A Tabu Search Approach to the Truck Scheduling Problem with Multiple Docks and Time Windows

Jan Van Belle[a,*], Paul Valckenaers[a], Greet Vanden Berghe[b], Dirk Cattrysse[a]

[a]*KU Leuven, Department of Mechanical Engineering,*
Celestijnenlaan 300B, 3001 Heverlee, Belgium
[b]*KAHO St.-Lieven, Combinatorial Optimization and Decision Support,*
Gebroeders De Smetstraat 1, 9000 Gent, Belgium

Abstract

While organizing the cross-docking operations, cross-dock managers are confronted with many decision problems. One of these problems is the truck scheduling problem. This article presents a truck scheduling problem that is concerned with both inbound and outbound trucks at multiple dock doors. The objective is to minimize the total travel time and the total tardiness. The truck scheduling problem under study is described in detail and a mathematical model of the problem is provided which can be solved to optimality with a mixed integer programming solver, at the expense of a high computation time. Next, a tabu search approach is presented. Experimental results on new benchmark instances indicate that the proposed tabu search is able to find good quality results in a short time period, thus offering potential for integration in cross-docking decision support systems.

Keywords: Cross-docking, Logistics, Truck scheduling, Tabu search

[*]Corresponding author. Tel.: +32 16 32 25 34.

Email addresses: `jan.vanbelle@mech.kuleuven.be` (Jan Van Belle), `paul.valckenaers@mech.kuleuven.be` (Paul Valckenaers), `greet.vandenberghe@kahosl.be` (Greet Vanden Berghe), `dirk.cattrysse@cib.kuleuven.be` (Dirk Cattrysse)

Preprint submitted to Computers & Industrial Engineering *January 7, 2013*

Research highlights

Inbound and outbound trucks have to be scheduled at the dock doors of a cross-dock. > The objective is the minimization of the total travel time and the total tardiness. > A mixed integer programming model of this truck scheduling problem is formulated. > A tabu search approach to solve the problem is described in detail. > Experiments on new benchmark instances show that good quality solutions are quickly found.

A Tabu Search Approach to the Truck Scheduling Problem with Multiple Docks and Time Windows

Abstract

While organizing the cross-docking operations, cross-dock managers are confronted with many decision problems. One of these problems is the truck scheduling problem. This article presents a truck scheduling problem that is concerned with both inbound and outbound trucks at multiple dock doors. The objective is to minimize the total travel time and the total tardiness. The truck scheduling problem under study is described in detail and a mathematical model of the problem is provided which can be solved to optimality with a mixed integer programming solver, at the expense of a high computation time. Next, a tabu search approach is presented. Experimental results on new benchmark instances indicate that the proposed tabu search is able to find good quality results in a short time period, thus offering potential for integration in cross-docking decision support systems.

Keywords: Cross-docking, Logistics, Truck scheduling, Tabu search

1. Introduction

Cross-docking is a logistics strategy nowadays used by many companies in different industries (e.g. by less-than-truckload (LTL) logistics providers). The rationale of cross-docking is to transfer incoming shipments directly to outgoing vehicles without storing the goods in between. This is different from the approach used in a traditional distribution center, where four major functions of warehousing can be distinguished: receiving, storage, order picking and shipping. Cross-docking eliminates the two most expensive handling operations (storage and order picking) and can be described as *"the process of consolidating freight with the same destination (but coming from several origins), with minimal handling and with little or no storage between unloading and loading of the goods"* (Van Belle et al., 2012). If the shipments are temporally stored, this should be only for a short period of a time. A precise limit is difficult to define, but usually 24 hours is assumed to be the maximum storage time. A cross-dock has multiple loading docks (or dock doors) where trucks can dock to be loaded or unloaded. Incoming trucks are assigned to a 'strip door' where the freight is unloaded. Then the goods are moved to their appropriate 'stack door' and loaded on an outbound truck.

Cross-docking has several advantages: consolidation of shipments, shorter delivery lead times, reduction of costs, etc. However, organizing cross-docking operations is complex and challenging and cross-docking practitioners have to face several decision problems. Van Belle et al. (2012) present an extensive review of the existing literature about cross-docking problems, which range from strategic and tactical to operational problems. One of the operational problems is the truck scheduling problem. The *truck scheduling problem* is concerned with the assignment of inbound and outbound trucks to the different dock doors of a cross-dock (Boysen & Fliedner, 2010; Van Belle et al., 2012). The dock doors can be seen as resources that have to be scheduled over time. A solution of the problem defines *where* (at which dock door) and *when* a truck should be processed. A truck scheduling algorithm then has to find a solution that is optimal with regard to a certain objective function (e.g. minimization of the makespan or the total travel distance). Van Belle et al. (2012) divide the articles about the truck scheduling problem in three categories. A first category considers a simplified cross-dock with a single strip and a single stack door. Truck scheduling reduces in this case to sequencing the inbound and outbound trucks. The articles in the second category consider cross-docks with multiple strip and stack doors, but deal only with scheduling the inbound trucks. The last category then considers articles about scheduling both inbound and outbound trucks at multiple dock doors. The truck scheduling problem studied in this article belongs to this last category. The objective is to minimize the total travel time and the total tardiness (see Section 2).

The truck scheduling problem has attracted the attention of many researchers. Boysen & Fliedner (2010) present a review of articles about the truck scheduling problem and provide a classification of the considered problems. The classification is based on three elements of any truck scheduling problem which are noted as a 'tuple': the 'door environment', operational characteristics and the objective. Several attributes are specified for each of these three main elements. For instance, some attributes of the operational characteristics are pre-emption (allowed or not), processing time to load or unload a truck (fixed or not for all trucks) and intermediate storage (allowed or not).

Yu & Egbelu (2008) consider a truck scheduling problem of the first category. They present a mixed integer programming (MIP) model to minimize the makespan. Compared to the approach presented in this article, no arrival and departure times are considered and the products are assumed to be interchangeable. So, the product assignments from the inbound to the outbound trucks have to be determined as well. Next to the MIP model, Yu & Egbelu introduce a heuristic algorithm. Arabani et al. (2011) and Vahdani & Zandieh (2010) present several metaheuristic algorithms for the same problem.

As an example of the second category, McWilliams et al. (2005, 2008) consider scheduling inbound trucks at a cross-dock used in the parcel delivery industry. In such a cross-docking terminal, unloaded parcels are transported to outbound trucks by a fixed network of conveyors. As this network is stationary, the designation of doors as either strip or stack doors is fixed. This corresponds to the assumption made for the truck scheduling problem considered in this article. The travel time of the parcels is however not only dependent on the assignment of trucks to dock doors (as assumed in this article), but also on congestion of the conveyor network. McWilliams et al. present a simulation-based scheduling algorithm to minimize the makespan. As simulation optimization is computationally expensive, also a decomposition approach is proposed to tackle a similar problem (McWilliams, 2009b, 2010). The objective is now to balance the workload. The problem is formulated as a minimax programming model and is solved by several (meta)heuristic methods (i.a. simulated annealing). A dynamic version of this problem is also studied by McWilliams (2009a).

Miao et al. (2009) study a truck scheduling problem of the third category. It is assumed that the trucks are loaded or unloaded during a fixed time window, so the scheduling problem is reduced to determining the assignment of trucks to dock doors. This simplification cannot be applied to the problem presented in this article, as determining the time windows is explicitly part of the problem. Other differences are that the doors are not strictly divided into strip or stack doors and that the capacity of the cross-dock is limited. The objective is to minimize the operational cost (based on travel time) and the cost of unexecuted shipments. The problem is formulated as an integer programming model while it is solved with a tabu search and a genetic algorithm.

Another truck scheduling problem belonging to the third category is examined by Boysen (2010). In contrast to the approach in this article, products are not allowed to be intermediately stored and the travel times of the products inside the cross-dock are assumed to be negligible. While these assumptions are reasonable for small cross-docks for the food industry (on which Boysen focuses), this is not generally true. To solve the problem, the continuous time space is discretized and a (bounded) dynamic programming approach and a simulated annealing procedure are presented. Three different objectives can be taken into account: minimization of flow time, processing time or tardiness of the outbound trucks.

A problem related to truck scheduling is the *dock door assignment problem* which also tries to find the optimal assignment of inbound and outbound trucks to dock doors. However, time aspects are not taken into account and each truck has to be assigned to a different dock door (it is assumed that there are at least as much dock doors as trucks) (Van Belle et al., 2012). The dock door assignment problem has attracted a considerable amount of attention in literature as well (see for instance Bartholdi & Gue, 2000; Gue, 1999; Oh et al., 2006; Tsui & Chang, 1990, 1992; Yu et al., 2008).

For other articles about the dock door assignment problem and the truck scheduling problem, the interested readers are referred to the aforecited review articles (Boysen, 2010; Van Belle et al., 2012).

The remainder of this article is organized as follows. The next section describes the truck scheduling problem under study in detail and provides a mathematical model of the problem. Section 3 presents the proposed solution approach for finding good quality solutions in a short time period. The results of experimental tests on newly created benchmark instances are given in Section 4. In the last section, conclusions are drawn and future work is shortly discussed.

2. Problem description

The studied problem belongs to the third category of truck scheduling problems and considers a cross-dock with multiple strip and multiple stack doors. Both the inbound and outbound trucks have to be scheduled. The objective is a weighted combination of two objectives. On the one hand, transferring all the goods from inbound to outbound trucks has to be optimized in order to minimize the total workload. On the other hand has the total tardiness of the trucks, with respect to the assigned departure times, to be minimized. So, the considered objective function is the weighted sum of the total travel time and the total tardiness. The basic assumptions for the truck scheduling problem are as follows:

- An exclusive mode of service is considered, i.e. each dock door is either exclusively assigned to inbound or to outbound trucks (e.g. one side of the cross-dock is dedicated to inbound trucks and the other side to outbound trucks).
- Arriving goods are unloaded from the inbound trucks and transferred to the appropriate outbound dock where they are loaded into outbound trucks. Other internal operations - like sorting and labeling - are not considered. Sufficient personnel and equipment are assumed available for performing all loading, unloading and transferring operations.
- Preemption of loading or unloading a truck is not allowed. So, a docked truck has to be completely processed before it leaves the dock.
- For each truck, the (expected) arrival time is known.
- Departure times are defined for all trucks. The departure times are however not considered as hard constraints, but the tardiness of the trucks with respect to these times should be minimized.
- The transported freight is shipped in standardized cargo containers (e.g. pallets). As a consequence, the time required to load or unload one product unit is assumed to be fixed.
- The freight is loaded and unloaded sequentially, i.e. only one freight unit can be loaded or unloaded at the same time. So, the loading or unloading time of a truck is directly proportional to the number of freight units.
- The time needed to transfer goods from inbound to outbound trucks is directly proportional to the distance between the dock doors to which the trucks are assigned.
- Intermediate storage inside the cross-dock is allowed. This means that goods can be unloaded from an inbound truck before the appropriate outbound truck is available. The capacity of the storage area is infinite.
- The truck changeover time is fixed.
- Products are not interchangeable, i.e. any arriving product unit is dedicated to a specific outbound truck.
- The sequence in which goods are loaded or unloaded is not taken into account.

In accordance with the classification scheme proposed by Boysen & Fliedner (2010), this truck scheduling problem can be represented by $[E|r_j, t_{io}|*]$.

A mathematical model of the problem is presented next. The problem consists of n trucks (n_1 inbound trucks and n_2 outbound trucks) and m dock doors (m_1 strip doors and m_2 stack doors). The following parameters are used:

n_1	number of inbound trucks
n_2	number of outbound trucks
m_1	number of strip doors
m_2	number of stack doors
f_{ij}	number of product units that have to be transported from inbound truck i to outbound truck j
v_{ij}	1 if product units have to be transported from inbound truck i to outbound truck j, 0 otherwise
t_{kl}	travel time between strip door k and stack door l
a_i	arrival time of inbound truck i
b_j	arrival time of outbound truck j
c_i	departure time of inbound truck i
d_j	departure time of outbound truck j
L	time needed to load or unload one product unit
T	truck changeover time
w_1	weighting factor for the total travel time
w_2	weighting factor for the total tardiness
M	big number

The following continuous decision variables are defined:

r_i	start time of inbound truck i (time at which truck i enters the dock)
s_j	start time of outbound truck j (time at which truck j enters the dock)
e_i	end time of inbound truck i (time at which truck i leaves the dock)
f_j	end time of outbound truck j (time at which truck j leaves the dock)
t_i	tardiness of inbound truck i
u_j	tardiness of outbound truck j

Finally, the following binary decision variables are used[1]:

$$x_{ik} \begin{cases} 1 & \text{if inbound truck } i \text{ is assigned to strip door } k \\ 0 & \text{otherwise} \end{cases}$$

$$y_{jl} \begin{cases} 1 & \text{if outbound truck } j \text{ is assigned to stack door } l \\ 0 & \text{otherwise} \end{cases}$$

$$z_{ijkl} \begin{cases} 1 & \text{if inbound truck } i \text{ is assigned to strip door } k \text{ and outbound truck } j \text{ is assigned to stack door } l \\ 0 & \text{otherwise} \end{cases}$$

$$p_{ij} \begin{cases} 1 & \text{if inbound trucks } i \text{ and } j \text{ are assigned to the same strip door and truck } i \text{ is a predecessor of truck } j \\ 0 & \text{otherwise} \end{cases}$$

$$q_{ij} \begin{cases} 1 & \text{if outbound truck } i \text{ and } j \text{ are assigned to the same stack door and truck } i \text{ is a predecessor of truck } j \\ 0 & \text{otherwise} \end{cases}$$

The truck scheduling problem can then be formulated as a mixed integer programming model as follows:

$$\min \quad w_1 \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} \sum_{k=1}^{m_1} \sum_{l=1}^{m_2} f_{ij}\, t_{kl}\, z_{ijkl} \; + \; w_2 \left(\sum_{i=1}^{n_1} t_i + \sum_{j=1}^{n_2} u_j \right)$$

[1] Note that the variables z_{ijkl} are required to make the formulation linear.

subject to

$$\sum_{k=1}^{m_1} x_{ik} = 1 \quad (\forall i = 1 \dots n_1) \tag{1}$$

$$\sum_{l=1}^{m_2} y_{jl} = 1 \quad (\forall j = 1 \dots n_2) \tag{2}$$

$$x_{ik} + y_{jl} - 1 \leq z_{ijkl} \quad (\forall i = 1 \dots n_1, \forall j = 1 \dots n_2, \forall k = 1 \dots m_1, \forall l = 1 \dots m_2) \tag{3}$$

$$z_{ijkl} \leq x_{ik} \quad (\forall i = 1 \dots n_1, \forall j = 1 \dots n_2, \forall k = 1 \dots m_1, \forall l = 1 \dots m_2) \tag{4}$$

$$z_{ijkl} \leq y_{jl} \quad (\forall i = 1 \dots n_1, \forall j = 1 \dots n_2, \forall k = 1 \dots m_1, \forall l = 1 \dots m_2) \tag{5}$$

$$x_{ik} + x_{jk} - 1 \leq p_{ij} + p_{ji} \quad (\forall i, j = 1 \dots n_1, i \neq j, \forall k = 1 \dots m_1) \tag{6}$$

$$p_{ij} + p_{ji} \leq 1 \quad (\forall i, j = 1 \dots n_1) \tag{7}$$

$$y_{il} + y_{jl} - 1 \leq q_{ij} + q_{ji} \quad (\forall i, j = 1 \dots n_2, i \neq j, \forall l = 1 \dots m_2) \tag{8}$$

$$q_{ij} + q_{ji} \leq 1 \quad (\forall i, j = 1 \dots n_2) \tag{9}$$

$$r_j \geq a_j \quad (\forall j = 1 \dots n_1) \tag{10}$$

$$r_j \geq e_i + T - M(1 - p_{ij}) \quad (\forall i, j = 1 \dots n_1) \tag{11}$$

$$e_i \geq r_i + L \sum_{j=1}^{n_2} f_{ij} \quad (\forall i = 1 \dots n_1) \tag{12}$$

$$s_j \geq b_j \quad (\forall j = 1 \dots n_2) \tag{13}$$

$$s_j \geq f_i + T - M(1 - q_{ij}) \quad (\forall i, j = 1 \dots n_2) \tag{14}$$

$$f_j \geq s_j + L \sum_{i=1}^{n_1} f_{ij} \quad (\forall j = 1 \dots n_2) \tag{15}$$

$$f_j \geq e_i + \sum_{k=1}^{m_1} \sum_{l=1}^{m_2} t_{kl} z_{ijkl} + f_{ij} L - M(1 - v_{ij}) \quad (\forall i = 1 \dots n_1, \forall j = 1 \dots n_2) \tag{16}$$

$$t_i \geq e_i - c_i \quad (\forall i = 1 \dots n_1) \tag{17}$$

$$u_j \geq f_j - d_j \quad (\forall j = 1 \dots n_2) \tag{18}$$

$$r_i \geq 0, e_i \geq 0, t_i \geq 0 \quad (\forall i = 1 \dots n_1) \tag{19}$$

$$s_j \geq 0, f_j \geq 0, u_j \geq 0 \quad (\forall j = 1 \dots n_2) \tag{20}$$

The objective is to minimize the weighted sum of the total travel time and the total tardiness. Constraints (1) ensure that every inbound truck is assigned to a strip door and similarly, constraints (2) ensure that every outbound truck is assigned to a stack door. Constraints (3)–(5) define the correct relationship between the x_{ik}, y_{jl} and z_{ijkl} variables. The correct relationship between the x_{ik} and p_{ij} variables for the inbound trucks is expressed by constraints (6) and (7). Note that constraints (7) enforce that $p_{ii} = 0$. In a similar way, constraints (8) and (9) express the relationship between the y_{il} and q_{ij} variables for the outbound trucks. Constraints (10) and (11) determine the start time of each inbound truck as the maximum of the arrival time of the truck and the end time of its predecessor plus truck changeover time:

$$r_j = \max(a_j, \max_{i=1 \dots n_1} p_{ij}(e_i + T)) \tag{21}$$

Due to the assumptions of sequential unloading and sufficient resource availability, the end time of each inbound truck is equal to its start time plus the time required to unload all products:

$$e_i = r_i + L \sum_{j=1}^{n_2} f_{ij} \tag{22}$$

This is expressed by constraints (12). For the outbound trucks, the start time can be defined in a similar way as for the inbound trucks:

$$s_j = \max(b_j, \max_{i=1 \dots n_2} q_{ij}(f_i + T)) \tag{23}$$

This is enforced by constraints (13) and (14). The end time is at least as great as its start time plus the time required to load all products. However, the end time is also constrained by the end times[2] of the inbound trucks that ship products for the outbound truck. The end time has to be greater than the latest end time of these inbound trucks augmented with the time to transfer the appropriate product units to the outbound truck and the time to load these units. So, the end time of the outbound trucks can be expressed as follows:

$$f_j = \max(s_j + L\sum_{i=1}^{n_1} f_{ij}, \max_{i=1\dots n_1} \quad v_{ij}(e_i + \sum_{k=1}^{m_1}\sum_{l=1}^{m_2} t_{kl}z_{ijkl} + f_{ij}L)) \tag{24}$$

In this expression, $\sum_{k=1}^{m_1}\sum_{l=1}^{m_2} t_{kl}z_{ijkl}$ denotes the travel time from inbound truck i to outbound truck j. The expression is enforced by constraints (15) and (16). Note, however, that this does not completely prevent parallel loading (of goods from different inbound trucks) if multiple inbound trucks have similar values of end time plus transfer time. Therefore, the loading sequences of the outbound trucks also have to be determined. This requires introducing extra variables and constraints and will be omitted in order not to unnecessarily complicate the formulation. For the tabu search approach however, an effective - yet simple - algorithm was developed for determining the correct end times (see Section 3). Constraints (17) and (19) then determine the tardiness of each inbound truck, which is defined as:

$$t_i = \max(e_i - c_i, 0) \tag{25}$$

Similarly, the tardiness of each outbound truck is given by:

$$u_j = \max(f_j - d_j, 0) \tag{26}$$

This is expressed by constraints (18) and (20). All continuous decision variables have to be greater than or equal to zero, which is expressed by constraints (19) and (20).

This problem formulation is quite large. For instance, a moderate cross-dock with 5 strip doors, 5 outbound dock doors, 10 inbound trucks and 10 outbound trucks requires a MIP model with 2860 variables and 8980 constraints. In order to avoid long computation times (as reported in Section 4), a heuristic method is proposed to solve it in a reasonable amount of time.

3. Tabu search approach

A tabu search (TS) approach (Glover, 1989) was developed for the truck scheduling problem. In the next paragraphs, the details of this approach are described.

Solution representation. A solution of this truck scheduling problem can be represented by a sequence of pairs of dock doors and trucks (see Figure 1). For n trucks, the solution will have a length of n. The first n_1 pairs correspond to the strip doors (ID) and inbound trucks (IT), the next n_2 pairs correspond to the stack doors (OD) and outbound trucks (OT). Figure 2 shows an example solution. The first row corresponds to the dock doors, the second row is a permutation of the trucks. If multiple trucks are assigned to the same dock door, the sequence in which these trucks appear in the solution defines the sequence in which the trucks are assigned to the dock door. For instance, for the solution shown in Figure 2, the sequence of the inbound trucks at strip door 1 is 1–3–5. This sequence completely defines the truck schedule. The start times of the trucks can be determined by equations (21) and (23) and the end times of the inbound trucks by equations (22). The end times of the outbound trucks can be determined either by equation (24) or by Algorithm 1. In Algorithm 1, the exact end times of the outbound trucks are determined (together with the start times) by considering their optimal loading sequences with respect to the tardiness (i.e. based on the first-come, first-served policy).

[2] As the unloading sequence of the inbound trucks is unknown, it is assumed that goods from an inbound truck can only be transferred after complete unloading of the truck, i.e. after its end time.

ID	OD
IT	OT

Figure 1: Solution representation.

1	2	1	2	1	2	1	2	1	2
1	2	3	4	5	1	2	3	4	5

Figure 2: Example solution representation for $n_1 = n_2 = 5$ and $m_1 = m_2 = 2$.

Initial solution. The inbound and outbound trucks are sorted by arrival times and the strip and stack doors are sorted by their average distance to the stack and strip doors respectively. The sorted trucks are then assigned one by one to the next dock door in the sorted list. If a truck is assigned to the last dock door in the list, the next truck is again assigned to the first dock door of the sorted list. For instance, if the arrival sequence of both inbound and outbound trucks is 1–2–3–4–5, and the sorted lists of the dock doors are 1–2 and 2–1, the solution shown in Figure 2 would be the initial solution.

Neighborhood. A composite neighborhood structure is used, consisting of two neighborhoods based on the following two moves:

- Swap move: two trucks are interchanged, i.e. the first truck is assigned to the dock door and the position in the sequence of the second truck, and vice versa (see Figure 3). If both trucks were assigned to the same dock door, only their position in the sequence would change.
- Insert move: a truck is assigned to another dock door (see Figure 4). The position of the truck at this dock door is determined by the position in the sequence of the solution.

Algorithm 1 Pseudocode of the algorithm to determine the exact end times of the outbound trucks.

```
function CALCULATEENDTIMES($ID, IT, OD, OT$)
    for $h \leftarrow 1 \dots n_2$ do              ▷ Iterate over the outbound trucks corresponding to their sequence
        $j \leftarrow OT_h \quad l \leftarrow OD_h$      in the solution.
        $s_j \leftarrow \max(b_j, dt_l)$            ▷ The start time of truck $j$ is the maximum of its arrival time and
                                       the time dock $l$ is available ($dt_l$).
        $f_j \leftarrow s_j$
        for $g \leftarrow 1 \dots n_1$ do          ▷ Iterate over the inbound trucks corresponding to their sequence in
            $i \leftarrow IT_g \quad k \leftarrow ID_g$      the solution.
            $lt_i \leftarrow 0$                      ▷ Determine the time at which goods from truck $i$ can be loaded
            if $v_{ij} = 1$ then                  into truck $j$ ($lt_i$). If truck $i$ contains goods for truck $j$, $lt_i$ is equal
                $lt_i \leftarrow e_i + t_{kl}$             to the end time of truck $i$ augmented with the travel time.
            end if
        end for
        $ITS \leftarrow$ SORT($IT, lt$)            ▷ Sort the inbound trucks by $lt_i$ (first-come, first-served).
        for $g \leftarrow 1 \dots n_1$ do          ▷ Iterate over the inbound trucks in order of their $lt_i$.
            $i \leftarrow ITS_g$
            $f_j \leftarrow \max(lt_i, f_j) + Lf_{ij}$   ▷ The end time of loading from truck $i$ into truck $j$ is equal to the
                                       start time (the maximum of $lt_i$ and the end time of loading from
                                       the predecessor of truck $i$) augmented with the loading time.
        end for
        $dt_l \leftarrow f_j + T$                  ▷ Adapt the time dock $l$ is available.
    end for
end function
```

1	2	1	2	1	2	1	2	1	2
1	**2**	3	4	**5**	1	2	3	4	5

↓

1	2	1	2	1	2	1	2	1	2
1	**5**	3	4	**2**	1	2	3	4	5

Figure 3: Example of a swap move: the inbound truck sequence at strip doors 1 and 2 changed from 1–3–5 and 2–4 to 1–3–2 and 5–4.

1	2	1	2	1	2	1	**2**	1	2
1	2	3	4	5	1	2	3	4	5

↓

1	2	1	2	1	2	1	**1**	1	2
1	2	3	4	5	1	2	3	4	5

Figure 4: Example of an insert move: the outbound truck sequence at stack doors 1 and 2 changed from 2–4 and 1–3–5 to 2–3–4 and 1–5.

Note that swap moves do not change the number of trucks assigned to the same dock door. Insert moves however allow that the trucks are redistributed over the dock doors. Both moves have to be applied to either the inbound part or the outbound part of the solution.

Tabu list. A tabu list with a fixed tenure t is used. The recent moves are stored in this list, not the recent solutions. The swap moves are characterized by the indices of the trucks that are interchanged (2 and 5 for the example in Figure 3), the insert moves by the index of the truck that is assigned to another dock door and that dock door (8 and 1 for the example in Figure 4). No aspiration criterion is used.

Termination criterion. The tabu search is stopped when the current best solution has not improved during i consecutive iterations or when the maximum allowed calculation time c has elapsed.

4. Experimental results

In this section, several test scenarios are defined and the solutions generated by the tabu search are compared with the exact solutions found by solving the MIP model. First, the experimental set-up is described, and then the results are discussed.

4.1. Experimental set-up

Several factors influence the hardness of the truck scheduling instances:

1. The problem size, i.e. the number of trucks and dock doors.
2. The ratio of the number of trucks to the number of dock doors. If this ratio is low, the truck scheduling problem is almost reducible to the assignment of trucks to dock doors. In case of a high ratio, sequencing the trucks at the dock doors has to be taken into account as well.
3. The flow mix, i.e. the distribution of the arriving goods for the outbound trucks. This characteristic can be represented by the 'density' of the flow matrix (containing f_{ij}). If this matrix is sparse, the inbound trucks contain products for only one or a few outbound trucks. If the flow matrix is dense, the inbound trucks transport goods for (almost) all outbound trucks.
4. The simultaneousness of the truck arrivals. Sequencing the trucks becomes more difficult if the trucks arrive within narrow time frames. If the truck arrivals are better spread over time, the optimal sequence can be derived from the arrival sequence.
5. The tightness of the time windows (formed by the arrival and departure times).

For all experiments, the I-shaped cross-dock shown in Figure 5 is considered. The cross-dock has three strip doors and three stack doors and the (rectilinear) travel times between the strip and stack doors are indicated on the figure. The aforementioned factors have been reflected while generating different problem instances by varying three parameters:

Number of trucks By changing the number of trucks while the number of dock doors is fixed, variations can be obtained in both factors 1 and 2. This parameter can have one of the following three values:

- Low: 4 or 5 inbound trucks and 4 or 5 outbound trucks.
- Medium: 6 or 7 inbound trucks and 6 or 7 outbound trucks.
- High: 8 or 9 inbound trucks and 8 or 9 outbound trucks.

Flow mix This parameter determines the number of outbound trucks for which the inbound trucks contain products (factor 3). There are three values:

- Low: each inbound truck contains items for 25 to 50 % of the outbound trucks.
- Medium: each inbound truck contains items for 50 to 75 % of the outbound trucks.
- High: each inbound truck contains items for 75 to 100 % of the outbound trucks.

The flow matrix (containing f_{ij}) can be determined based on this value. First, the outbound trucks to which the items are transferred are randomly chosen for each inbound truck. The number of items to be transferred to these outbound trucks is randomly sampled between 1 and C/nbr, in which C is the capacity (the total number of items one truck can transport) and nbr is the number of outbound trucks for which the inbound truck contains items. If this procedure leads to empty or overloaded trucks, the resulting flow matrix is discarded and the same procedure is repeated.

Time window This parameter determines how the arrival and departure times of the trucks are generated (and influences factors 4 and 5). For each truck, the arrival and departure times are sampled in a certain time interval. The length t of this time interval is directly proportional to the number of trucks arriving or leaving in that interval: $t = a * nbr$ for the arrival times and $t = d * nbr$ for the departure times, in which nbr is the number of (arriving or leaving) trucks. The values of a and d are determined by one of the three values of this parameter:

- Low: $a = 30$ and $d = 15$.
- Medium: $a = 20$ and $d = 10$.
- High: $a = 10$ and $d = 5$.

For the inbound trucks, the arrival times are sampled from the interval $[0, t]$. If e is the average arrival time of the inbound trucks, the arrival times of the outbound trucks are then sampled from the interval $[e, e + t]$. The departure times of the trucks are sampled from the interval $[f, f + t]$, in which f is the arrival time of the truck.

Each combination of parameter values (27 in total) denotes a problem type. For each problem type 10 instances were generated. These benchmark instances have been made available to the public at `http://address.not.included.because.double.blind.review`. The parameters that are fixed are shown in Table 1.

4.2. Experimental results

All 270 problem instances have been solved with the proposed tabu search (TS) and with the CPLEX© solver[3]. All the experiments were run on a PC with two 2.40 GHz hex-core processors and with 96 GB RAM.

[3] IBM ILOG CPLEX Optimization Studio 12.5.

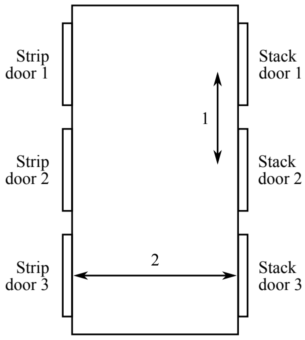

Figure 5: The cross-dock designed for the experiments.

Table 1: Fixed parameter values.

Variable	Value
L	2
T	3
w_1	1
w_2	2
C	33
t	16
i	10000
c	5 s

As it takes CPLEX© a long time to find the optimal solution for the larger problems, an upper bound of two hours (7200 s) is applied. If the optimal solution is not found within this time limit, the current best solution generated by the solver is reported. As explained in Section 2, the MIP model makes use of the simplified equation (24) to calculate the end times of the outbound trucks. For a fair comparison of the resulting objective values, the tabu search implementation makes also use of this simplified expression. For illustrative purposes, the results of applying the proposed tabu search with exact end times of the outbound trucks (computed by Algorithm 1) are also determined.

Table 2 summarizes the results. The first column shows the problem types, indicated by the value (L = low, M = medium, H = high) for its three parameters: number of trucks, flow mix and time window. For each problem type, the average objective value over its 10 instances is calculated. These values are shown in the next three columns for the tabu search approach with the simplified expression (TS1), the tabu search approach with the exact end times (TS2) and the MIP approach. The lowest average values for all problem types are indicated in bold. The fifth column indicates how many of the 10 problem instances have been proven optimal by CPLEX©. In order to compare the tabu search (TS1) and the exact approach, the best of both approaches is determined for each instance. The next two columns then show how many times tabu search or CPLEX© was the winning approach (in case of a draw, both are counted). The solution of the tabu search approach is then compared with this best solution. If TS_i is the solution of problem instance i found by tabu search and BC_i the solution found by CPLEX©, the average relative deviation (in percentage) is calculated as follows:

$$100\left(\frac{TS_i - \min(TS_i, BC_i)}{\min(TS_i, BC_i)}\right)$$

The average value for each problem type is shown in the next column. Finally, the average calculation times of the three approaches are shown in the last three columns.

Table 2 shows that tabu search is able to find good results in a short time period. For the problems with the lower parameter values, tabu search (TS1) was almost always able to find the optimal solution. For all

Table 2: The experimental results for the 27 considered problem types (L = low, M = medium, H = high).

Problem type	Objective value			# MIP optimal	# TS1 best	# MIP best	Relative dev. (%)	Calculation time (s)		
	TS1	TS2	MIP					TS1	TS2	MIP
L L L	**518.300**	528.299	**518.300**	10	10	10	0.000	0.167	0.455	0.215
L L M	**592.072**	595.331	**592.072**	10	10	10	0.000	0.166	0.457	0.609
L L H	**837.478**	856.069	**837.478**	10	10	10	0.000	0.157	0.433	1.299
L M L	**526.391**	549.479	**526.391**	10	10	10	0.000	0.149	0.541	0.802
L M M	**654.059**	682.388	**654.059**	10	10	10	0.000	0.157	0.568	1.046
L M H	**830.229**	860.750	**830.229**	10	10	10	0.000	0.147	0.511	0.762
L H L	**684.285**	698.018	**684.285**	10	10	10	0.000	0.171	0.811	2.519
L H M	**601.945**	612.169	**601.945**	10	10	10	0.000	0.160	0.721	3.150
L H H	**957.740**	974.641	**957.740**	10	10	10	0.000	0.154	0.682	4.588
M L L	**695.459**	699.873	**695.459**	10	10	10	0.000	0.421	1.820	3.452
M L M	**962.622**	974.925	**962.622**	10	10	10	0.000	0.433	1.754	4.816
M L H	**1236.073**	1254.682	**1236.073**	10	10	10	0.000	0.419	1.751	4.254
M M L	**615.812**	624.649	**615.812**	10	10	10	0.000	0.445	2.747	2.236
M M M	**1004.442**	1021.004	**1004.442**	10	10	10	0.000	0.415	2.498	20.426
M M H	1443.384	1461.398	**1442.850**	10	8	10	0.030	0.445	3.042	79.618
M H L	875.440	884.003	**875.140**	10	9	10	0.035	0.430	3.359	12.897
M H M	1150.058	1170.172	**1149.558**	10	9	10	0.050	0.430	3.186	53.642
M H H	1704.644	1723.057	**1704.087**	9	9	9	0.065	0.514	3.929	1769.894
H L L	1033.019	1038.157	**1032.119**	10	8	10	0.148	0.902	4.519	12.263
H L M	1197.026	1207.316	**1197.013**	10	9	10	0.001	0.859	4.560	35.782
H L H	**1859.446**	1885.563	1861.349	5	9	7	0.025	1.056	4.921	4284.281
H M L	1037.082	1041.362	**1036.773**	10	9	10	0.024	1.027	5.002	704.793
H M M	**1151.924**	1155.059	**1151.924**	9	10	10	0.000	1.017	5.002	1305.639
H M H	1985.397	2006.045	**1982.719**	3	8	6	0.228	1.007	5.001	5218.501
H H L	1281.025	1292.770	**1279.053**	9	6	10	0.189	1.027	5.002	1419.368
H H M	1712.346	1729.040	**1712.254**	4	9	8	0.022	1.162	5.002	4897.159
H H H	**2437.626**	2466.332	2439.104	1	7	6	0.055	1.037	5.002	6516.383

problems, the average relative deviation compared to the best solution found is lower than 0.25 %. When looking at the results of the individual problem instances, the largest observed relative deviation is lower than 2 %. CPLEX$^{©}$ was not able to find the optimal solution for all problem instances within the time limit of two hours. For 30 of the 270 problem instances, the optimal solution was not found. Tabu search was able to find the same solution for 2 out of these 94 instances, and a better solution for 14 problem instances. In total, the tabu search approach found a solution at least as good as the CPLEX$^{©}$ result for 250 problem instances. Tabu search is also very efficient. The solutions of all problem instances were found in less than 2 s, and the average calculation time per problem type is less than 1.2 s.

To prevent loading the outbound trucks in parallel, the second tabu search implementation (TS2) makes use of Algorithm 1 to determine the end times of the outbound trucks. This implies that the objective values obtained with TS2 are at least as large as the optimal objective values of the MIP model. Comparing the results of TS1 and the MIP approach shows that the average relative deviation between the objective values of both is 1.5 %, with a maximum deviation of 12.5 %. This indicates that also TS2 is able to generate good quality results. The calculation of the exact end times is however computationally more expensive, which results in higher calculation times for TS2 than for TS1.

5. Conclusions and future work

This article introduces the truck scheduling problem that is concerned with scheduling both inbound and outbound trucks at multiple dock doors. The objective is to minimize the weighted combination of total travel time and total tardiness. The problem is formulated as a mixed integer programming model that can be solved to optimality for small size problems. A tabu search approach has been introduced as well as an effective procedure for computing good quality solutions. A set of benchmark instances with varying characteristics was developed. The experimental results indicate that the proposed tabu search is able to find good results in a short time period. For all considered problem instances, the average relative deviation compared to the best solution found is lower than 0.25 %, while the average calculation time is less than 1.2 s.

Although various real-world details are taken into account, several others have not been considered (e.g. interchangeable products, limited storage capacity and internal congestion). Also, uncertainty and variability are not taken into account and the problem is assumed to be static (while in practice trucks arrive late, equipment fails, etc.). So, future research should incorporate these issues in the truck scheduling problem in order to increase the applicability.

The authors advocate another approach and will use the solution obtained with the proposed tabu search approach as an input to a logistic execution system (LES) (Van Belle et al., 2011). This software system is responsible for the real-time execution of the logistic operations of the cross-dock. It is based on a self-organizing and decentralized approach in order to improve the responsiveness and proactiveness and to handle changes and disturbances as business-as-usual. The LES ensures that the generated truck schedule can be executed in reality by accounting for omitted details and by reacting to variations and disturbances. On the other hand, the LES can benefit from the global view provided by a good quality truck schedule to improve its performance. So, the idea of the cooperation is to combine the robustness and flexibility of the LES with the optimization of the organizational objectives by the presented scheduling approach.

References

Arabani, A. R. B., Ghomi, S. M. T. F., & Zandieh, M. (2011). Meta-heuristics implementation for scheduling of trucks in a cross-docking system with temporary storage. *Expert Systems with Applications*, *38*, 1964–1979.

Bartholdi, J. J., III, & Gue, K. R. (2000). Reducing labor costs in an LTL crossdocking terminal. *Operations Research*, *48*, 823–832.

Boysen, N. (2010). Truck scheduling at zero-inventory cross docking terminals. *Computers & Operations Research*, *37*, 32–41.

Boysen, N., & Fliedner, M. (2010). Cross dock scheduling: Classification, literature review and research agenda. *Omega*, *38*, 413–422.

Glover, F. (1989). Tabu search—part I. *ORSA Journal on Computing*, *1*, 190–206.

Gue, K. R. (1999). The effects of trailer scheduling on the layout of freight terminals. *Transportation Science*, *33*, 419–428.

McWilliams, D. L. (2009a). A dynamic load-balancing scheme for the parcel hub-scheduling problem. *Computers & Industrial Engineering*, *57*, 958–962.

McWilliams, D. L. (2009b). Genetic-based scheduling to solve the parcel hub scheduling problem. *Computers & Industrial Engineering*, *56*, 1607–1616.

McWilliams, D. L. (2010). Iterative improvement to solve the parcel hub scheduling problem. *Computers & Industrial Engineering*, *59*, 136–144.

McWilliams, D. L., Stanfield, P. M., & Geiger, C. D. (2005). The parcel hub scheduling problem: A simulation-based solution approach. *Computers & Industrial Engineering*, *49*, 393–412.

McWilliams, D. L., Stanfield, P. M., & Geiger, C. D. (2008). Minimizing the completion time of the transfer operations in a central parcel consolidation terminal with unequal-batch-size inbound trailers. *Computers & Industrial Engineering*, *54*, 709–720.

Miao, Z., Lim, A., & Ma, H. (2009). Truck dock assignment problem with operational time constraint within crossdocks. *European Journal of Operational Research*, *192*, 105–115.

Oh, Y., Hwang, H., Cha, C. N., & Lee, S. (2006). A dock-door assignment problem for the Korean mail distribution center. *Computers & Industrial Engineering*, *51*, 288–296.

Tsui, L. Y., & Chang, C.-H. (1990). A microcomputer based decision support tool for assigning dock doors in freight yards. *Computers & Industrial Engineering*, *19*, 309–312.

Tsui, L. Y., & Chang, C.-H. (1992). An optimal solution to a dock door assignment problem. *Computers & Industrial Engineering*, *23*, 283–286.

Vahdani, B., & Zandieh, M. (2010). Scheduling trucks in cross-docking systems: Robust meta-heuristics. *Computers & Industrial Engineering*, *58*, 12–24.

Van Belle, J., Saint Germain, B., Valckenaers, P., Van Brussel, H., Bahtiar, R., & Cattrysse, D. (2011). Intelligent products in the supply chain are merging logistic and manufacturing operations. In S. Bittanti, A. Cenedese, & S. Zampieri (Eds.), *Preprints of the 18th IFAC World Congress* (pp. 1596–1601).

Van Belle, J., Valckenaers, P., & Cattrysse, D. (2012). Cross-docking: State of the art. *Omega*, *40*, 827–846.

Yu, V. F., Sharma, D., & Murty, K. G. (2008). Door allocations to origins and destinations at less-than-truckload trucking terminals. *Journal of Industrial and Systems Engineering*, *2*, 1–15.

Yu, W., & Egbelu, P. J. (2008). Scheduling of inbound and outbound trucks in cross docking systems with temporary storage. *European Journal of Operational Research*, *184*, 377–396.