detailed Problem Description

Determining appropriate crane areas is an operational problem, so that the layout of the transshipment yard, e.g., the number and length of the parallel railway tracks and the number n of cranes, is fixed. Furthermore, we assume that a pulse of trains (one train per track) is already parked at the yard. Note that this train set might be an actual pulse of trains already waiting or being expected at the yard. In this case, crane areas are newly determined for any pulse of trains, which can easily be implemented 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, the given 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 and truck drivers, a transshipment yard is subdivided into smaller line segments referred to as slots, whose numbers and borders are drawn on the truck lanes 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, which covers the size of a railway car of type "Lgs580" carrying one forty feet container (FEU) or two twenty feet containers (TEU). The typical yard length is 700 meters, so that such a representative transshipment yard would be subdivided into $T = 50$ slots. We presuppose that freight trains are parked and adjusted in such a way that each container exactly falls into one slot. Thus, aggregated over all tracks the resulting workload of each slot can be determined.

With these input parameters, it is the aim of the Yard Partition problem (YPP) to divide the slots $t = 1, \ldots, T$ of a transshipment yard into n disjunct crane areas, so that the maximum workload is minimized over all cranes $i = 1, \ldots, n$. As cranes process a given pulse of trains in parallel, all cranes have to wait idle until the last crane finishes container transhipment. The min-max objective therefore balances the workload among cranes and reduces the total processing time for a pulse of trains.

We assume that the actual workload w_t induced by containers parked at slot t, is determined in a preprocessing phase by assigning a specific weight to each container. In the simplest case this weight would be equal to one, so that only the number of containers per slot were counted. However, weights can also cover crane distances or even the exact processing time of a container. Figure 2 shows an example of how to preprocess the workload w_t of slots $t = 1, \ldots, T$.

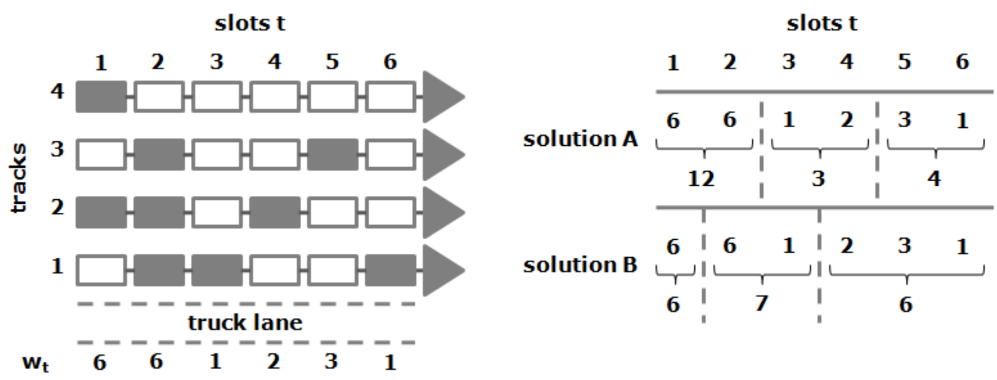

Figure 2: Example data

Example: In the example of Figure 2 a transshipment yard is subdivided into six slots. On four parallel tracks a pulse of trains is parked, where grey boxes symbolize containers to be transshipped from train to trucks, which are assumed to wait in the slot of the respective container. White boxes represent containers which remain on the train, e.g., because they are dedicated to the trains' next destinations. If each slot's workload is to be approximated by the number of containers per slot, workload w_t of slots 1 to 6 amounts to: $w_1 = 2$, $w_2 = 3$, $w_3 = 1$, $w_4 = 1$, $w_5 = 1$ and $w_6 = 1$, respectively. However,

n	number of gantry cranes, where cranes $i = 1, \ldots, n$ are ordered from left to right
T	number of slots in the yard (with $t = 1, \ldots, T$)
w_t	weight of slot t, e.g., time span for processing all containers of slot t
x_i	integer variable: last slot of the yard area of crane i

Table 1: Notation

as the length of crane moves varies with the tracks from which containers are fetched, a more appropriate weight could consider aggregated crane distances per slot. In Figure 2 each container is weighted with the number of tracks in between the container position and the truck lane. Thus, the weight of the first slot amounts to: $w_1 = 4 + 2 = 6$. Another alternative is to weight each container with the actual processing time. Such an approach is chosen in the simulation study of Section 5.

Weights w_t per slot $t = 1, \ldots, T$ and the number n of cranes form the basic input parameters of a YPP-instance. On the right-hand side of Figure 2 two alternative solutions for the YPP are displayed. Solution A chooses equally sized yard areas, so that for a given number of $n = 3$ crane areas, borders are located after slots 2 and 4, respectively. The maximum workload of 12 is assigned to the first crane. Solution B is the optimal solution with area borders after slots 1 and 3, respectively. Here, the workload is evenly balanced with a maximum of 7 accumulating for crane 2.

Making use of the notation summarized in Table 1, the YPP can be formalized as a mathematical program by objective function (1) and constraints (2) to (5):

$$\text{(YPP) Minimize } Z(X) = \max_{i=1,\ldots,n} \left\{ \sum_{t=x_{i-1}+1}^{x_i} w_t \right\} \tag{1}$$

subject to

$$x_i \geq x_{i-1} + 1 \qquad \forall\, i = 1, \ldots, n \tag{2}$$

$$x_n = T \tag{3}$$

$$x_0 = 0 \tag{4}$$

$$x_i \in \mathbb{N} \qquad \forall\, i = 1, \ldots, n \tag{5}$$

In objective function (1) the maximum workload over all cranes $i = 1, \ldots, n$ is minimized, where the workload of crane i is determined by adding up workload w_t over all assigned slots t from the left border ($x_{i-1}+1$) up to the right border (x_i) of the respective crane area. Constraints (2) ensure that cranes are considered consecutively and a minimum area length of one slot is imposed. Furthermore it is enforced that the yard area of the first crane starts in the first slot (equation (3)) and last crane's area ends in the last slot (equation (4)). Finally, variables x_i are defined as integer variables (constraints (5)).

4 A Dynamic Programming Procedure

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. A stage i comprises a set of states (i, t), where each state represents a possible right-hand border (slot t) of the respective crane area i. As the minimum area width is a single slot, each stage $i = 1, \ldots, n-1$ exactly contains $T - n + 1$ states. The final stage n merely contains a single state (n, T), because the last crane n has to cover the remaining yard length up to final slot T (see equation (3)).

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

$$f_{1t} = \sum_{i=1}^{t} w_t \quad \forall t = 1, \ldots, T - n + 1 \tag{6}$$

Then, partial objective values f_{it} can be calculated for each state of the remaining stages by a basic recursion as follows:

$$f_{it} = \min_{i-1 \leq j < t-1} \left\{ \max \left\{ f_{i-1j}; \sum_{\tau=j+1}^{t} w_\tau \right\} \right\} \quad \begin{array}{l} \forall i = 2, \ldots, n-1,\ t = i, \ldots, i + T - n \\ \text{and } i = n,\ t = T \end{array} \tag{7}$$

Here, partial objective value f_{it} of a state (i, t) 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: $j = i - 1, \ldots, t - 1$. For any of these possible predecessor states, the workload amounts to the maximum of partial objective value f_{i-1j} of predecessor state $(i - 1, j)$ (first term of maximum function) and the additional workload of the actual yard area of crane i ranging from slot $j + 1$ to t (second term of maximum function). Lastly, final state (n, T) is reached and f_{nT} 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 which are part of the optimal solution.

Example (cont.): For the input data of Figure 2 the resulting DP graph is shown in Figure 3. The bold sketched optimal solution value for three cranes amounts to a min-max workload of 7 associated with crane 2. Optimal yard areas range up to slot 1, 3 and 6, respectively. Note, that in an alternative optimal solution borders are located after slots 1 and 2, respectively.

The DP procedure considers n stages, while each stage contains less than T nodes. All nodes of a stage i are only connected to nodes of the adjacent stage $i + 1$, thus, the number of edges to be investigated is bounded by $O(n \cdot T^2)$. Furthermore, due to (2)

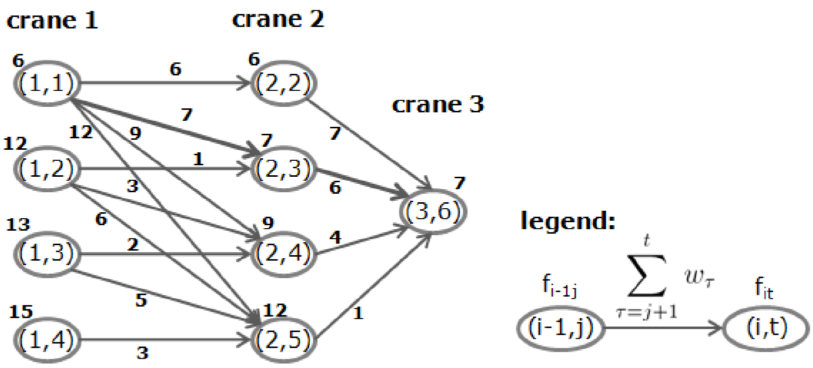

Figure 3: DP graph for the example

for feasible instances the number of cranes n is bounded from above by the number of slots T. It follows that the DP procedure has a polynomial runtime complexity of $O(T^3)$, so that optimal yard partitions can efficiently be determined even for large instances of real-world size.

5 Design of Yard Simulation

In this section, we elaborate on the setup of a comprehensive computational study, which simulates real-world transshipment yard operations. In this study, we investigate possible accelerations of train processing if optimal yard areas are applied instead of equally size areas, with the latter being a common real-world policy. To derive test instances some assumptions on the yard layout, the composition of freight trains with containers, the way trains and trucks are parked on the yard (parking policy), technical parameters of gantry cranes and the sequencing of crane movements are required. All of these are described in the following.

Yard Layout: The investigated yard layouts are based on German transshipment yards. Here, a typical yard length is 700 meters and slots are adjusted to standardized railcars having a 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 typical vertical distance of $d^v = 7$ meters between neighboring tracks. Finally, the number G of parallel tracks and the number n of gantry cranes are varied as follows: $G \in \{2, 3, 4, 5\}$ and $n \in \{2, 3, 4, 5\}$, so that differently sized transshipment yards are investigated.

Train Composition: 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. For each railcar of a train parameter $Prob$ decides whether or not it carries a container to be processed. Whenever a uniform random number out of the interval $[0, 1]$ is smaller than the actual value of parameter $Prob$ the respective container is selected to be transshipped. Otherwise it is assumed that the respective railway car remains empty or carries a container dedicated to a later destination of the train. To investigate different workloads in the yard, $Prob$ is varied as follows: $Prob \in \{0.4, 0.6, 0.8, 1.0\}$. We further assume a unidirectional container flow, so that all containers are assumed to arrive on train and to leave the yard by truck.

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 may it not carry a container to be processed) being located at slot 1. Furthermore, a parking policy for trucks is to be established. To reduce the distance of container movements, trucks aim at parking directly next to their respective container. If multiple containers on different tracks are located in the same slot, we assume that truck drivers have enough time between two container moves to manoeuvre trucks. Thus, we presuppose any truck waits for its container in the same slot.

Technical crane parameters: A gantry crane moves in horizontal and vertical direction simultaneously making use of different engines. In horizontal direction the whole crane moves on special crane tracks, whereas vertically merely the steeple cab carrying the spreader is moved. Thus, the maximum time span for vertical and horizontal movement decides on the processing time of the 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 on average merely 20-25 moves per crane and hour can be processed (see Rotter, 2004). Thus, 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) who apply comparable crane parameters.

Crane Movement: In a yard setting with static container assignment, 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 the following simple policy for crane movement is applied. Each crane starts container processing at its left-hand area border. Then, the crane transships all containers of the respective slot from train to truck (according to ascending distance of tracks) and moves to the next slot until it reaches its right-hand border.

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 10 times, so that $4 \cdot 4 \cdot 4 \cdot 4 \cdot 10 = 2560$ different instances were obtained.

symbol	description	values
G	number of tracks	2, 3, 4, 5
n	number of gantry cranes	2, 3, 4, 5
T	length of transshipment yard (in slots)	50
d^h	horizontal distance between two adjacent slots (in meter)	14
d^v	vertical distance between two adjacent tracks (in meter)	7
μ	expected train length (in slots)	43
σ	standard deviation of train length	2, 4, 6, 8
$Prob$	probability of a railway car carrying a container to be processed	0.4, 0.6, 0.8, 1.0
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

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. As a weight for each single container move $c \in C$ with distance d_c between train and truck we calculate processing time t_c as follows:

$$t_c = d_c \cdot v^l + 2 \cdot t^p + d_c \cdot v^e \quad \forall\, c \in C \tag{8}$$

Processing time t_c consists of an empty move of the crane from the truck lane to the container's position on train, a pick-operation, a loaded move back to the truck lane and a container drop on truck. Weights w_t of slot t amount to the sum of all processing time of containers C_t located in slot t: $w_t = \sum_{c \in C_t} t_c \,\forall\, t = 1, \ldots, T$. 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. 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.

6 Results

As a performance measure, 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 each single deviation is measured by $\frac{P(EQU)-P(OPT)}{P(OPT)} \cdot 100$ with $P(EQU)$ and $P(OPT)$ being the makespan when crane areas are equally sized or optimally partitioned, respectively. Table 3 list both measures in dependency of the parameters number G of tracks and number n of cranes, which together reflect the size of a transshipment yard.

	n				
G	2	3	4	5	total
2	7.6/16.0	7.3/24.4	6.2/26.4	4.8/27.2	6.4/23.5
3	12.4/17.1	10.5/22.4	9.4/24.1	6.7/26.2	9.8/22.5
4	17.3/16.8	14.6/21.9	12.7/22.9	8.7/26.0	13.3/21.9
5	21.7/16.3	17.9/20.6	16.2/21.8	10.9/25.0	16.7/20.9
total	14.7/16.6	12.6/20.6	11.1/23.8	7.8/26.1	11.6/22.2

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

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

The results reveal a significant potential for accelerating train processing. Depending on the size of the yard, possible absolute accelerations (avg abs) deviate between 21.7 minutes with five tracks (high overall workload) and two cranes (low division of labor) and 4.8 minutes with two tracks (low overall workload) and five cranes (high division of labor). Interestingly, the relative acceleration (avg rel) of train processing performs contrarily. This is explained by the fact, that with a low overall workload and a high division of labor the resulting makespans are on a much lower level, since a pulse of trains is processed much faster. Therefore, a similar absolute reduction in makespan leads to a much higher relative reduction. In relative numbers train processing is accelerated between 27.2 % and 16.0 % by applying optimal crane areas.

Further conclusions (in the sense 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 4 displays the average relative deviation (avg rel in %) and the average absolute deviation (avg abs in minutes) in dependency of the parameters: number G of tracks, number n of cranes, standard deviation σ of train length and probability $Prob$ of a railcar carrying a container to be processed, respectively.

From these charts the following conclusions can be drawn:

- With an increasing number G of tracks the overall workload increases and, thus, the absolute speed-up (avg abs) of optimal crane areas raises as well. In turn, an increasing overall workload reduces the relative acceleration (avg rel). Furthermore, with only a few tracks already a single shorter train leads to a rather unleveled workload in equally size crane areas. With more tracks the probability of shorter trains being compensated by longer ones and, thus, a more leveled workload is much higher. Hence, avg rel decreases with additional railway tracks.

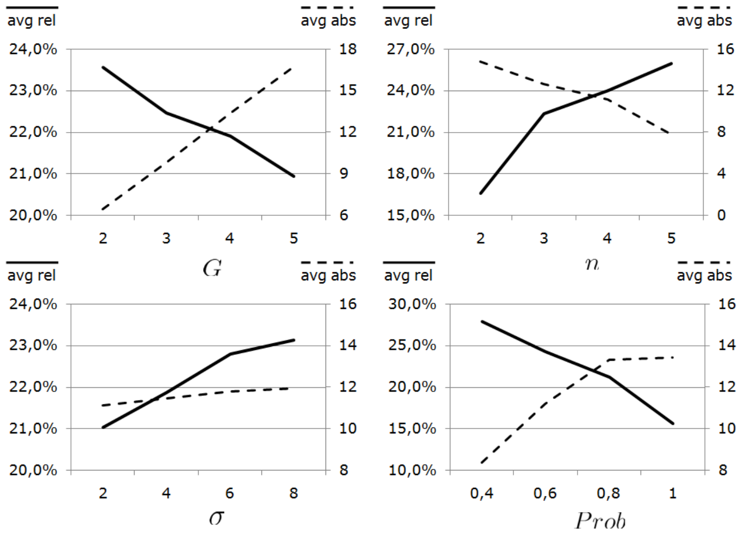

Figure 4: Absolute and relative speed-up in dependency of parameters of instance generation

- 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 become. The relative gap is also influenced by the fact that with an increasing number of cranes, yard areas per crane become shorter. With fairly short crane areas the probability of unbalanced slots being leveled by others decreases.

- With increasing standard deviation σ train lengths more and more vary, which in turn increases the imbalance of the overall workload and, thus, enlarges the disadvantage (measured by avg rel) of equally sized crane areas. On the other hand, the overall workload remains nearly unaffected by an increasing variation of train lengths, so that avg abs does not vary considerably.

- If the probability $Prob$ of railway cars carrying a container to be processed increases, the overall workload raises. Thus, the absolute speed-up (avg abs) induced by optimal crane areas increases as well. On the other hand, the imbalance of workload among crane areas decreases in $Prob$. In the extreme case, when all containers are to be processed ($Prob = 1$), differences of workload between cranes is exclusively caused by diverging train lengths. Thus, the relative speed-up (avg rel) decreases with increasing $Prob$.

Even though the sensitivity analysis revealed that performance improvements depend to some extend on the parameter constellation, the overall speed-up of 11.6 minutes (or 22.2%) on average suggests that optimal yard areas can be recommended independent of the respective real-world transshipment yard setting.

7 Conclusions

The paper on hand introduces the Yard Partition problem, which aims at evenly distributing the total workload among all gantry cranes of a transshipment yard by an appropriate dimensioning of crane areas. The resulting problem is formalized and an exact Dynamic Programming procedure with polynomial runtime complexity is developed. In a comprehensive computational study the potential acceleration of train processing is evaluated as compared to a wide-spread real-world policy of equally sized crane areas. The results reveal that significant speed-ups can be realized, the extent of which depends on the actual yard setting. Thus, it is advisable for real-world transshipment yards which make use of disjunct crane areas to optimally determine them in accordance with the actual pulse of trains. Moreover, such fixed but varying crane areas are easy to implement in practice. While a dynamic assignment of container moves to cranes requires a complex online control of cranes and a continuous online optimization of container moves, so that crane interferences are avoided, fixed areas can simply be determined with every new pulse of trains and communicated to the crane operator. Such a communication can simply be implemented 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. This way, crane interferences are ruled out with certainty and crane operators can immediately and autonomously react on unforseen events.

However, there remain some future research challenges. On the one hand, partitioning the transshipment yard into crane areas is closely interdependent with determining the horizontal and vertical parking positions of trains. Thus, further improvements of train processing might be gained by additionally varying the parking positions of trains. On the other hand, alternative transshipment yards exclusively dedicated to a rail-rail container transshipment exist (see Alicke, 2002; Boysen and Pesch, 2008). Here, containers might need to be moved from one crane area to another, so that the total scheduling problem cannot be easily decomposed. For this kind of yards, determining fixed crane areas remains an open challenge.

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

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

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

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

[12] 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.

[13] European Conference of Ministers of Transport, 1986. Terminology for Combined Transport. http:/www.cordis.lu/transport.

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

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

[16] 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.

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

[18] 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.

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

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

[21] 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.

[22] 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.

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