Logistics Research (2018) 11:6
DOI_10.23773/2018_6

Optimization of coil relocations in multilocation capacitated warehouses

Karl Schneeberger[1] · Karl F. Doerner[1, 2] · Michael Schilde[1]

Received: 9 Mai 2017 / Accepted: 24 Mai 2018 / Published online: 2 July 2018

ABSTRACT

In this paper, we consider a steel coil storage and distribution problem for a steel producing company. The company has several warehouses, and relocations of steel coils to different warehouses before they are dispatched are inevitable. These relocations are driven by a lack of capacity or the necessity of a relocation (e.g., if a coil is dispatched by ship on a specific day, the coil must be relocated to the designated warehouse). These relocations are performed by specific truck and trailer and are time-consuming and costly. In an effort to bring greater efficiency to this problem, we developed a fast matheuristic solution method that has a simple design and performs well. We apply our technique to a set of large real world data (with up to 87 days). To evaluate the performance for realistic relocations, we consider many different sizes of coils (considering 225 different coil types, varying in dimensions). Therefore, the complexity of the problem to solve is high and many decisions concerning the best relocations must be made. The results of the matheuristic approach show that it is possible to improve the current solution of the steel producing company by as much as 11% for larger test cases (considering a time frame of at least 30 days). In addition, we provide insights on the value of information (no information versus all information about the production and distribution). Our results show that more information leads to the better results (i.e., less relocations). But, also with no information on further production and distribution, we can improve the current real world solutions.

KEYWORDS: steel coils · coil relocation · capacitated multilocation warehouses

Karl Schneeberger[1]
karl.schneeberger@univie.ac.at

Karl F. Doerner[1, 2]
karl.doerner@univie.ac.at

Michael Schilde[1]
michael.schilde@univie.ac.at

[1] University of Vienna, Department of Business Administration, Christian-Doppler-Laboratory for Efficient Intermodal Transport Operations, Oskar-Morgenstern-Platz 1, 1090 Vienna, Austria

[2] Data Science, University of Vienna, Vienna, Austria

1 INTRODUCTION

We consider a real world problem of a steel producing company, namely, their steel coil distribution (dispatch and relocation of coils) and storage problem. The company's primary problem is finding the best-fitting warehouse and doing as few coil relocations of coils as possible. Steel coils (also simply called "coils") are finished steel products that are wrapped after rolling the metal in kilometer long strips. These coils are very heavy (up to 36 tons) and very large. As a result, many coils have to be dispatched by train or ship because they are too heavy and oversized for transport by truck due to legal restrictions. Handling the coils is also difficult, and most need to be moved by crane (within the warehouse) and then transported to other warehouses by special heavy trucks and trailers or forklifts that are able to transport coils.

For the coil distribution and storage problem, we consider a medium-term time horizon (up to 87 days) in which we have to decide in advance which coils to relocate from one warehouse to another to minimize the number of subsequent relocations between different warehouses. In addition, each warehouse has a different total capacity available to store different coil dimensions. These dimensions are divided into width (b) and diameter (r) of a coil and the number of possible storage locations. The warehouses in question are only a few hundred meters away from one another. Loading and unloading is the most time consuming part (the traveling time is negligible) and therefore only the number of relocations is minimized and not the total distance traveled.

Coils can be stored on pallets if necessary or without any loading equipment. The coils without any loading equipment are stored on wedges, which are fixed in the ground and therefore hold the coils in place. Another possibility is to store coils on top of other coils. In this instance, at least two coils have to be stored together in the same shape to place a third coil on top of these (this is allowed only for coils without any loading equipment). Coils with and without any loading equipment can also be stored in high-bay racking warehouses if they are not too heavy (maximum weight must be considered).

Relocations occur because each coil is distributed from a production plant to a directly connected warehouse. The incoming coils are stored or dispatched at the same day. Relocations must be performed when there is a lack of available capacity (see Figure 1; for two coils is only one capacity left, therefore a relocation of one must be made) or if a coil needs to be relocated to another warehouse (e.g., it is dispatched with other coils by train or ship from another warehouse). Internal relocations within the same warehouse are not considered in this work, because there is sufficient crane capacity in all the warehouses to facilitate intrawarehouse relocations.

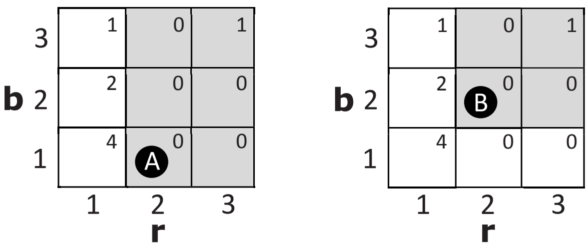

Figure 1: The two parts of the figure show coil dimensions b and r and the number of available storage slots. For example, there are four storage slots available for coils with dimensions $b = 1$ and $r = 1$. A coil can be stored in a slot of its same dimensions (e.g., coil A with $b = 1$ and $r = 2$) or in a slot with larger dimensions ($b \geq 1$ and $r \geq 2$); this is shown as "gray area" for coil A. Considering only coil A, the problem is feasible because one storage slot is available ($b = 3$ and $r = 3$). If coil B is also considered and there is only one storage slot available for coils A and B, then one of them must be relocated.

The relocation of coils to other warehouses are mainly due to necessity or lack of capacity (we can only optimize the relocations due to a lack of capacity). In either case, coils must be relocated to facilitate a feasible solution to the problem. The numerical experiments (see Section 4) and, in more detail, a real world case study show that the relocations can be reduced relative to the real world relocations the company is making. In practice, the dispatcher is responsible for the relocations and dispatches made in real world. At the moment, the dispatcher has no decision support and therefore relocations and dispatches are made based on his/her expert knowledge.

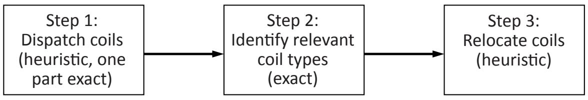

Figure 2: Overview of the solution approach.

Figure 2 provides an overview of the solution approach. We offer a detailed description of each step in Section 3. First, we dispatch all available coils by considering the maximum dispatch capacity (heuristic, one part exact; see Section 2.2). The dispatch capacity is the maximum amount of coils that can be sent within a day. Second, we identify coil types (with dimensions b and r) that need to be relocated due to a lack of capacity (exact; see Section 2.3). After identifying these coil types we relocate the best-fitting coils to create a feasible solution (heuristic). We detail this solution approach in Section 3. For Steps 1 and 2, we use the models presented in Sections 2.2 and 2.3. We divided the solution approach into three parts, because we were not able to obtain a feasible solution for the test cases with five periods using the integrated model presented in Section 2.1. Splitting the problem in these three parts makes it possible to handle some parts (of Step 1 and 2) exact, while solving the last step (relocation of coils) with a heuristic. The models used for Step 1 and 2 are easy to solve and fit therefore well to the matheuristic solution approach.

1.1 Related work

The problem we consider herein is related to the literature on intrawarehouse relocation as well as inventory models in lateral transshipment.

The relocation of coils is the main topic of our work. Therefore, we first provide an overview of papers that deal with this topic. Zäpfel and Wasner [9] consider optimal scheduling at a single coil warehouse to ensure the proper handling of coils. They treat the actual sequencing, scheduling, and routing of the movement of goods. For this purpose they formulate the problem as a generalization of the classical job-shop sequencing problem and solve it by means of a local search algorithm. Xie et al. [8] extend their work and study a multicrane scheduling problem for coils that are mainly considered in single warehouses of steel companies. In order to properly treat the problem they consider the internal relocations including necessary moves and solved the problem with a greedy heuristic algorithm. Tang et al. [6] consider two different types of steel products for their shuffling problem, namely, coils and steel plates, and they treat these two product types separately. In their work they include a mixed integer linear program and they solve the problem using a tabu search metaheuristic.

In addition, we take a closer look at papers that deal with the reshuffling problem in more detail. Pazour and Carlo [5] consider a general warehouse reshuffling (or the relocation of items) problem as part of their work and they solve it with several heuristics (simple heuristic reshuffling the closest load, general reshuffling heuristic, and a simulated-annealing-based heuristic; the general reshuffling heuristic performs best for medium and large instances). They provide a detailed literature overview on reshuffling problems. Our work considers the multilocation warehouse problem, which is part of the inventory models associated with lateral transshipment. Archibald et al. [1] consider a two-depot inventory problem with stock transfer. The transfer is triggered by demand, which cannot be fulfilled with the given stock of a depot, and therefore a transfer from another warehouse or an emergency order has to be made. A more detailed literature review of the multilocation warehouse problem can be found in Paterson et al. [4].

Comparing this work to intrawarehouse relocations treated in the literature differs mainly by the consideration of different types of storage slots (i.e., different dimensions b and r), ensuring that the capacity of each warehouse is not exceeded, and that relocations between different warehouses are minimized.

The main difference between the multilocation warehouse problems considered in related literature and the current work is that a transshipment is forced by products that are needed from a different warehouse. Our model is supposed to store coils as long as possible in the same warehouse until they are dispatched or until too many coils of the same type cannot be stored in the same warehouse anymore. Only if a lack of capacity or the necessity for a relocation occurs a relocation must be made. Therefore, we consider capacitated warehouses in which coils can be stored within given dimensions b and r.

Other related topics are the loading, unloading, and pre-marshalling of blocks (e.g., containers and steel plates). The problem is also known as the block relocation problem (BRP). Lehnfeld and Knust [3] present a detailed overview of related work on these topics. In their work they focus on problems where the storage is organized in stacks and therefore the items can be put on top of each other. The items are defined to be cuboids, e.g., containers, wooden plates, or steel plates. In addition, they mention that problems dealing with the storage of round items (i.e., coils) require a different layout and stacking conditions and exclude them from their survey. Any problem that relates to to multi-warehouse relocation problems is not part of their survey. All mentioned papers consider at least a single yard, a single warehouse, a single container ship, or a single tram/bus depot.

Zehendner et al. [10] investigate the online container relocation problem, which is part of the BRP. This problem handles containers which have to be retrieved from a single bay in a container terminal by minimizing the number of relocations. The worst case and average performance of a leveling heuristic (relocates containers to the lowest empty position) is analyzed in more detail.

Tricoire et al. [7] present new methods for the BRP. An unrestricted BRP is treated that yields more opportunities for optimization and they investigate very large instances for a single block. Several fast heuristics are integrated into a new metaheuristic construction framework. In addition, various factors influencing branch-and-bound algorithms for the BRP are investigated by the authors.

1.2 Problem description

The contribution of our paper is twofold. First, methodologically, we create a mixed integer program that handles real-world aspects and develop a fast matheuristic approach that can solve very large real world test problems and improve real world relocations up to 11% for larger test cases (considering a time frame of at least 30 days; see Table 11). A heuristic solution approach is necessary due to long computation times solving the presented model to optimality (see Table 10). We consider different warehouses with different capacities for specific coil slots. In addition, the interaction (via the relocation of coils) between the warehouses is considered. We additionally consider different kinds of slots (with dimensions b and r) and the possibility to store coils in slots with larger dimensions. Second, to account for managerial impact, we evaluate the value of information with respect to capacity or dispatch information and its effect on subsequent periods.

Before presenting the model formulation in more detail, we provide a brief verbal description of the problem. Our solution approaches minimize the number of coil relocations between warehouses, consider different slots (dimensions b and r), and include different slot sizes for specific coil types. Each coil is distributed from a production plant to a specific warehouse. In this warehouse, a coil can be stored within its dimensions, b and r, or in a bigger slot. If a coil is moved to another warehouse within the considered time horizon, it is denoted as relocation. Also, if a coil is distributed from another warehouse it first has to be relocated. The distribution of a coil must be handled within a distribution time window if this time window is entirely outside the planning horizon, the coil may not be sent. If the planning horizon partially overlay this time window, the coil can be distributed (optionally). If the time window is entirely within the planning horizon, the coil must be distributed. The capacity of each warehouse and the maximum number of dispatches and relocations must be met. Mandatory relocations are considered, e.g., if a coil is sent from another warehouse. Some coils may not be relocated after being moved to their distribution warehouse.

In our experiments, we test different scenarios with respect to the available degree of information (information about the production and distribution). In some cases, no information about the production and distribution during the planning horizon is available. In others, the information for a certain number of days, or even for the entire planning horizon, is available.

The remainder of the paper is organized as follows: Section 2 provides an overview of the integrated model (Section 2.1), the subproblem in Section 2.2 used for Step 1 (Figure 2), and the subproblem (Section 2.3) used for Step 2 (Figure 2). Section 3 provides the detailed solution approach. Section 4 details the solutions of the comparison between the integrated model and the matheuristic solution approach, the real world test cases, the study on the value of information, and the comparison between a day-by-day approach and the matheuristic solution approach. Section 5 gives some conclusions and an outlook on future research on this topic.

2 PROBLEM FORMULATION

First, we present a mixed-integer linear programming formulation for the integrated problem of real world coil relocation and distribution in Section 2.1. Second, we present two subproblem formulations in Sections 2.2 of our matheuristic (Step 1; see Figure 2) and 2.3 (Step 2; see Figure 2) of the integrated problem, which we then use in our matheuristic solution approach described in Section 3. By using the integrated model it is not possible to find optimal solutions, even for test instances with only a five day planning horizon, therefore we develop this matheuristic solution approach. Further details about the computational results can be found in Section 4, Table 10.

2.1 Integrated model

Name	Description
T	Set of time periods
C	Set of coils
$H \subset C$	Set of coils already stored
W	Set of warehouses
S_{cw}	Storage slots needed by coil c in warehouse w
B^c	Coil width c
R^c	Coil diameter c
B	Set of all coil widths
R	Set of all coil diameters
C^{br}	Set of coils with width b and diameter r
A_w^{br}	Number of available storage slots for a coil type with width b and diameter r for each warehouse w
K_c	Set of possible warehouses of coil c
D	Maximum number of coil dispatches each day
M	Maximum capacity of relocations between warehouses each period
$\underline{T_c}, \overline{T_c}$	Earliest and latest dispatch period for coil c
$\underline{T}, \overline{T}$	First and last period in the planning horizon
T_c'	Period when coil c enters first the warehouse, if $T_c' = \underline{T}$ coil c is in stock at the initial time period
W_c'	Starting warehouse of coil c

Table 1: Input data for the integrated coil relocation model.

Name	Description
x_{cw}^t	Indicator if coil c is at warehouse w at the end of the period t
y_{cw}^t	Indicator if coil c is dispatched to a customer from warehouse w in period t
z_{cwv}^t	Indicator if coil c is relocated from warehouse w to warehouse v in period t
$m_{wbr}^{tb'r'}$	Number of coils in period t with dimensions b and r in warehouse w assigned to the available slots with dimensions b' and r' (e.g., 3 coils of the dimensions $r = 3$ and $b = 3$ are stored in warehouse w and period t in slots with the dimensions $r' = 4$ and $b' = 4$)

Table 2: Decision variables for the integrated coil relocation model.

$$\min \sum_{c \in C} \sum_{w \in W} \sum_{v \in W} \sum_{t \in T} z_{cwv}^{t} \tag{1}$$

subject to:

$$x_{cW_c'}^{\underline{T}} = 1 \qquad \forall c \in H \tag{2}$$

$$x_{cw}^{t} + x_{cv}^{t+1} + y_{cv}^{t+1} - 1 \leqslant z_{cwv}^{t+1} \qquad \forall \begin{matrix} c \in C, w \in K_c, \\ v \in K_c \setminus \{w\}: t \geqslant T_c' \end{matrix} \tag{3}$$

$$x_{cw}^{t} + x_{cv}^{t+1} + y_{cv}^{t+1} \geqslant 2 z_{cwv}^{t+1} \qquad \forall \begin{matrix} c \in C, w \in K_c, \\ v \in K_c \setminus \{w\}: t \geqslant T_c' \end{matrix} \tag{4}$$

$$y_{cw}^{T_c'} + x_{cw}^{T_c'} = z_{cW_c'w}^{T_c'} \qquad \forall c \in C, w \in K_c \setminus \{W_c'\} \tag{5}$$

$$\sum_{w \in K_c} \sum_{t \in [\underline{T_c}, \overline{T_c}]} y_{cw}^{t} = 1 \qquad \forall c \in C: \overline{T_c} \leqslant \overline{T} \tag{6}$$

$$\sum_{w \in K_c} \sum_{t \in [\underline{T_c}, \overline{T}]} y_{cw}^{t} \leqslant 1 \qquad \forall \begin{matrix} c \in C: \overline{T_c} > \overline{T} \wedge \\ \underline{T_c} \leqslant \overline{T} \end{matrix} \tag{7}$$

$$\sum_{w \in K_c} \sum_{t \in T} y_{cw}^{t} = 0 \qquad \forall \begin{matrix} c \in C: \overline{T_c} > \overline{T} \wedge \\ \underline{T_c} > \overline{T} \end{matrix} \tag{8}$$

$$\sum_{w \in K_c} y_{cw}^{T_c'} + \sum_{w \in K_c} x_{cw}^{T_c'} = 1 \qquad \forall c \in C: T_c' > \underline{T} \tag{9}$$

$$\sum_{w \in K_c} x_{cw}^{t+1} + \sum_{w \in K_c} y_{cw}^{t+1} = \sum_{w \in K_c} x_{cw}^{t} \qquad \forall c \in C, T_c' \leqslant t < \overline{T} \tag{10}$$

$$\sum_{b'=b}^{B} \sum_{r'=r}^{R} m_{wbr}^{tb'r'} = \sum_{c \in C^{br}} S_{cw} x_{cw}^{t} \qquad \forall \begin{matrix} w \in W, t > \underline{T}, \\ b \in B, r \in R \end{matrix} \tag{11}$$

$$\sum_{b'=1}^{b} \sum_{r'=1}^{r} m_{wb'r'}^{tbr} \leqslant A_w^{br} \qquad \forall \begin{matrix} w \in W, t > \underline{T}, \\ b \in B, r \in R \end{matrix} \tag{12}$$

$$\sum_{c \in C} \sum_{w \in K_c} \sum_{v \in K_c \setminus \{w\}} z_{cwv}^{t} \leqslant M \qquad \forall t \in T \tag{13}$$

$$\sum_{c \in C} \sum_{w \in K_c} y_{cw}^{t} \leqslant D \qquad \forall t \in T \tag{14}$$

$$x_{cw}^{t} \in \{0,1\} \qquad \forall \begin{matrix} c \in C, w \in K_c, \\ t \in [T_c', \overline{T_c}]: t < \overline{T} \end{matrix} \tag{15}$$

$$y_{cw}^{t} \in \{0,1\} \qquad \forall c \in C, w \in K_c, t \in [\underline{T_c}, \overline{T_c}] \tag{16}$$

$$z_{cwv}^{t} \in \{0,1\} \qquad \forall \begin{matrix} c \in C, w \in K_c, v \in K_c, \\ t \in [T_c', \overline{T_c}]: v \neq w \wedge t < \overline{T} \end{matrix} \tag{17}$$

$$m_{wbr}^{tb'r'} \in \mathbb{N} \qquad \forall \begin{matrix} w \in W, t > \underline{T}, b \in B, r \in R, \\ b' \in B, r' \in R: b' \leqslant b \wedge r' \leqslant r \end{matrix} \tag{18}$$

The input data for the integrated model is shown in Table 1, and the decision variables in Table 2.

The objective (1) is to minimize the total number of relocations. First, at the beginning of the time horizon, all coils are fixed to their initial warehouse if they are stored; this holds only for coils that are in stock at $\underline{T}$ (Constraints (2)). Constraints (3) and (4) ensure that a relocation has to be made if a coil is stored in or dispatched from another warehouse. If coil c is stored in warehouse w in period t and moves to another warehouse v in period $t+1$ or is dispatched from warehouse v in period $t+1$, a relocation is required (Constraints (3)). If we force a relocation of coil c (mandatory relocation) in period $t+1$ from warehouse w to warehouse v, we have to ensure that the coil is stored or dispatched in period $t+1$ (Constraints (4)). If a coil is stored in or dispatched from another warehouse during the period the coil enters first, a relocation must be considered (Constraints (5)). If the time window is entirely within the planning horizon, it must be dispatched (Constraints (6)). If the time windows exceeds the end of the planning horizon, the coil can optionally be dispatched (Constraints (7)). If the time window is entirely outside the planning horizon, the coil cannot be dispatched (Constraints (8)). Constraints (9) fulfill the requirement that a coil has to be stored or dispatched if the coil enters the first time, except for coils that are already stored at $\underline{T}$. If a coil is stored in a certain time period, it must be stored in or dispatched from a relevant warehouse (Constraints (10)). Constraints (11) and (12) ensure that the maximum storage available for each storage dimension and warehouse is met. Each stored coil c in period t and warehouse w (with the number of storage slots needed) must be assigned to storage dimensions b' and r' (Constraints (11)). The maximum capacity of each storage dimensions b and r for warehouse w must be met (Constraints (12)). The maximum capacity for relocations has to be considered (Constraints (13)) as well as the maximum number of coil dispatches (Constraints (14)). Constraints (15) through (17) define the binary decision variables. Constraints (18) define the integral decision variables.

The presented model (especially Constraints 11 and 12) is a variant of the generalized assignment problem (GAP). According to Ceselli and Righini [2], the GAP is NP-hard, and even determining a feasible solution is NP-complete.

Name	Description
I, J	Set of items J and knapsack I
C_{ij}	Costs assigning item j to knapsack i
R_{ij}	Weight of item j if assigned to knapsack i
B_i	Capacity of knapsack i
q_i	Binary variable equals 1 if item j is assigned to knapsack i

Table 3: Input data and decision variables of the GAP.

$$\min \sum_{i\in I}\sum_{j\in J} C_{ij}q_{ij} \tag{19}$$

subject to:

$$\sum_{j\in J} R_{ij}q_{ij} \leqslant B_i \qquad \forall i \in I \tag{20}$$

$$\sum_{i\in I} q_{ij} = 1 \qquad \forall j \in J \tag{21}$$

$$q_{ij} \in \{0,1\} \qquad \forall i \in I, j \in J \tag{22}$$

The input data and the decision variables of the GAP are shown in Table 3. The GAP consists of the Objective (19), and the Constraints (20) to (22). Constraints (12) of our model relate to Constraints (20) of the GAP (ensuring that the maximum capacity restriction is met), while Constraints (11) of our model correspond to Constraints (21) of the GAP (each item/coil has to be assigned to a knapsack/storage slot). Because of the different objective functions (our model considers only the relocations between the warehouses; the GAP minimizes the total assignment costs) of the presented model compared to the GAP we conclude that our problem is at least NP-complete.

The following model (Section 2.2) is part of the solution approach described in Section 3. It is used for Step 1 of the solution approach (see Figure 2) to determine the minimum total number of relocations for a specific warehouse and period and the coils dispatched for a specific warehouse and period. The goal here is to minimize the number of relocations because the model should primarily force coils to be dispatched (if possible) and not relocated (only if it is necessary). An overview of how it is used in the solution approach is given in Figure 3. Therefore, we use the model described in Section 2.2, to determine the coils to dispatch and forward this information to adjust the information about the actual storage position of each coil and warehouse (needed for Step 2). Only the information on the dispatched coils y_{br} is forwarded. Information of the number of relocated coils z_{br} and the number of coils assigned to specific dimension $m_{br}^{b'r'}$ is needed to provide correct information on the dispatched coils y_{br}. First, all coils that are determined by the model are dispatched. Second, the residual dispatch capacity is used to send additional coils to avoid relocations (see Section 3).

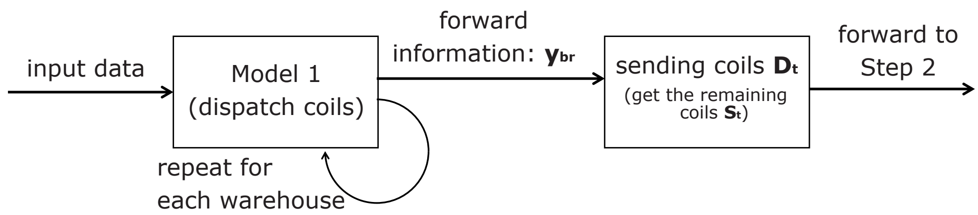

Figure 3: An overview of Step 1 (see Figure 2)

2.2 Step 1: Model to determine the minimum number of relocations and dispatches of coils

Input	Description
S_{br}	Storage needed for all coils of dimensions b and r
A^{br}	Number of available storage slots for a coil type with width b and diameter r
T_{br}	Number of coils available to dispatch for a coil type with width b and diameter r (not all coils are available because of their dispatch time windows; already dispatched coils are excluded; we go through all coils that are available to be dispatched and count their amount for dimensions b and r)
B	Set of all coil widths
R	Set of all coil diameters
O_1, O_2	Constants to balance the parts of the objective function

Table 4: Input data for the model to obtain the minimum number of relocations and dispatches of coils for a specific period and warehouse.

Decision variable	Description
z_{br}	Number of coils to relocate with dimensions b and r
y_{br}	Number of coils dispatched with dimensions b and r
$m_{br}^{b'r'}$	Number of assigned coils with dimensions b and r to the available slot of dimensions b' and r'

Table 5: Decision variables for the model to get the minimum number of relocations and dispatches of coils for a specific period and warehouse.

$$\min \sum_{b \in B} \sum_{\substack{r \in R: \\ S_{br} > 0}} (O_1 z_{br} + O_2 y_{br}) \tag{23}$$

subject to:

$$\sum_{\substack{b' \in B: \\ b' \geqslant b}} \sum_{\substack{r' \in R: r' \geqslant r \wedge \\ A_{b'r'} > 0}} m_{br}^{b'r'} = S_{br} - z_{br} - y_{br} \qquad \forall_{S_{br} > 0 \wedge T_{br} > 0}^{b \in B, r \in R:} \tag{24}$$

$$\sum_{\substack{b' \in B: \\ b' \geqslant b}} \sum_{\substack{r' \in R: r' \geqslant r \wedge \\ A_{b'r'} > 0}} m_{br}^{b'r'} = S_{br} - z_{br} \qquad \forall_{S_{br} > 0 \wedge T_{br} = 0}^{b \in B, r \in R:} \tag{25}$$

$$\sum_{\substack{b' \in B: \\ b' \leqslant b}} \sum_{\substack{r' \in R: r' \leqslant r \wedge \\ S_{b'r'} > 0}} m_{b'r'}^{br} \leqslant A_{br} \qquad \forall b \in B, r \in R: A_{br} > 0 \tag{26}$$

$$y_{br} \leqslant T_{br} \qquad \forall b \in B, r \in R: T_{br} > 0 \tag{27}$$

$$m_{br}^{b'r'} \in \mathbb{N} \qquad \forall_{b' \geqslant b \wedge r' \geqslant r \wedge S_{br} > 0 \wedge A_{b'r'} > 0}^{b \in B, r \in R, b' \in B, r' \in R:} \tag{28}$$

$$z_{br} \in \mathbb{N} \qquad \forall b \in B, r \in R: S_{br} > 0 \tag{29}$$

$$y_{br} \in \mathbb{N} \qquad \forall b \in B, r \in R: T_{br} > 0 \tag{30}$$

Table 4 provides input data for the model to obtain the minimum number of relocations and dispatches of coils for a specific period and warehouse, and Table 5 provides the decision variables. The decision variables z_{br} (number of relocated coils) and $m_{br}^{b'r'}$ (number of assigned coils to specific dimension) are needed to provide correct information of the dispatched coils y_{br} and are not forwarded.

The objective function (23) consists of two parts. The first aims to minimize the total number of relocations (O_1 is weighted highest to ensure this). The second aims to minimize the number of coils dispatched and is weighted less (with O_2) than the first part (O_1) to ensure that coils are only dispatched if a relocation can be prevented. The models purpose is to dispatch coils and therefore prevent relocations (see Section 3).

Constraints (24) and (25) ensure that the total number of stored coils in dimension b and r are stored in the warehouse, get relocated or dispatched (for Constraints (24)). The maximum available slots for each dimension guarantee that no more coils are stored than there is storage available for a specific dimension b and r (Constraints (26)). The maximum available coils to dispatch for each dimension b and r may not be exceeded (Constraints (27)). Constraints (28) through (30) define the integral decision variables.

The following model (Section 2.3) is part of the solution approach described in Section 3. It is used for Step 2 of the solution approach (see Figure 2) to determine the minimum total number of relocations for a specific warehouse and period. It also finds the free slots for a specific coil type in each warehouse and period, and the number of relocations of coil types b and r for a warehouse and period. A detailed information of how it is used in the solution approach is given in Figure 4. We use the model described in Section 2.3 to determine the number of relocations needed for dimensions b and r and period t (z_{br}). This information of z_{br} is saved in P_t and passed on to Step 3. Additionally, we use the information of assigned coils to storage slots ($m_{br}^{b'r'}$) and the number of available storage locations (A^{br}) to determine the unused storage slots. This information is saved in U_t and handed on to Step 3.

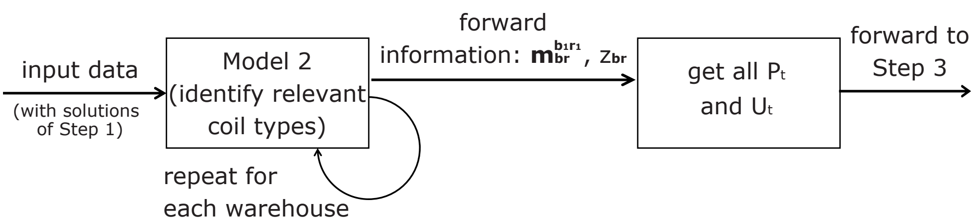

Figure 4: An overview of Step 2 (see Figure 2).

2.3 Step 2: Model to determine the minimum number of relocations for a specific warehouse and period

Name	Description
S_{br}	Storage needed of all coils with dimensions b and r
A^{br}	Number of available storage for a coil type with width b and diameter r
B	Set of all coil widths
R	Set of all coil diameters
O_1, O_2, O_3	Constants to balance the parts of the objective function

Table 6: Input data for the minimum relocation model of a specific time period and warehouse.

Name	Description
z_{br}	Number of coils to relocate with dimensions b and r
$m_{br}^{b'r'}$	Number of assigned coils with dimensions b and r to the available slot of dimensions b' and r'

Table 7: Decision variables for the minimum relocation model of a specific time period and warehouse.

$$\min \sum_{b\in B} \sum_{\substack{r\in R:\\ S_{br}>0}} (O_1 z_{br} + O_2 z_{br} br) + \sum_{b\in B} \sum_{\substack{r\in R:\\ S_{br}>0}} \sum_{\substack{b'\in B:\\ b'\geqslant b}} \sum_{\substack{r'\in R:\\ r'\geqslant r\wedge\\ A_{b'r'}>0}} O_3 m_{br}^{b'r'} b'r' \tag{31}$$

subject to:

$$\sum_{\substack{b'\in B:\\ b'\geqslant b}} \sum_{\substack{r'\in R: r'\geqslant r\wedge\\ A_{b'r'}>0}} m_{br}^{b'r'} = S_{br} - z_{br} \qquad \forall b \in B, r \in R: S_{br} > 0 \tag{32}$$

$$\sum_{\substack{b'\in B:\\ b'\leqslant b}} \sum_{\substack{r'\in R: r'\leqslant r\wedge\\ S_{b'r'}>0}} m_{b'r'}^{br} \leqslant A_{br} \qquad \forall b \in B, r \in R: A_{br} > 0 \tag{33}$$

$$m_{br}^{b'r'} \in \mathbb{N} \qquad \forall_{b'\geqslant b \wedge r'\geqslant r \wedge S_{br}>0 \wedge A_{b'r'}>0}^{b\in B, r\in R, b'\in B, r'\in R:} \tag{34}$$

$$z_{br} \in \mathbb{N} \qquad \forall b \in B, r \in R: S_{br} > 0 \tag{35}$$

Table 6 provides the input data for the model to obtain the minimum number of relocations for a specific warehouse and period, and Table 7 provides the decision variables.

The objective function (31) consists of three parts. The first aims to minimize the total number of relocations (O_1 is weighted highest to ensure this). The second aims to relocate the smallest possible coil dimensions b and r (O_2 is weighted less than O_1). The third ensures that coils are assigned to the smallest possible storage slot dimensions b' and r' (e.g., for dimension $b = 4$ and $r = 4$ it is preferred to store them in $b' = 4$ and $r' = 4$, rather than in $b' = 5$ and $r' = 5$). The third part makes it easier to find the best-fitting warehouse for the relocation process (a coil can be stored for several periods without additional relocation, see Section 3).

Constraints (32) ensure that all stored coils of dimensions b and r are stored in the warehouse or get relocated (which affects the total number of relocations). Constraints (33) guarantee that no more coils are stored than storage is available for a specific dimension b and r. Constraints (34) and (35) define integral decision variables.

3 SOLUTION APPROACH

Notation	Description
T	All considered periods
m	Number of periods for which something is known about future
M	Maximum capacity of relocations between warehouses each period
D	Maximum dispatch capacity
E	Set of coils preventing a relocation due to be dispatched
A_t	Coils available to be dispatched in period t
R_t	Set of relocated coils in period t
D_t	Set of dispatched coils in period t
S_t	Set of stored coils in period t
P_t	Set of minimum number of potential relocation slots (with dimensions b and r) in period t for a specific warehouse
U_t	Set of unused storage

Table 8: Notation for the used algorithms.

Figure 2 provides an overview of the basic elements of the solution approach, and Algorithm 1 provides an outline of the solution approach. Table 8 explains the notation. Step 1 and 2 of Figure 2 are implemented using an exact model (Lines 6 and 7 of Algorithm 1). A more detailed description of Step 1 is shown in Algorithm 2. Step 3 of Figure 2 is implemented as a greedy heuristic that relocates a coil with big dimensions to a free warehouse with the most capacity left (Line 8 in Algorithm 1). A more detailed description of Step 3 is given in Algorithm 3.

First, we get the storage information of all coils for all periods without any dispatch information (the warehouse that each coil enters is used for each period; Lines 3 and 4 of Algorithm 1). This includes only the periods we know in advance (t to $t+m$). Afterward, the dispatch information for each coil is used to determine which coils are dispatched within periods t to $t+m$ (Line 6). For a detailed description see Algorithm 2. After determining the set of coils to be dispatched, we obtain the minimum number of coils relocated for each slot, warehouse, and period t' (Line 7). Therefore, we use the model of the subproblem described in Section 2.3 (Step 2 in Figure 2; see Figure 4 for more details) and solve it to optimality. The objective function (31) of the subproblem described in Section 2.3 is essential to obtain the minimum number of relocations for a given warehouse and period t. The smallest possible slots are needed to identify the best-fitting warehouse for each coil, especially to use coils with larger dimensions b and r (which replace the coils with smaller dimensions and keep the problem feasible) to relocate if no suitable coil is found. This situation can be caused when coils are not allowed to be relocated. We also forward information about unused storage, which is the third part of the objective function (31) of the subproblem described in Section 2.3. This information is needed to find the best-fitting warehouse for a coil.

Step 1 is shown as a function in Algorithm 2 and is needed for the dispatch process. The information about whether a coil is dispatched within a period is determined by two cases. First, if enough dispatch capacity D is available for all coils that are available to dispatch on a given day (dispatched coils are excluded), then all these coils are dispatched (Line 3 and 4). Second, if there is not sufficient capacity (the number of available coils to dispatch is greater than the dispatch capacity D ; Line 5 to 12), it is necessary to find the coils that should be dispatched first. Therefore, we use the subproblem defined in Section 2.2 (Step 1 in Figure 2; see Figure 3 for more details) and solve it to

Algorithm 1 Overview of the solution approach

1: $R_t \leftarrow \emptyset$, $D_t \leftarrow \emptyset$, $S_t \leftarrow \emptyset$, $P_t \leftarrow \emptyset$, $U_t \leftarrow \emptyset$
2: **for** $t = 1 \ldots |T|$ **do**
3:     **for** $t' = t \ldots t+m$ **do**
4:         $S_{t'} \leftarrow$ CoilsStoredWithoutDispatchInformation(t')
5:     **for** $t' = t \ldots t+m$ **do**
6:         $D_{t'}, S_{t'} \leftarrow$ Step1DispatchCoils(t',$S_{t'}$,$D_{t'}$)
7:         $P_{t'}, U_{t'} \leftarrow$ Step2GettingMinPotentialSlotsToRelocate($S_{t'}$,$P_{t'}$,$U_{t'}$)
8:     $R_t,S_t,P_t,U_t \leftarrow$ Step3RelocateCoils(t,m,R_t,S_t,P_t,U_t)
    return R_t,D_t,S_t

Algorithm 2 Overview of the dispatch process

1: **function** STEP1DISPATCHCOILS(t,S_t,D_t)
2: $E \leftarrow \emptyset$
3: **if** $|A_t| \leqslant D$ **then**
4: D_t, $S_t \leftarrow$ DispatchAllCoils(t,A_t)
5: **else**
6: $E \leftarrow$ IdentifyCoilsPreventRelocationAllWarehouses(t,A_t)
7: **for each** $c \in E$ **do**
8: **if** HighestDimensionsOfResidualCoils(c,E) & $|D_t| <$ D **then**
9: $D_t \leftarrow$ DispatchCoil(t,c)
10: **for each** $c \in A_t$ **do**
11: **if** HighestDimensionsOfResidualCoils & $|D_t| <$ D **then**
12: $D_t \leftarrow$ DispatchCoil(t,c)
13: $S_t \leftarrow$ ActualizeStoredCoils(t,D_t)
return D_t,S_t

optimality for each warehouse in period t (Line 6). After identifying the set of coils preventing relocation, we dispatch all of these coils that are available to send first (Line 7 to 9) and then the residual coils are dispatched (Line 10 to 12) based on the largest dimensions first. As a result, we take into account that coils with smaller dimensions b and r have more storage slots available to them than larger ones. In Line 13, we update the stored coils in period t.

Step 3 is shown as a function in Algorithm 3 (Line 8 in Algorithm 1). This Step is needed to find the best-fitting coil to relocate without restricting the maximum number of relocations allowed (Line 5 and 11). If enough relocation capacity is left in period t (Line 5), we can look for the best-fitting coil in that period (Line 6). If there is not sufficient relocation capacity (Line 8 through 13), we have to find a period (t_n) with enough relocation capacity left; this holds only if no relocation period and coil is found (Line 11). If we find a suitable period (Line 7 or 13) we update all subsequent periods' information (Lines 14 through 15).

To find the best-fitting coil (Lines 6 and 12), we revisit all potential coils for relocation to identify specific dimensions b and r that match the restrictions of warehouse (P_t). First, we try to find coils to relocate that have to be relocated anyway (e.g., a coil that has to be dispatched from another warehouse because it is dispatched together with others). Second, we observe coils with the same or larger dimensions b and r. The second step is only processed if no coil is found in the first step (coils that must be relocated anyway). Then, the possible coils are sorted according to their longest minimum possible stay in a warehouse. The longest minimum possible stay is influenced by the number of known periods of future production data (m) and also by the residual capacity of a warehouse in upcoming periods. We select the coils with the highest values and sort the equal ones according to their average number of potential storage leftover and choose the coil with the longest average available storage time in a certain warehouse. Figure 5 demonstrates how to calculate the number of potential storage left. Potential storage left represents the amount of storage that would be leftover

Algorithm 3 Detailed overview of Step 3 (relocate coils)

1: **function** STEP3RELOCATECOILS(t,m,R_t,S_t,P_t,U_t)
2: **for** $p = 1 \ldots |P_t|$ **do**
3: **repeat**
4: $t' \leftarrow 0$
5: **if** $|R_t| < M$ **then**
6: R_t, $U_t \leftarrow$ FindingBestFittingCoilsToRelocate(m,P_t,U_t)
7: $t' \leftarrow t$
8: **else**
9: **for** $t'' = 1 \ldots t$ **do**
10: $t_n \leftarrow t - t''$
11: **if** $|R_{t_n}| < M$ & NoDayAndCoilFound **then**
12: R_{t_n}, $U_{t_n} \leftarrow$ FindingBestFittingCoilsToRelocate(t_n,m,P_t,U_t)
13: $t' \leftarrow t_n$
14: **for** $t'' = t' \ldots t + m$ **do**
15: R_t,S_t,P_t,$U_t \leftarrow$ GetNewInfoOfRelocatedCoils(t'',R_t,S_t,P_t,U_t)
16: **until** NoMoreRelocationsNeeded(P_t)
return R_t,S_t,P_t,U_t

if the considered coil is stored in a specific warehouse and therefore indicates a degree of capacity utilization.

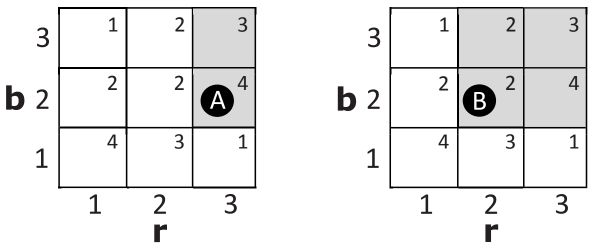

Figure 5: The two parts of the figure show coil dimensions b and r for each slot and the available storage. For example, for dimensions $b = 2$ and $r = 3$ are four storage slots available. Calculating the free storage slots for coil A: Only the same dimensions or larger are considered (the "gray area"). There are six residual places left if the coil is stored in the warehouse; this includes the residual storage of $b = 3, r = 3$ (3) and $b = 2, r = 3$ (4), and storing coil A (1) (in total, $3 + 4 - 1$). For coil B, 10 residual storage slots are left.

4 COMPUTATIONAL EXPERIMENTS

4.1 Data description

For the real world case study, we obtained data from a steel producing company. We were able to get several data sets representing different time periods (up to 87 days), which enabled us to create the necessary input data. The data include a very large amount of coils (with up to 76,488 coils) to consider. Therefore, the test cases only differ with respect to the number of coils, the coil information (e.g., sizes, dispatch dates), the utilization of all warehouses, and the number of considered periods. For each coil, we have access to all relocation, storage, and dispatch data. The company also provided the capacity of the different warehouses (we consider nine different warehouses) to ensure a reasonable comparison between the real world relocations and the relocations of the solution approach described in Section 3.

The considered coils, and therefore the real world relocations, can differ between the considered time horizons. The relocations are dependent on the utilization of the different warehouses. To account for these different scenarios, we created 15 test cases to determine how well our solution approach works. Comparison between the solutions generated by the solution approach and the real world scenario is only possible because we use the same dispatch information (therefore the same dispatch date) as occurred in the real world. Several relocations needed to be made, and these could differ if we did not use the same dispatch

Test case	Periods	Number of Coils				Average utilization
				Incoming		
		Initial	Total	Total	Average	
1	5	13,486	15,716	2,230	557.5	76.7%
2	5	13,274	16,067	2,793	698.3	79.7%
3	5	15,468	18,570	3,102	775.5	89.2%
4	15	13,463	20,248	6,785	484.6	74.2%
5	15	14,382	25,405	11,023	787.4	86.6%
6	15	14,542	24,711	10,169	726.4	82.2%
7	30	13,193	34,795	21,602	744.9	81.9%
8	30	13,859	34,179	20,320	700.7	78.0%
9	30	13,355	30,709	17,354	598.4	74.7%
10	45	13,121	46,323	33,202	754.6	82.7%
11	45	13,993	47,030	33,037	750.8	83.2%
12	45	14,711	45,905	31,194	709.0	80.3%
13	60	13,106	57,854	44,748	758.4	83.3%
14	60	13,591	57,016	43,425	736.0	81.1%
15	87	13,097	76,488	63,391	737.1	81.2%

Table 9: Overview of the main part of the data experiments (data for the results of Table 14). Each test case has a different number of considered periods (ranging from 5 to 87 periods). The number of considered coils is divided into initial number of coils (initial stock), the number of all coils considered, and the coils that are incoming over all periods (total coils reduced by initial coils, and average coils incoming per day). In addition, the average utilization of all warehouses is shown in the last column. This value is calculated over all warehouses over all periods. Therefore, there are warehouses with a utilization of nearly 100%, whereas other warehouses have a lower utilization.

information (e.g., if coils were not dispatched in the same period, the necessity of relocating the coil to another warehouse is not given). Therefore, the dispatch time window is fixed to the dispatch dates of the real world case.

The parameters of objective function (23) are weighted with $O_1 = 10000$ and $O_2 = 0.1$, and the parameters of objective function (31) are weighted with $O_1 = 10000$, $O_2 = 0.1$ and $O_3 = 0.000001$. For all test cases, $B = 15$ and $R = 15$. Therefore, we consider 225 different types of coils. In addition, parameter $M = 500$ (the maximum number of coils that can be handled in one period) and parameter $D = 3000$ (high number, because we have to dispatch all coils due to their real world dispatch dates). If no information is available, $m = 0$ (see Algorithm 1 and data description Table 8), if full information is available $m = T$ (e.g., test case 15 has 87 periods, therefore $m = 87$).

Table 9 shows some detailed information for test cases 1 to 15. We consider six different time periods: three instances with five periods (test cases 1 to 3), three instances with 15 periods (test cases 4 to 6), three instances with 30 periods (test cases 7 to 9), three instances with 45 periods (test cases 10 to 12), two instances with 60 periods (test cases 13 and 14), and one instance with 87 periods (test case 15). In addition, we show the total number of initial, total, and incoming coils considered. The incoming coils are the ones produced during the considered time periods. The initial ones are the ones already stored at the beginning of the time horizon. E.g., for test case 1, at the beginning of the time horizon we have 13,486 coils stored and each period on average 557.5 coils arrive from the production sites (in total 2,230 coils). Therefore, in total, we consider 15,716 coils for this test case. We also provide the average utilization of all considered warehouses for each test case, e.g., for test case 1 the average utilization of the warehouses is 76.7%.

The computational experiments section is divided into four parts:

- Comparison between the integrated model and the matheuristic solution approach (Section 4.2.1)
- Comparison between the matheuristic solution approach (full information an no information available) and the real world situation (Section 4.2.2)
- The value of information for test case 15, with 87 days considered (Section 4.2.3)
- Computational results due to resetting the time windows (earliest and latest dispatch period) of coils (Section 4.2.4)
- Comparison between a day-by-day exact approach and the matheuristic solution approach with no information available (Section 4.2.5)

4.2 Results

For computational testing of the matheuristic solution approach, we used IBM ILOG CPLEX Optimization Studio (Version 12.7.1) and Microsoft Visual C# 2013 for Windows x86-64. For the first part (the

Instance	Periods	# reloc.		Gap			Runtime [sec.]	
		M	IM	LB	M to LB	IM to LB	M	IM
S1	2	147	147	147	0.0%	0.0%	0.21	31
S2	2	100	100	100	0.0%	0.0%	0.03	13
S3	2	173	173	173	0.0%	0.0%	0.05	13
S4	2	158	158	158	0.0%	0.0%	0.08	14
S5	2	123	123	123	0.0%	0.0%	0.08	14
S6	3	360	360+	344	4.4%	4.4%	0.31	259,200
S7	3	462	458*	455	1.5%	0.7%	0.64	224,412
S8	3	403	402	402	0.2%	0.0%	0.39	440
S9	3	318	317	317	0.3%	0.0%	0.17	508
S10	3	445	445+	443	0.4%	0.4%	0.39	259,200
S11	4	563	561	561	0.4%	0.0%	0.84	46,895
S12	4	582	577*	575	1.2%	0.3%	0.64	164,448
S13	4	535	532*	529	1.1%	0.6%	0.42	158,386
S14	4	279	272*	270	3.2%	0.7%	0.49	132,616
S15	4	377	367*	362	4.0%	1.4%	0.49	116,751

*Table 10: The table shows the computational results of a comparison between the matheuristic solution approach (M) and the integrated model solved by CPLEX (IM). The column "# reloc." represents the total number of relocations needed. In addition, we provide the lower bound (LB) generated by CPLEX to compare the results. The column "Gap" shows all gaps to the LB. The last column, "Runtime [sec.]", is the runtime of the matheuristic and the integrated model to find the provided solutions. We obtained two solutions of the integrated model (instance S6 and S10) where the runtime of 3 days is exceeded (shown as + in column "# reloc" of IM). In addition, for five test instances we are not able to find an optimal solution due to a lack of memory (marked as * in column "# reloc" of IM)*

comparison between the integrated model and the matheuristic solution approach, see Table 10) we use a high-performance computer with one node using two Intel Xeon E5-2650v2 (2.6GHz, 8 cores) and 256 GB of memory, with IBM ILOG CPLEX Optimization Studio (Version 12.7.1). All other calculations were performed using an Intel i7-4810MQ Processor (2.80 GHz, 4 cores) and 32 GB of memory.

4.2.1 Comparison of the integrated model and the matheuristic solution approach

The results of Table 10 show the comparison between the matheuristic solution approach (with full information) and the integrated model. Test cases with two periods (S1 to S5) can be solved to optimality for both, the matheuristic solution approach and the integrated model solved by CPLEX. The test cases with three periods (S6 to S10) can only be solved to optimality for test cases S8 and S9 with the integrated model solved by CPLEX and the other test cases run out of memory (S7) or the time limit of three days was exceeded (test cases S6 and S10). The results of the matheuristic solution approach are slightly worse than the ones calculated by CPLEX. For test case S6 the gap to the lower bound is 4.4% and is therefore the same as for the one calculated by the integrated model solved by CPLEX. The other test cases vary between 0.2% and 1.5% gap to the lower bound. For test cases S11 to S15, only for one test case (S11) the optimal solution was found. The gap for the matheuristic solution approach is 0.4%. The other test cases (S11 to S15) run out of memory for the integrated model solved by CPLEX. The gaps vary between 0.3% to 1.4% for the integrated model and 0.4% to 4.0% for the matheuristic solution.

The results show that the matheuristic solution approach seems to work well for small instances with fast computation times compared to the integrated model solved by CPLEX. In addition, we tested cases with more than four days planning horizon, but we cannot find a feasible solution within a reasonable time.

4.2.2 Comparison of the matheuristic solution approach with full and no information

Table 11 shows the results of all 15 test cases. The initial coils in stock and the incoming coils vary between the different scenarios as the result of different time horizons considered and the utilization of the different warehouses over time. The number of relocations made are divided into three main categories: the real world case, the results with complete information, and the results with no information.

To compare the results in more detail, we consider the differences between the real world solutions and those generated by solution approach. This part is divided into the absolute (real world, full information, and no information) and relative (potential reduction if complete or no information is available) values. The results show it is possible to reduce the real world relocations even if no information is available about future production and dispatch dates. If more

Test case	Pe-ri-ods	Number of relocations						Differences			
		Real world		Full information		No information		Real world		Potential reduction	
		Total	Average	Total	Average	Total	Average	Full info	No info	Full info	No info
1	5	544	136.0	369	92.3	369	92.3	175	175	32.2%	32.2%
2	5	1,050	262.5	876	219.0	877	219.3	174	173	16.6%	16.5%
3	5	1,128	282.0	931	232.8	934	233.5	197	194	17.5%	17.2%
4	15	3,216	229.7	2,722	194.4	2,725	194.6	494	491	15.4%	15.3%
5	15	3,801	271.5	3,458	247.0	3,466	247.6	343	335	9.0%	8.8%
6	15	3,093	220.9	2,716	194.0	2,731	195.1	377	362	12.2%	11.7%
7	30	7,045	242.9	6,331	218.3	6,409	221.0	714	636	10.1%	9.0%
8	30	6,603	227.7	5,896	203.3	5,958	205.4	707	645	10.7%	9.8%
9	30	8,892	306.6	8,152	281.1	8,200	282.8	740	692	8.3%	7.8%
10	45	11,184	254.2	10,151	230.7	10,338	235.0	1,033	846	9.2%	7.6%
11	45	10,803	245.5	9,634	219.0	9,794	222.6	1,169	1,009	10.8%	9.3%
12	45	9,858	224.0	8,775	199.4	8,891	202.1	1,083	967	11.0%	9.8%
13	60	14,561	246.8	13,136	222.6	13,506	228.9	1,425	1,055	9.8%	7.2%
14	60	14,224	241.1	12,789	216.8	13,079	221.7	1,435	1,145	10.1%	8.0%
15	87	20,681	240.5	18,663	217.0	19,186	223.1	2,018	1,495	9.8%	7.2%

Table 11: Results for 15 different test cases. The table is split into two major parts: number of relocations and differences between the real world case and the optimized number of relocations with the matheuristic solution approach. The number of relocations made are divided into the relocations made in the real world, the relocations made if complete information is available, and the relocations made if no information is available. These columns are divided into total and average number of relocations (the average number of relocations made for the considered time horizon). The differences between full and no information are shown in the differences column. First, we compare the real world relocations with the relocations made with complete and no information. Second, we show the relative relationship of real world relocations with relocations with complete and no information.

information is available the relocations can be reduced, except for test case 1 (369 relocations with complete and no information, difference from real world = 175 [32.2%] relocations). The reason for no reduction due to more information (test case 1) is the considered time horizon. If enough capacity is left and few relocations have to be made (because of few bottlenecks in capacity), the lack of information has no drawback and the result is still favorable.

The more days we consider, the more the impact of information about production and dispatch increases. The impact for small test cases (up to 15 days) is small, whereas if we consider more than 15 days the importance of information increases. For test cases 1 to 3, the differences in the reductions of the real world relocations are 0 (complete information 175, no information 175) for test case 1; 1 (174 complete information, 173 no information) for test case 2, for relative reduction of 0.1% (16.6% complete information, 16.5% no information); and 3, for a relative reduction of 0.3%. For test cases 4 to 6, the relative reductions vary between 0.1% and 0.5% (test case 6: relative reduction with complete information is 12.2%, and the relative reduction with no information is 11.7%). For test cases 7 to 9, these values vary between 0.5% and 1.1%; for test cases 10 to 12, they vary between 1.2% and 1.7%; and test cases 13 to 15, the values vary between 2.0% and 2.5%. The highest reduction of all test cases with at least 30 days considered (test cases 7 to 15), and full information is test case 12 (reduction of 11.0%). The lowest is test case 9 (reduction of 8.3%). For the same test cases with no additional information available, the highest value is for test case 12 (with 9.8%), and the lowest is for test case 13 (with 7.2%).

4.2.3 Value of information

The value of information differs between the considered time horizons. To provide a more detailed overview of the value of information, we consider more granular (varying the days known in advance) results for the largest test case (test case 15, with 87 days considered). For this test case, we consider a total of 76,488 coils, with 13,097 coils in stock in the first period considered. Therefore, an average of 737.1 coils are produced and stored over time. Table 12 and Figure 6 show the impact of known days in advance with information about the production and dispatch dates. The more information that is available, the more the total number of relocations can be reduced. This only holds for a specific amount of days known in advance. If more than 40 days of information is known, no further reductions can be made relative to knowing 40 days of production and dispatch data in advance. In other words, there is a diminishing return on the number of days information is known, with 40 being the optimal number (test case 15).

We consider the real world relocations made for test case 15 (87 days considered) in greater detail. We split the relocations into two parts: mandatory relocations (fixed relocations) and relocations due to a lack of capacity or other undefined reasons (others). The fixed relocations contain 75% (15,489) of all relocations made (20,681; see Table 11). Thus the residual 5,192 relocations (25 percent) are the relocations of coils that can be reduced by the model if possible. These other relocations can differ from the real world relocations if relocations are not necessary due to a lack of capacity or if a better-fitting coil can be relocated.

Days known	# of relocations	Difference to full info	Difference [%]	Difference to real world
0	19,186	523	2.8%	-1,495
1	19,133	470	2.5%	-1,548
2	19,058	395	2.1%	-1,623
5	18,941	278	1.5%	-1,740
10	18,836	173	0.9%	-1,845
15	18,778	115	0.6%	-1,903
20	18,739	76	0.4%	-1,942
30	18,694	31	0.2%	-1,987
40	18,663	0	0.0%	-2,018
87	18,663	0	0.0%	-2,018
Real world	20,681			

Table 12: The table shows a detailed overview of test case 15 (87 days considered). The column "days known" represents the information known in advance (production and dispatch information). For example, "0 days known" represents no information available only on same day as considered, "2 days known" represents the production and dispatch information of the first period and the next two days. Number of relocations ("# of relocations") are the overall relocations required to obtain a feasible solution, "Difference to full info" is the absolute difference to the case of all information being available, and "Difference %" the relative ones. The last column represents the absolute difference to the real world case.

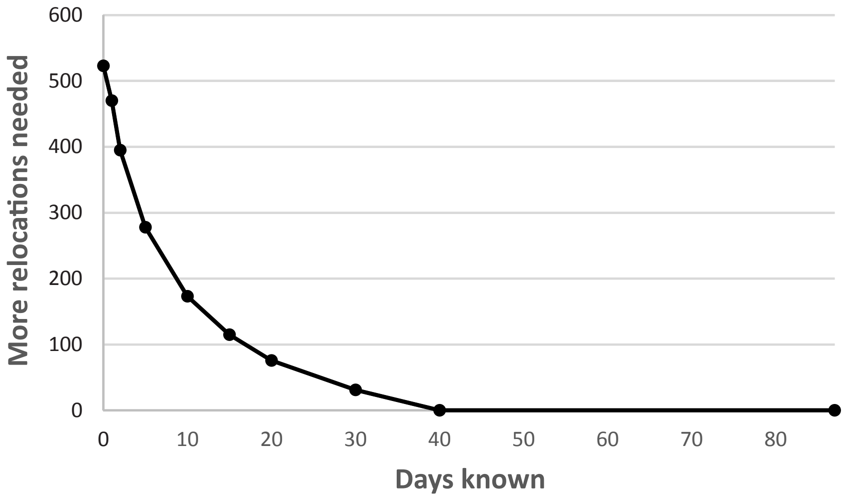

Figure 6: The values in Table 12 represent the slope and show the decrease in additional relocations that are necessary (relative to the case of complete information) as more production and dispatch information becomes available. This is a more detailed graphical overview of test case 15 (87 days considered).

We also consider the solutions from the approach proposed in Section 3 if all or no information is available. Therefore, we consider only the relocations that are not mandatory (the mandatory ones cannot be changed). If complete information is available, we can reduce 39% (2,018 relocations) and therefore need only 3,174 relocations compared with 5,192. If no information is available, we can only reduce 29% (1,495). Therefore, we need 3,697 additional relocations relative to the mandatory ones to ensure a feasible solution.

The relocations saved compared with all relocations made are the same as shown in Table 11. With complete information available, we can reduce 9.8% and with no information we can reduce 7.2%. In total, 523 additional relocations can be saved if complete information is available compared with no information.

4.2.4 Increase of coils' dispatch time windows

In addition, we consider a resetting of the time window of coils (earliest $\underline{T_c}$ and latest $\overline{T_c}$ dispatch period of a coil). Therefore, we adjusted the real world data with respect to the information about the earliest and latest dispatch period. We increase the dispatch date of coils to be one day before and one day after the dispatch date of the original test cases. E.g., if the original dispatch

Test case	Periods	Full information			No information		
		No sending TW	Sending TW	Difference	No sending TW	Sending TW	Difference
1	5	369	314	14.9%	369	314	14.9%
2	5	876	837	4.5%	877	837	4.6%
3	5	931	865	7.1%	934	865	7.4%
4	15	2,722	2,589	4.9%	2,725	2,591	4.9%
5	15	3,458	3,355	3.0%	3,466	3,362	3.0%
6	15	2,716	2,620	3.5%	2,731	2,624	3.9%
7	30	6,331	6,250	1.3%	6,409	6,299	1.7%
8	30	5,896	5,721	3.0%	5,958	5,757	3.4%
9	30	8,152	7,996	1.9%	8,200	8,026	2.1%
10	45	10,151	9,937	2.1%	10,338	10,066	2.6%
11	45	9,634	9,322	3.2%	9,794	9,414	3.9%
12	45	8,775	8,523	2.9%	8,891	8,569	3.6%
13	60	13,136	12,864	2.1%	13,506	13,154	2.6%
14	60	12,789	12,364	3.3%	13,079	12,596	3.7%
15	87	18,663	18,494	0.9%	19,186	18,963	1.2%

Table 13: The table shows how the relocations change if we extend the earliest and latest dispatch period of coils. We provide the solutions of both cases, with full and no information available. The columns "No sending TW" provide the results without this extension. Therefore the values are also given in Table 11 (for both, full and no information available). The columns "Sending TW" show the results generated with the modified input data.

date was day 5 we set T_c to 4 and the $\overline{T_c}$ to 6. The maximum number of coil dispatches each day (M) is set to the maximum coil dispatches of all periods of each test case.

The results of Table 13 show that the solutions of all test cases can be improved by extending the dispatch period. This can be observed for both, the case if we consider no information and full information. In general the improvement with no information is higher than the ones with full information available. The extension of the time windows for test case 15 reduce the relocations needed from 18,633 to 18,494 (reduction of 0.9%) with full information available, and the relocations needed from 19,186 to 18,963 (reduction of 1.2%) with no information available. Test cases with at least 30 days vary between 0.9% to 3.3% reduction gap with full information available, and vary between 1.2% to 3.9% reduction gap for no information available.

4.2.5 Comparison of the matheuristic solution approach and the day-by-day approach

Table 14 shows the comparison between an exact day-by-day approach (solving one day with the starting warehouses fixed to the results of the day before; model Section 2.1; Constraint 13 is not considered to generate feasible solutions), the solution approach (Section 3) with no additional information (so we have the same information available as the exact day-by-day approach), and the real world relocations. We observe that the exact day-by-day approach is always worse than the solution approach with no information (by at least 0.96%). It becomes even worse than the real world relocations if more periods are considered (with up to 20.88% worse for test case 15).

The main advantage of the matheuristic approach is that we can consider how many slots (free space in a warehouse) are available for a given coil (see Figure 5). The model stores coils within warehouses that may not have many slots left and therefore causes a relocation in subsequent periods.

5 CONCLUSIONS AND OUTLOOK

Our results can be used to support practitioners by helping them minimize relocations of steel coils between warehouses. Our developed matheuristic (Section 3) can handle different kinds of information available about production and dispatch and can reveal the value of this information.

The results of the test cases show that the solution approach offers a suitable representation of the real world scenario because we consider both, the mandatory relocations and the relocations necessary due to lack of capacity. The presented solutions offer substantial improvements when complete information is available (maximum 11.0%, minimum 8.3%, average 10.0% for test cases with at least 30 days; see Table 11). However, even if no information about future production and dispatch is available, the number of relocations can be reduced (maximum 9.8%, minimum 7.2%, average 8.4% for test cases with at least 30 days;

Test case	Day-by-day (exact)		No information (solution approach)		# of real world reloc.	Differences day-by-day to no info
	CT [sec.]	# reloc.	CT [sec.]	# reloc.		
1	123.8	386	2.1	369	544	4.61%
2	131.2	899	3.0	877	1,050	2.51%
3	143.5	943	3.9	934	1,128	0.96%
4	430.5	2,799	14.4	2,725	3,216	2.72%
5	694.6	3,809	27.3	3,466	3,801	9.90%
6	470.8	3,073	23.6	2,731	3,093	12.52%
7	952.0	7,251	71.2	6,409	7,045	13.14%
8	847.9	6,861	60.1	5,958	6,603	15.16%
9	850.6	8,761	50.7	8,200	8,892	6.84%
10	1,543.2	11,882	156.9	10,338	11,184	14.94%
11	1,440.6	11,277	135.9	9,794	10,803	15.14%
12	1,449.7	10,332	120.5	8,891	9,858	16.21%
13	2,109.2	15,682	261.0	13,506	14,561	16.11%
14	1,986.6	15,207	229.4	13,079	14,224	16.27%
15	3,119.3	23,192	562.3	19,186	20,681	20.88%

Table 14: The table shows a comparison between the day-by-day approach and the solution approach with no information (see Section 3). Therefore, we run the model (Section 2.1) for each day and consider the warehouse of each coil from the previous day (as starting warehouse). The results of the day-by-day exact approach and for the solution approach with no information are split in two parts: computation time in seconds (CT [sec.]) and number of relocations (# reloc.). The number of relocations of the no information part are similar to the those in Table 11. This table also provides the number of real world relocations (# of real world reloc.) and the differences between the day-by-day and the solution approach with no information.

see Table 11). For all test cases, we are able to improve the number of relocations with and without information available.

The solution approach shows the value of information. The more information (up to a specific number of known days) that is available, the better the solutions (less relocations) are for time horizons with more than five days. For practical applications the costs of more information and the savings due to the reduction of relocations must be taken into account. In addition, our results show that an increase of the earliest and latest dispatch periods can improve the results with no information and full information available. The impact on test cases with no information available is bigger. For test case 15, the reduction of relocations needed is 0.9% (a reduction of 169 coils relocated) if all information is available, and 1.2% (a reduction of 223 coils relocated) if no information is available.

Future work should focus on the integration of internal relocations. Because of the high utilizations of warehouses, internal relocations should get more focus if the solution approach is implemented. Therefore, internal relocations should also be minimized to reduce the internal warehouse costs. An integrated solution approach with internal and external relocations that considers the detailed costs of each internal or external relocation would be worthwhile. The implementation of the problem can be built on the presented matheuristic solution approach. First, the relocations between the warehouses could be minimized for all considered periods. Second, this information can be used to optimize internal relocations. This information can be passed to the first phase (relocations between the warehouses) and can be re-optimized with the additional information available. By repeating these two steps the quality of the overall solution may be improved.

ACKNOWLEDGEMENTS

The financial support of the Austrian Federal Ministry of Science, Research and Economy and the National Foundation for Research, Technology and Development is gratefully acknowledged.

We also thank Michael Wasner, Cornelia Jetzinger (at an early stage), and Biljana Roljic for their valuable input.

The computational results presented herein were achieved, in part, using the Vienna Scientific Cluster (VSC).

We thank both reviewers for their thoughtful comments and efforts improving our manuscript.

REFERENCES

1. T. W. Archibald, S. A. E. Sassen, and L. C. Thomas. An optimal policy for a two depot inventory problem with stock transfer. *Management Science*, 43(2):173–183, 1997.
2. Alberto Ceselli and Giovanni Righini. A branch-and-price algorithm for the multilevel generalized assignment problem. *Operations Research*, 54(6):1172–1184, 2006.
3. Jana Lehnfeld and Sigrid Knust. Loading, unloading and premarshalling of stacks in storage areas: Survey and classification. *European Journal of Operational Research*, 239(2):297–312, 2014.
4. Colin Paterson, Gudrun Kiesmüller, Ruud Teunter, and Kevin Glazebrook. Inventory models with lateral transshipments: A review. *European Journal of Operational Research*, 210:125–136, 2011.
5. Jennifer A. Pazour and Hèctor J. Carlo. Warehouse reshuffling: Insights and optimization. *Transportation Research Part E*, 73:207–226, 2015.
6. Lixin Tang, Ren Zhao, and Liu Jiyin. Models and Algorithms for Shuffling Problems in Steel Plants. *Naval Research Logistics*, 59(7):502–524, 2012.
7. Fabien Tricoire, Judith Scagnetti, and Andreas Beham. New insights on the block relocation problem. *Computers and Operations Research*, 89:127–139, 2018.
8. Xie Xie, Yongyue Zheng, and Yanping Li. Multi-crane scheduling in steel coil warehouse. *Expert Systems with Applications*, 41(6):2874–2885, 2014.
9. Günther Zäpfel and Michael Wasner. Warehouse sequencing in the steel supply chain as a generalized job shop model. *International Journal of Production Economics*, 104(2):482–501, 2006.
10. Elisabeth Zehendner, Dominique Feillet, and Patrick Jaillet. An algorithm with performance guarantee for the online container relocation problem. *European Journal of Operational Research*, 259(1):48–62, 2017.