On the complexity of the crossdock truck-scheduling problem

Q. Fabry[1,3], A. Agnetis[2], L. Berghman[1] and C. Briand[3]

[1] Université de Toulouse - Toulouse Business School,
20 BD Lascrosses – BP 7010, 31068 Toulouse Cedex 7, France
l.berghman@tbs-education.fr

[2] Università degli Studi di Siena, DIISM, Siena, Italy
agnetis@diism.unisi.it

[3] LAAS-CNRS, Université de Toulouse, UPS, Toulouse, France
{quentin.fabry,cyril.briand}@laas.fr

Keywords: crossdocking, truck scheduling, complexity.

1 Introduction

Crossdocking is a warehouse management concept in which items delivered to a warehouse by inbound trucks are immediately sorted out, reorganized based on customer demands and loaded into outbound trucks for delivery to customers, without requiring excessive inventory at the warehouse (J. van Belle *et al.* 2012). If any item is held in storage, it is usually for a brief period of time that is generally less than 24 hours. Advantages of crossdocking can accrue from faster deliveries, lower inventory costs, and a reduction of the warehouse space requirement (U.M. Apte and S. Viswanathan 2000, N. Boysen *et al.* 2010). Compared to traditional warehousing, the storage as well as the length of the stay of a product in the warehouse is limited, which requires an appropriate coordination of inbound and outbound trucks (N. Boysen 2010, W. Yu and P.J. Egbelu 2008).

The crossdock truck-scheduling problem (CTSP), which decides on the succession of truck processing at the dock doors, is especially important to ensure a rapid turnover and on-time deliveries. The problem studied concerns the operational level: trucks are allocated to the different docks so as to minimize the storage usage during the product transfer. The internal organization of the warehouse (scanning, sorting, transporting) is not explicitly taken into consideration. We also do not model the resources that may be needed to load or unload the trucks, which implies the assumption that these resources are available in sufficient quantities to ensure the correct execution of an arbitrary docking schedule. In this abstract, we present some new complexity results that refer to a situation in which the number of docks (or doors) at the terminal is small, namely one or two. This situation has been indeed addressed in the literature, e.g. (A. Chiarello *et al.* 2018). However most authors focus on tardiness objectives, while we focus on minimizing overall soujourn time of the pallets, which is especially meaningful for perishable goods or for reducing stock holding costs.

This abstract is structured as follows: Section 2 formalizes the problem and introduces some basic notations, Section 3 addresses the complexity of the crossdocking truck scheduling problem in various scenarios (complexity proofs are not provided for the sake of conciseness), then a few concluding remarks are provided.

2 Detailed problem statement

We consider a crossdocking warehouse where *inbound trucks* $i \in I$ need to be unloaded and *outbound trucks* $o \in O$ need to be loaded (where I is the set of all inbound trucks and O is the set of all outbound trucks). The warehouse features n docks that can be used both for loading and unloading. The unloading and loading processing times of trucks $i \in I$ and $o \in O$ are referred to as p_i and p_o, respectively. Similarly, let W_i (respectively, W_o) denote the number of *pallets* to be unloaded from i (respectively, to be loaded on o). We let w_{io} denote the number of pallets that must be transferred from i to o. It is sometime convenient to visualize an instance of the problem through a bipartite graph $G_T = (I, O, P)$ called *transfer graph*. In G_T, the two node sets correspond to inbound and outbound trucks respectively, and there is an arc (i, o) if

$w_{io} > 0$. The arc set P expresses start-start precedence constraints, i.e., if $(i, o) \in P$, truck $o \in O$ cannot start being loaded before truck $i \in I$ starts being unloaded. In this paper we consider two scenarios:

(i) There is no relationship between the number of pallets that need to be loaded/unloaded and the processing time of a truck. In this case, for any two trucks h and k, in general $W_h/p_h \neq W_k/p_k$. We say that in this scenario processing times are *unrelated*;

(ii) The loading/unloading time of a truck is proportional to the number of pallets that must be loaded/unloaded. For simplicity, in this case we assume that the processing times are expressed in terms of number of pallets being moved, i.e.,

$$p_i = W_i = \sum_{o \in O} w_{io}, \forall i \in I \tag{1}$$

and

$$p_o = W_o = \sum_{i \in I} w_{io}, \forall o \in O \tag{2}$$

We say that in this scenario processing times are *correlated*. Notice that, in this case,

$$\sum_{o \in O} p_o = \sum_{i \in I} p_i. \tag{3}$$

It is assumed that there is sufficient workforce to load/unload all docked trucks at the same time. Hence, a truck assigned to a dock does not wait for the availability of a material handler.

Products can be transshipped directly from an inbound to an outbound truck if the outbound truck is placed at a dock. Otherwise, the products are temporarily stored and will be loaded later on. The problem is to determine time-consistent start times s_i and s_o of unload and load tasks $i \in I$ and $o \in O$ so as to minimize the *total time spent in the warehouse by all pallets* (total flow time). For each pallet which has to be transferred from i to o such a flow time equals $s_o - s_i$. Therefore, the total flow time is

$$\sum_{(i,o) \in P} w_{io}(s_o - s_i). \tag{4}$$

In what follows, $CTSP(n, U)$ denotes the problem with n gates and unrelated processing times, while $CTSP(n, C)$ denotes the problem with n gates and correlated processing times.

Due to (1) and (2), it is easy to show that problem $CTSP(n, C)$ consists in finding the feasible schedule that minimizes

$$\sum_{o \in O} p_o s_o - \sum_{i \in I} p_i s_i. \tag{5}$$

3 Complexity results

Let us first consider the problem CTSP when $n = 1$, i.e., the crossdocking platform has a single gate, and let us start with the special case in which the transfer graph G_T is complete, i.e., it is a "1-biclique" (Figure 1). This means that $w_{io} > 0$ for each $i \in I$ and $o \in O$, i.e., each inbound truck has at least one pallet that must be transferred to each outbound truck.

Let us consider the unrelated problem CTSP(1,U) in the "1-biclique" case. Since G_T is complete, in any feasible schedule *all* inbound trucks must be consecutively scheduled, before all outbound trucks. So, the problem consists of deciding in which order they should be scheduled. The following property holds.

Theorem 1. *When G_T is a biclique, $CTSP(1, U)$ is solved by first scheduling all inbound trucks in nonincreasing order of the ratio p_i/W_i, then all outbound trucks in nondecreasing order of the ratio p_o/W_o.*

□

Note that such an optimal sequence can be obtained in $O(n \log n)$. Concerning problem $CTSP(1, C)$, recalling (1) and (2), Theorem 1 implies that, when G_T is a biclique, $CTSP(1, C)$ is solved by scheduling all inbound trucks before all outbound trucks, in any order. Theorem 1 easily extends to the case in which G_T consists of k disjoint bicliques (e.g., see Figure 2 with $k = 3$).

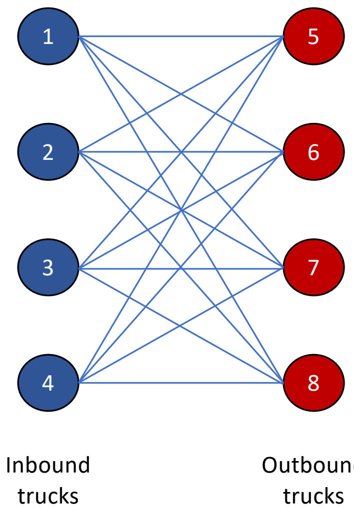

Fig. 1. G_T in the case of 1-biclique.

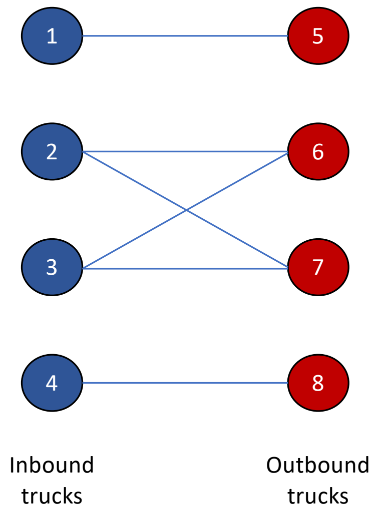

Fig. 2. G_T in the case of 3-biclique.

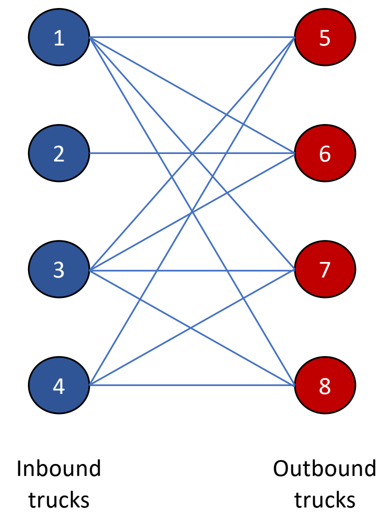

Fig. 3. G_T in the general case.

Corollary 1. *When G_T is a collection of bicliques, $CTSP(1, U)$ is solved by sequencing the trucks involved in each biclique consecutively as dictated by Theorem 1, and then sequencing the bicliques in any order.* □

Let us now turn to problem $CTSP(1, U)$ when G_T has a general structure (see Figure 3), which can be stated in decision form as follows.

"Given a positive integer H, is there a truck sequence at the dock such that the total flow time does not exceed H?"

The following result holds.

Theorem 2. *CTSP(1,U) is NP-complete.*

Proof. Reduction from OPTIMAL LINEAR ARRANGEMENT. □

The complexity of $CTSP(1, C)$ when G_T has a general structure remains open.

Let us now turn to $CTSP(2, C)$, i.e., the case in which there are two gates and processing times are correlated. The following result holds:

Theorem 3. *CTSP(2,C) is NP-complete even when G_T is a 1-biclique.*

Proof. Reduction from PARTITION. □

4 Conclusion

In conclusion, we summarize our findings in the following table (where NPC stands for NP-Complete). Note that the case where G_T has a bi-clique structure is significant to determines the frontier between the polynomial and NP-complete cases. Moreover, as it is always possible (by removing arcs) to transform a general G_T graph in order to give it a bi-clique structure, having efficient methods to solve the biclique case can give good lower bounds for the general case.

n	C , 1-biclique	C	U, 1-biclique	U
1	$O(n)$	open	$O(n \log n)$ (Th. 1)	NPC (Th.2)
2	NPC (Th.3)	NPC (Th.3)	NPC (Th.3)	NPC (Th.3)

References

U.M. Apte and S. Viswanathan, Effective cross docking for improving distribution efficiencies, International Journal of Logistics: Research and Applications, 3 (3), 291–302.

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

N. Boysen and M. Fliedner and A. Scholl, Scheduling inbound and outbound trucks at cross docking terminals, OR Spectrum, 32, 135–161.

Chiarello, A., Gaudioso, M., Sammarra, M., Truck synchronization at single door crossdocking terminals, OR Spectrum 40(2), 395–447 (2018).

M. E. Dyer and L. A. Wolsey, Formulating the single machine sequencing problem with release dates as a mixed integer problem, Discrete Applied Mathematics, 26, 255–270.

Garey, M.R., R.L. Graham, D.S. Johnson, D.E.Knuth, Complexity results for bandwidth minimization, *SIAM Journal on Applied Mathematics*, 34, 477–495.

J.K. Lenstra and A.H.G. Rinnooy Kan and P. Brucker, Complexity of machine scheduling problems, Annals of Discrete Mathematics, 1, 343–362.

M. Lombardi and M. Milano, A min-flow algorithm for Minimal Critical Set detection in Resource Constrained Project Scheduling, Artificial Intelligence, 182-183, 58–67.

J. van Belle and P. Valckenaers and D. Cattrysse, Cross docking: State of the art, Omega,40 (6), 827–846.

W. Yu and P.J. Egbelu, Scheduling of inbound and outbound trucks in cross docking systems with temporary storage, *International Journal of Production Economics*, 184, 377–396.