Cross-docking: State of the art — Source link

Jan Van Belle, Paul Valckenaers, Dirk Cattrysse

Institutions: Katholieke Universiteit Leuven

Published on: 01 Dec 2012 - Omega-international Journal of Management Science (Pergamon)

Topics: Cross-docking

Cross-docking: State of the art

Jan Van Belle*, Paul Valckenaers*, Dirk Cattrysse*

Katholieke Universiteit Leuven, Department of Mechanical Engineering, Celestijnenlaan 300B, B-3001 Heverlee (Leuven), Belgium

Abstract

Cross-docking is a logistics strategy in which freight is unloaded from inbound vehicles and (almost) directly loaded into outbound vehicles, with little or no storage in between. This article presents an overview of the cross-docking concept. Guidelines for the successful use and implementation of cross-docking are discussed and several characteristics are described that can be used to distinguish between different cross-dock types. In addition, this article presents an extensive review of the existing literature about cross-docking. The discussed articles are classified based on the problem type that is tackled (ranging from more strategical or tactical to more operational problems). Based on this review, several opportunities to improve and extend the current research are indicated.

Keywords: Cross-docking, Logistics, Classification

1. Introduction

Cross-docking is a logistics strategy nowadays used by many companies in different industries (e.g. retail firms and less-than-truckload (LTL) logistics providers). The basic idea behind cross-docking is to transfer incoming shipments directly to outgoing vehicles without storing them in between. This practice can serve different goals: the consolidation of shipments, a shorter delivery lead time, the reduction of costs, etc. The role of cross-docking in industry even seems to increase [1, 2].

In a traditional warehouse, goods are first received and then stored, for instance in pallet racks. When a customer requests an item, workers pick it from the storage and ship it to the destination. From these four major functions of warehousing (receiving, storage, order picking and shipping), storage and order picking are usually the most costly. Storage is expensive because of the inventory holding costs, order picking because it is labor intensive. One approach to reduce

*Corresponding author

Email addresses: jan.vanbelle@mech.kuleuven.be (Jan Van Belle), paul.valckenaers@mech.kuleuven.be (Paul Valckenaers), dirk.cattrysse@cib.kuleuven.be (Dirk Cattrysse)

costs could be to improve one or more of these functions or to improve how they interact. Cross-docking however is an approach that eliminates the two most expensive handling operations: storage and order picking [3–6].

A definition of cross-docking provided by Kinnear [7] is: "receiving product from a supplier or manufacturer for several end destinations and consolidating this product with other suppliers' product for common final delivery destinations". In this definition, the focus is on the consolidation of shipments to achieve economies in transportation costs. The Material Handling Industry of America (MHIA) defines cross-docking as "the process of moving merchandise from the receiving dock to shipping [dock] for shipping without placing it first into storage locations" [8]. The focus is now on transshipping, not holding stock. This requires a correct synchronization of incoming (inbound) and outgoing (outbound) vehicles. However, a perfect synchronization is difficult to achieve. Also, in practice, staging is required because many inbound freight need to be sorted, consolidated and stored until the outbound shipment is complete. So, this strict constraint is relaxed by most authors. Cross-docking then can be described as *the process of unloading freight from inbound vehicles and loading these goods into outbound vehicles, with minimal handling and with little or no storage in between*. If the goods are temporally stored, this should be only for a short period of a time. An exact limit is difficult to define, but many authors talk about 24 hours. If the goods are placed in a warehouse or on order picking shelves or if the staging takes several days or even weeks, it is not considered as cross-docking but as (traditional) warehousing. However, even if the products are staged for a longer time, some companies still consider it cross-docking, as long as the goods move from supplier to storage to customer virtually untouched except for truck loading [1, 9]. Many organization use a mixture of warehousing and cross-docking to combine the benefits of both approaches [10].

Note that there is some confusion about the relation between cross-docking and flow-through distribution. Some authors consider it as identical (e.g [10]), while for others it are two different strategies (e.g. [11]).

A terminal dedicated for cross-docking is called a cross-dock. In practice, most cross-docks are long, narrow rectangles (I-shape), but other shapes are also used (L, T, X, . . .) [3]. 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 its appropriate 'stack door' and loaded on an outbound truck. Mostly, there is no special infrastructure to stage freight. If goods have to be stored temporarily, they are placed on the floor of the cross-dock (e.g. in front of the dock door where the departing truck is or will be docked). However, it is possible that the cross-dock contains for instance a pallet storage, certainly if cross-docking is combined with warehousing.

Figure 1 presents a schematic representation of the material handling operations at an I-shaped cross-dock with 10 dock doors. Incoming trucks are either directly assigned to a strip door or have to wait in a queue until assignment. Once docked, the freight (e.g. pallets, packages or boxes) of the inbound truck is unloaded and the destination is identified (e.g. by scanning the barcodes at-

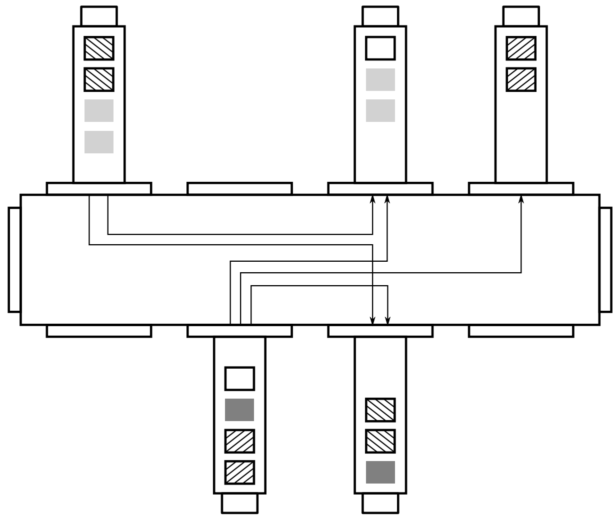

Figure 1: Material handling at a typical cross-dock.

tached to the goods). Then, the goods are transported to the designated stack door by some material handling device, such as a worker operating a fork lift or a conveyor belt system. There, the goods are loaded onto an outbound truck that serves the dedicated destination. Once an inbound truck is completely unloaded or an outbound truck is completely loaded, the truck is replaced by another truck.

Cross-docking corresponds with the goals of lean supply chain management: smaller volumes of more visible inventories that are delivered faster and more frequently [12]. In literature, several other (possibly intertwined) advantages of cross-docking are mentioned (e.g. [1, 4, 13–15]):

- Cost reduction (warehousing costs, inventory-holding costs, handling costs, transportation costs, labor costs)
- Consolidation of shipments
- Shorter delivery lead time
- Improved customer service
- Reduction of storage space
- Faster inventory turnover
- Fewer overstocks
- Improved resource utilization
- Reduced risk for loss and damage

- Better match between shipment quantities and actual demand
- Better control of the distribution operation

These advantages make cross-docking an interesting logistic strategy that can give companies considerable competitive advantages. Wal Mart is a well-known example [16], but also several other companies have reported the successful implementation of cross-docking (e.g. Eastman Kodak Co. [12], Goodyear GB Ltd. [7], Dots, LLC [17] and Toyota [9]).

Although cross-docking has already been applied in the 1980's (e.g. by Wal Mart), it has only attracted attention from academia much later and mostly during the recent years. For instance, more than 85 % of the academic articles found by the authors are published from 2004 on. During these years, a considerable amount of articles have been published and because of the growing interest from industry [1, 2], we expect that still more research on this topic will be performed the coming years.

The objective of this article is to present an overview of the cross-docking concept. First, guidelines for the successful use and implementation of cross-docking will be discussed. Further, several characteristics will be described to distinguish between different types of cross-docks. Next, the article will provide a review of the existing literature about cross-docking. The discussed articles are classified based on the problem type. These problems range from more strategical or tactical to more operational problems. This review can help (future) cross-docking practitioners to find the correct literature to start or improve their cross-docking operations. Without a proper implementation, it is impossible to benefit from the above-mentioned advantages. Based on the provided review, the authors try to identify gaps of knowledge and interesting areas for future research.

The term cross-docking usually refers to the situation in which trucks or trailers[1] are loaded and unloaded at a cross-docking terminal. However, the operations to handle freight at a harbor or airport are sometimes very similar. At a harbor for instance, containers are unloaded from a ship and temporarily placed onto the quay until they are loaded onto another ship or onto a truck. An airport can also be seen as a kind of cross-dock for transferring passengers and their baggage. This article focuses on the typical cross-docking in which goods are transferred between trucks at a cross-dock, but in literature several articles can be found that deal with similar problems specific for harbors or airports (e.g. how to determine the layout of an airport terminal [18, 19], how to assign airplanes to gates [20, 21], etc.).

To the best of our knowledge, only two articles present a review of cross-docking articles. Boysen and Fliedner [13] discuss articles about the truck scheduling problem and provide a classification of the considered problems. The approach taken here is however more general and several problem types related to cross-docking are discussed, among which the truck scheduling problem (see

[1] In the following pages, the terms truck, trailer and vehicle will be used interchangeably.

Section 4.4). Agustina et al. [22] provide a general picture of the mathematical models used in cross-docking articles. These models are classified based on their decision level (operational, tactical or strategic) and then subdivided by problem type. However, the authors do not completely agree with the proposed classification (the considered problem types and the assignment of articles to problem types), so another classification is presented in this article. Also, the review is more extensive; more articles are included and the articles are discussed in more detail. This article also includes a general overview of cross-docking and proposes a classification of cross-docks.

The article is organized as follows. The next section discusses in which situations cross-docking is a suitable strategy and deals with the requirements for a successful implementation. In Section 3, the characteristics are discussed that can be used to differentiate between alternative cross-docking systems. The literature review is presented in Section 4. The discussed articles are classified based on the problem type they deal with. The conclusions with opportunities to improve and extend the current research are summarized in Section 5.

2. When and how to use cross-docking?

Although cross-docking is nowadays used by many companies, it is probably not the best strategy in all cases and circumstances. In this section, we shortly describe existing literature that gives some guidelines for the successful use and implementation of cross-docking.

Apte and Viswanathan [10] discuss some factors that influence the suitability of cross-docking. A first important factor is the *product demand rate*. If there is an imbalance between the incoming load and the outgoing load, cross-docking will not work well. Hence, goods that are more suitable for cross-docking are the ones that have demand rates that are more or less stable (e.g. grocery and regularly consumed perishable food items). For these products, the warehousing and transportation requirements are much more predictable, and consequently the planning and implementation of cross-docking becomes easier. The *unit stock-out cost* is a second important factor. Because cross-docking minimizes the level of inventory at the warehouse, the probability of stock-out situations is higher. However, if the unit stock-out cost is low, the benefits of cross-docking can outweigh the increased stock-out cost, and so cross-docking can still be the preferred strategy. As shown in Figure 2, cross-docking is therefore preferred for products with a stable demand rate and low unit stock-out cost. The traditional warehousing is still preferable for the opposite situation with an unstable demand and high unit stock-out costs. For the two other cases, cross-docking can still be used when proper systems and planning tools are in place to keep the number of stock-outs to a reasonable level.

Some other factors that can influence the suitability of cross-docking are the distance to suppliers and customers (higher distances increase the benefits of consolidation), the product value and life cycle (a larger reduction in inventory costs for products with a higher value and shorter life cycle), the demand quantity (a larger reduction in inventory space and costs for products with a higher

Unit stock-out costs		*Product demand rate*: Stable and constant	*Product demand rate*: Unstable or fluctuating
	High	Cross-docking can be implemented with proper systems and planning tools	Traditional warehousing preferred
	Low	Cross-docking preferred	Cross-docking can be implemented with proper systems and planning tools

Figure 2: Suitability of cross-docking (adapted from [10]).

demand), the timeliness of supplier shipments (to ensure a correct synchronization of inbound and outbound trucks), etc. [10, 23, 24].

Some authors use a more quantitative approach to study the suitability of cross-docking. For instance, Galbreth et al. [4] compare the transportation and handling costs between a situation in which a supplier has to ship goods to several customers with only direct shipments and a situation in which also indirect shipments via a cross-dock are possible. For the second situation, a mixed integer programming (MIP) model is proposed to determine which goods should go directly from supplier to customer and which goods should be shipped via a cross-dock to meet the (known) demands. The transportation costs are modeled in a realistic way: fixed for truckload shipping, while the less-than-truckload shipping costs are modeled using a modified all-unit discount (MAUD) cost function. The holding costs at the customers are proportional to the quantity and the holding time between arrival time and due date. The costs for the two situations are compared under varying operating conditions. The authors conclude that cross-docking is more valuable when demands are less variable and when unit holding costs at customer locations are higher. On the other hand, it is less valuable when the average demands are close to truck load capacity.

Other quantitative approaches make a comparison between a situation with a cross-dock and a situation with a traditional distribution center. For instance, Kreng and Chen [25] compare the operational costs. Besides the transportation and holding costs, the production costs (more specific the setup costs) of the goods at the supplier are taken into account. When a cross-dock is used, more frequent deliveries to the cross-dock are required and the batch size needs to be smaller, which causes higher setup costs. Waller et al. [26] look to both situations from an inventory reduction perspective.

Schaffer [6] discusses the successful implementation of cross-docking. When a company wants to introduce cross-docking, the introduction should be prepared very well. If the necessary equipment is already available and because cross-docking seems simple, one easily assumes that cross-docking can be implemented without much effort. However, cross-docking itself is quite complex and requires a high degree of coordination between the supply chain members (e.g. the timing of arrival and departure). So, the requirements for successful cross-docking should be understood thoroughly and the implementation should be planned carefully. Following Schaffer, the requirements for a successful implementation fall into six categories.

Partnershipping The implementation of cross-docking at one member of the supply chain results frequently in increased cost and effort for other parts. So, the different partners should work together to minimize these additional costs and the partner that implements cross-docking should compensate the other partners for any (reasonable) remaining additional cost.

Confidence For successful cross-docking, it has to be absolutely certain that the correct products with the required quality are available when needed. Therefore, the different parties should agree on specifications for all appropriate requirements and should conduct a test program to check the ability of all parties to meet the requirements.

Communication between supply chain members Since cross-docking is a real-time operation, information (e.g. shipping time and quantity, scheduled arrival time) must be available quickly. This is possible by communicating via electronic data interchange or similar techniques.

Communication and control Once material arrives, it must move without interruption through the facility. A warehouse management system (WMS) is needed to accomplish this task.

Personnel, equipment and facilities By introducing cross-docking, costs for storing and picking are greatly reduced, but the requirements for receiving and shipping increase. Sufficient resources should be allocated to ensure that all requirements are met.

Tactical management Cross-docking requires a high level of tactical execution to work properly. It is necessary that enough resources are provided to perform the supervisory task well.

According to Witt [9] and to Yu and Egbelu [27], software (e.g. a WMS) plays an important role in the successful implementation of cross-docking. The required (automated) hardware for a cross-docking system might come off the shelf and is easily available today. But the software needs to be tailored to the specific requirements and is in general relatively less developed, although it is as important as hardware to cross-docking success. This is also confirmed by a survey among professionals who are involved in cross-docking and who denote IT system support as a key barrier to effective cross-docking [1, 2]. Hence, the system requirements need to be carefully defined and studied in order to prevent installing the physical system to discover afterwards there is no information and communication system in place for successful operation.

This software system can only work correctly if it is fed with accurate and timely information. Compared to regular distribution, the information flow to support cross-docking is significantly more important [24]. For instance, to coordinate the inbound and outbound trucks to the appropriate docks, the arriving

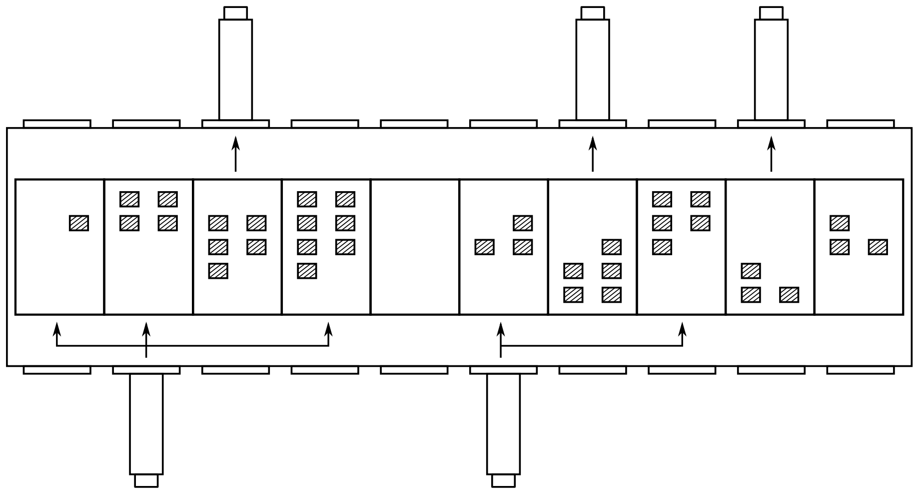

Figure 3: A single-stage cross-dock in which the products are staged in zones corresponding to the stack doors (adapted from [28]).

time and the destination of the freight needs to be known before the physical arrival of the goods (e.g via advance shipping notice (ASN)). Several information technology tools are available to realize this information flow, e.g. electronic data interchange (EDI), shipping container marking (SCM), bar-coding and scanning of products using universal product code (UPC) [10]. Regardless of which technology is chosen, the supply chain partners must be able and willing to deliver the required information via this technology. A good cooperation across the supply chain can make or break the cross-docking implementation [6, 9, 24].

3. Cross-dock characteristics

Several characteristics can be considered to distinguish between various types of cross-docks (and cross-docking). A common distinction made in literature is based on the number of touches [1] or stages [28]. In *one-touch* cross-docking, products are touched only once, as they are received and loaded directly in an outbound truck. This is also called *pure* cross-docking [10, 29]. In a *two-touch* or *single-stage* cross-dock, products are received and staged on the dock until they are loaded for outbound transportation. Usually, the goods are put into zones corresponding to their strip or stack door (see Figure 3). In the case of a *multiple-touch* or *two-stage* cross-dock, products are received and staged on the dock, then they are reconfigured for shipment and are loaded in outbound trucks. In a typical configuration, the incoming freight is first put in zones corresponding to the strip doors. The goods are then sorted to the zones corresponding to the stack doors (see Figure 4).

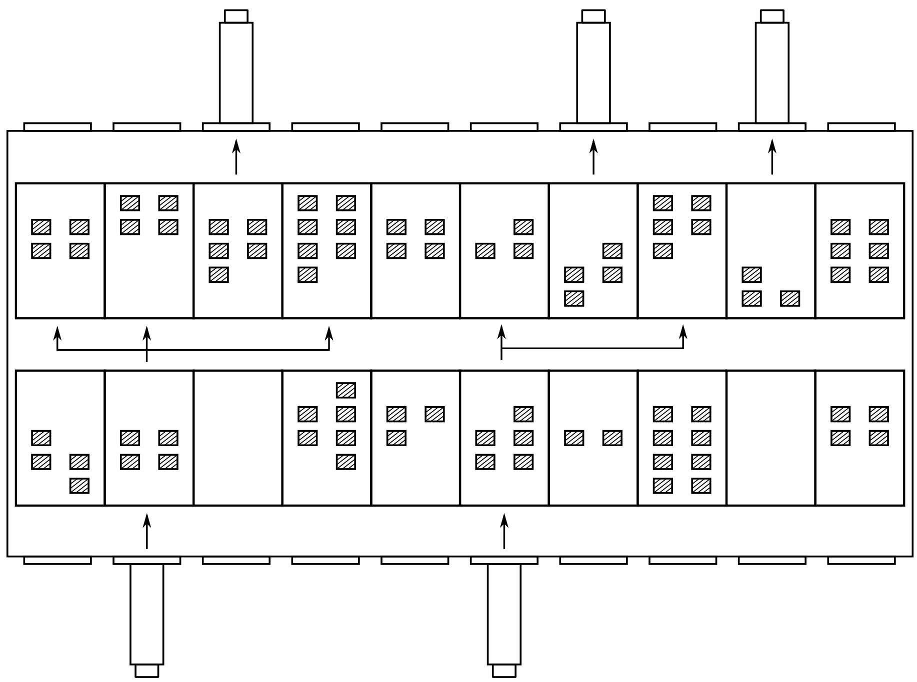

Figure 4: A two-stage cross-dock in which the products are staged in zones corresponding to the strip and stack doors and are sorted in between (adapted from [28]).

Another distinction can be made according to when the customer is assigned to the individual products [30]. In *pre-distribution* cross-docking, the customer is assigned before the shipment leaves the supplier who takes care of preparation (e.g. labeling and pricing) and sorting. This allows faster handling at the cross-dock. On the other hand, in *post-distribution* cross-docking, the allocation of goods to customers is done at the cross-dock.

Still some other distinctions are possible. The German supermarket retailer Metro-AG for instance distinguishes *source-oriented* and *target-oriented* cross-docking based on the location of the cross-dock terminals relative to suppliers and customers [31]. Napolitano [32] distinguishes five types of cross-docking based on the intented use: *manufacturing*, *transportation*, *distribution*, *retail* and *opportunistic* cross-docking. In [29], even eight types of cross-docking are distinguished.

In this section, several characteristics are described that can be used to distinguish between different cross-dock types[2]. For the articles discussed in Section 4, the characteristics of the considered cross-docks will be summed up according to the characteristics described here[3]. These characteristics can be divided in three groups: physical characteristics, operational characteristics and characteristics about the flow of goods[4]. In the next sections, these groups will be described in more detail.

3.1. Physical characteristics

The physical characteristics are characteristics of the cross-dock that are supposed to be fixed (for a rather long time). We consider the following physical characteristics.

Shape. Cross-docks can have a large variety of shapes. The shape can be described by the letter corresponding to the shape: *I, L, U, T, H, E, ...*

Number of dock doors. A cross-dock is also characterized by the number of dock doors it has. In practice, cross-docks range in size from 6-8 doors to more than 200 doors, and even a cross-dock with more than 500 doors exists [33]. In literature, sometimes the number of dock doors is limited to only 1 or 2. In these cases, the idea is not to model a realistic cross-dock, but to gain some insight by studying a simplified model.

---

[2]Some of the characteristics described here are similar to the characteristics used by Boysen and Fliedner [13] to make a classification of truck scheduling problems. However, Boysen and Fliedner consider not only real world characteristics, but also characteristics of the (mathematical) models.

[3]At least for the articles in which the internal details of the cross-dock are considered (Section 4.3, 4.4, 4.5 and 4.8).

[4]This classification is rather vague. For some characteristics, it is not clear in which group they fit best or it is clear that they belong to multiple groups. For instance, the fact that freight is temporarily stored can be seen as a physical, operational or flow characteristic.

Internal transportation. The transportation inside the cross-dock can be executed *manually* (e.g. by workers using forklifts) or there can be an *automated* system in place (e.g. a network of conveyor belts). The available infrastructure will of course be dependent on the type of freight that is handled in the cross-dock. For instance, LTL carriers handle mostly palletized freight and so make use of forklifts. Conveyor systems on the other hand are among others used by parcel carriers, as they deal with many (small) packages.

Temporary storage. In pure cross-docking, the arriving freight is directly transported to outbound trucks, so no storage is needed. In practice however, this is rarely the case. In general, the goods are temporarily stored on the floor of the cross-docking terminal (e.g. in front of the stack doors) or even in a (small) warehouse. However, because of space constraints, it can be the case that it is not possible to store freight intermediately.

3.2. Operational characteristics

Some operational decisions can influence the functioning of the cross-dock. These operational constraints lead to the following characteristics.

Service mode. According to Boysen and Fliedner [13], the service mode of a cross-dock determines the degrees of freedom in assigning inbound and outbound trucks to dock doors. In an *exclusive* mode of service, each dock door is either exclusively dedicated to inbound or outbound trucks. If this service mode is used, mostly one side of the cross-docking terminal is assigned to inbound trucks and the other side to outbound trucks. A second mode is *mixed* mode. In this mode, inbound and outbound trucks can be processed at all doors. These two modes can also be combined. In this *combination* mode, a subset of doors is operated in exclusive mode while the rest of the doors is operated in mixed mode.

Pre-emption. If pre-emption is allowed, the loading or unloading of a truck can be interrupted. This truck is then removed from the dock and another truck takes its place. The unfinished truck has to be docked later on to finish the loading or unloading.

Temporary storage. Temporary storage can also be seen as an operational characteristic. In many cases it is allowed to store goods temporarily if it is possible. But even if it is possible, it can be an operational decision that storage is not allowed and products need to be immediately transferred, for instance to avoid congestion inside the cross-dock.

3.3. Flow characteristics

The characteristics of the flow of goods that have to be processed by a cross-dock can be very different. We consider the following characteristics.

Arrival pattern. The arrival times of the goods are determined by the arrival times of the inbound trucks. The arrival pattern can be *concentrated* at one or more periods if the inbound trucks arrive together at (more or less) the same times. For instance, a cross-dock in the LTL industry serving a certain geographical area usually receives freight at two periods. Goods that have to be transported from inside that area to another area are picked up during the day and all pickup trucks arrive in the evening at the cross-dock. The goods are then sorted during the night and the outbound trucks leave in the morning. To simplify the problem, several articles assume that the inbound trucks arrive together (at the beginning of the time horizon). On the other hand, freight from outside the region but destined for that area arrives in the early morning and is then distributed during the day. Another possibility is that the arrival pattern is *scattered* and the inbound trucks arrive at different times during the day. The arrival pattern has an influence on the congestion of the cross-dock and on the scheduling of workers and resources.

Departure time. The departure times of the trucks can be restricted or not. In many cases there are *no* restrictions and the trucks leave the cross-dock after all freight is loaded or unloaded. However, it is also possible that the trucks have to depart before a certain point in time, for instance in order to be on time for a next transportation task. In this case, there can be restrictions imposed on the departure times of the *inbound* trucks only, so that these trucks have to be unloaded on time. In a similar way, it is possible that only the *outbound* trucks have to leave the cross-dock before a certain moment[5]. For instance, in the parcel delivery sector, the outbound trucks usually leave at a fixed point in time. Parcels arriving late have to wait until another truck departs for the same destination. It is also possible that *both* inbound and outbound trucks have restricted departure times.

Product interchangeability. The freight handled at a cross-dock is in general not interchangeable. In this case, all products are dedicated to a specific *destination*[6] or a specific outbound *truck* (pre-distribution). Information about the destination or the dedicated truck is normally known before the products arrive at the cross-dock. It is however also possible that interchangeability of products is *allowed* (post-distribution). In this situation, only the type of products to be loaded on the outbound trucks and the corresponding quantity is known[6]. When the products are interchangeable, usually some value-added activities (e.g. labeling) need to be performed.

Temporary storage. Cross-docking aims to avoid storage, but in practice goods can be temporarily stored. However, it is also possible that goods are not allowed to be stored and have to be transported directly from inbound to outbound

[5] This point in time can be dependent on the (due dates of the) actual load of the truck.

[6] The assignment of the products to a specific outbound truck is then an operational decision.

truck. For instance, if refrigerated products have to be cross-docked in a non-cooled terminal, these products have to be directly moved to a cooled outbound truck.

4. Literature review

Cross-docking practitioners are confronted with many decisions they have to take during the design and operational phase of cross-docks. For instance, cross-dock managers have to decide how to assign arriving trucks to dock doors, how to optimize the workforce, which staging strategy will be used, how the trucks have to be loaded, etc. These decisions can have a serious impact on the efficiency, so they have to be carefully taken. In literature, several decision problems are studied. Some of these problems are more concerned about decisions with effects on a longer term (strategical or tactical), while others deal with short-term decisions (operational). This section gives a review of the existing literature about cross-docking problems[7].

First, some existing literature about tactical/strategical decisions is discussed: where will a cross-dock (or cross-docks) be located and what is the best layout of a cross-dock. Operational problems tackled in literature deal with the assignment of trucks to dock doors or the scheduling of the trucks, and the location where goods will be temporarily stored. Some authors also consider operational issues at the supply chain level and consider vehicle routing or try to optimize the flow of goods through a network of cross-docks. Finally, some articles that study other issues related to cross-docking are discussed.

4.1. Location of cross-docks

The location of one or more cross-docks is part of the design of a distribution network or supply chain. An important strategical/tactical decision that has to be made concerns the position of these cross-docks. This problem cannot be handled isolated from the decisions that determine how the goods flow through this network. The determination of the flow of goods is discussed in Section 4.7, but problems that also involve a decision about the location are considered here. The problem where to locate facilities (e.g. distribution centers or plants) has attracted a considerable amount of attention[8]. The articles discussed here regard facilities which are considered as cross-docks.

A first study about the location of cross-docks is performed by Sung and Song [34]. In the considered problem, goods have to be transported from supply to demand nodes via a cross-dock (direct shipments are not allowed). The cross-dock can be chosen from a set of possible cross-dock locations, each with an associated fixed cost. The demands are assumed to be known and there are two types of vehicles with a different capacity and cost. The aim is to

[7] Cross-docking practitioners are also confronted with problems that are not specific for cross-docking (e.g. the packing of loads inside trucks). These problems are not considered.

[8] Several references can be found in the articles discussed in this section.

find which cross-docks should be used and how many vehicles are needed on each link in order to minimize the total cost. This total cost consists of the fixed costs of the used cross-docks and the transportation costs. The authors present an integer programming model of the problem. This model is very similar to the model of Donaldson et al. [35] and Musa et al. [36] (discussed in Section 4.7) and similar simplifying assumptions are applied. The difference is the addition of the location decision and the fact that direct shipments are not considered. Because the problem is NP-hard, tabu search is proposed to solve the problem. The solutions determine how the goods flow through the network. Based on this flow, the number of vehicles can be derived by solving a subproblem. Some computational experiments are performed on generated test instances and indicate that the proposed algorithm finds good feasible solutions within a reasonable time.

Sung and Yang [37] extend this work and propose a small improvement to the tabu search algorithm. The authors also present a set-partitioning-based formulation of the problem and propose a branch-and-price algorithm based on this formulation to obtain exact solutions. The computational results show that this algorithm gives better results in terms of the number of (small-scale) problem instances solved and the required computation time compared with the results obtained by solving the integer programming model with the optimization software package CPLEX.

Gümüs and Bookbinder [38] study a similar problem, but now direct shipments are allowed and multiple product types are considered (multicommodity). The facility cost for each cross-dock consists of a fixed cost and a throughput cost charged per unit load. The transportation cost also has two components: a fixed cost for each truck and a variable cost per unit load per unit distance. A last cost that is taken into account is the cost for in-transit inventory. In this approach, the synchronization of inbound and outbound trucks is not taken into account. The authors provide a mixed integer programming model of the problem. By solving several smaller problem instances optimally (with the optimization software packages LINGO and CPLEX), the influence of several cost parameters is studied. Concerning the location problem, the authors conclude that the optimal number of cross-docks is an increasing function of the ratio between the (fixed) truck cost and the (fixed) facility cost.

A different approach is taken by Jayaraman and Ross [39]. They study a multi-echelon problem in which goods (from multiple product families) have to be transported from a central manufacturing plant to one or more distribution centers. From there, the goods are moved via cross-docks to the customers. The problem is tackled in two stages. In the first stage, a strategic model is used to select the best set of locations for the distribution centers and cross-docks. The model has to be run any time there is a need to change the network design. The authors provide an integer programming formulation that aims to minimize the fixed costs associated with operating open warehouses and cross-docks and the transportation costs from the warehouses to the cross-docks and from the cross-docks to the customers. Demand splitting is not allowed: customers have to be assigned to single cross-docks while cross-docks have to be assigned to single warehouses only. In the second stage, an operational model decides upon the quantities of each product type that need to be transported to the customers via distribution centers and cross-docks. The model tries to minimize the transportation costs while satisfying customer demand. This model is less restrictive than the first model (it relaxes for instance the assumption that cross-docks need to be assigned to only one warehouse) and can be executed once the number of warehouses and cross-docks are known from solving the first model. Both models are more simplified compared to the previous approaches. For instance, individual vehicles are not considered and the transportation cost is proportional to the quantity to ship. Solving the developed models optimally is only possible for small problem instances. Therefore, the authors propose a simulated annealing approach. The computational experiments on generated problem instances indicate that the heuristic is able to give results with a deviation of about 4 % of the optimal solution (obtained with LINGO), but 300 to 400 times faster.

In [40], the same authors present two other heuristics to tackle the problem. Both heuristics are based on simulated annealing but use an extra mechanism to avoid locally optimal solutions. The first heuristic makes use of a tabu list, the second heuristic allows a sudden re-scaling of the 'system temperature'. For both heuristics, the solution quality and computational performance are tested for different 'cooling schemes'. The experimental results indicate that the simulated annealing heuristic combined with tabu search gives better solutions in slightly more time.

Bachlaus et al. [41] also consider a multi-echelon supply chain network, including suppliers, plants, distribution centers, cross-docks and customers. The goal is to optimize the material flow throughout the supply chain and to identify the optimal number and location of suppliers, plants, distribution centers and cross-docks. The problem is formulated as a multi-objective optimization model that tries to minimize the total cost and to maximize the plant and volume flexibility. Because of the computational complexity of the problem, the authors propose a variant of particle swarm optimization (PSO) to design the supply chain. Some computational experiments are conducted and the results show that the proposed solution approach gives better results than a genetic algorithm and two other PSO variants.

4.2. Layout design

Once the location of the cross-dock is determined, another strategical/tactical decision that has to be made is to choose the layout of the cross-dock. The layout is interpreted as the dimension and shape of the cross-dock, as well as the dimension and shape of the internal cross-dock areas and their arrangement.

Bartholdi III and Gue [3] focus on the shape of a cross-dock. Most existing cross-docks are long, narrow rectangles (I-shape), but there are also cross-docks shaped like an L, U, T, H or E. The cross-dock shape is sometimes determined by simple constraints (e.g. size and shape of the lot on which it will stand), but in this article the focus is on how the shape affects cross-dock performance. Several experiments are performed in which the labor costs (estimated by the total travel time[9]) are measured for different shapes. The experiments suggest that an I-shape is the most efficient for smaller cross-docks (fewer than about 150 doors). For docks of intermediate size, a T-shape is best and for more than 200 doors (approximately) an X-shape is best. Cross-docks with a T or X-shape have a greater 'centrality'. However, they achieve this at the cost of additional corners which reduce the labor efficiency (2 inside and 2 outside corners for T, 4 inside and 4 outside corners for X). An inside corner renders some doors unusable, while doors around an outside corner have less floor space available to stage freight. So, these additional corners are a fixed cost, which begins to pay of for larger docks. It is however not always easy to predict which shape is better, because this also depends on e.g. the freight flow pattern.

Other articles deal with the design of the storage area where the freight can be temporarily staged (on the floor or in racks). In many cases, the freight is placed in several parallel rows and the workers can move between these rows. Vis and Roodbergen [15] deal with the operational decision where to temporarily store incoming freight (see Section 4.5). The proposed algorithm can also be used during the design phase to determine the optimal number of parallel storage rows and their lengths.

The (single-stage or two-stage) storage area can also be organized in parallel lanes directly next to each other that can only be accessed at both ends. Gue and Kang [28] make use of simulation to study the behavior of these so-called staging queues. The results suggest that, for a single-stage storage area, it is better to have more short lanes than fewer long ones, at least when the workers follow a rational approach. The results also indicate that two-stage cross-docking has a significantly lower throughput than single-stage cross-docking.

4.3. Dock door assignment

If the location and the layout of the cross-dock are set, the good functioning of the cross-dock will be determined by the operational decisions. A first operational decision has to be taken when an inbound or outbound truck arrives at the cross-dock: at which dock door should it be assigned? A good assignment can increase the productivity of the cross-dock and can decrease the (handling) costs. So, the *dock door assignment problem* tries to find the 'optimal' assignment of inbound and outbound trucks to dock doors. It is assumed that there are at least as much dock doors as trucks, so each truck will be assigned to a different door and time aspects are not taken into account. If this condition is not fulfilled, the dock doors can be seen as (scarce) resources that have to be scheduled over time. This is the so-called *truck scheduling problem*. Both problems can be quite complex due to the number of doors and the dynamic nature of the problem. This section deals with the dock door assignment problem, while truck scheduling problems are discussed in Section 4.4.

[9]Here and in the following pages, the travel time is the time required to transfer the goods internally from the inbound to the outbound truck.

The assignment of dock doors can be executed on a mid-term or short-term horizon [13]. Several articles solve the assignment problem on a *mid-term* horizon. Then, each dock door serves a specific inbound or outbound destination for a longer period of time (e.g. 6 months)[10]. All trucks coming from the same origin or having the same destination are assigned to the same dock. Such a fixed assignment is more easy for workers because they know exactly to which dock door they need to ship each load, but it comes at the expense of a reduced flexibility. Even if a fixed assignment is used, it is important that the dock doors are reassigned when there is a significant change in the shipping pattern.

When data about the inbound trucks is known far enough in advance, the assignment of the trucks can be solved on a *short-term* horizon. The trucks itself are assigned to the dock doors based on the actual freight flow. This 'floating dock' concept is put forward by Peck [42] who studied the material handling operations in an LTL terminal. Such an assignment implies that the workers are every day confronted with a different door for the same destination and have to take care that the freight is loaded into the correct truck. The use of modern information technology (e.g. bar code or RFID scanning together with a WMS) can be useful for this end.

A combination of both is also possible. Several articles consider a cross-dock in which destinations are assigned to stack doors (so the outbound trucks are assigned on a mid-term horizon), while the assignment of the inbound trucks is done on a short-term horizon.

The characteristics of the cross-docks considered in the following articles are summarized in Table 1. As time aspects are neglected and there are enough available dock doors, the pre-emption, arrival pattern and departure time characteristic are not relevant here and are not shown.

In his dissertation, Peck [42] develops a detailed simulation model of an LTL terminal and tries to assign the trucks to dock doors in order to minimize the travel time of the shipments. It is assumed that the travel time to transport the products between two trucks can be expressed as a function of the distance, based on the actual contents of the trucks and the required means of transport (2-wheeler, 4-wheeler or forklift). The designation of doors as either strip or stack doors is fixed beforehand. The problem is formulated as an integer programming model and because of the computational complexity, a heuristic (greedy balance algorithm) is provided to solve it. Simulation shows that his heuristic improves an assignment based on experience and intuition.

Another early study about the assignment of trucks to dock doors is performed by Tsui and Chang [43]. In this article, a cross-dock is considered in which no storage is provided; all shipments go directly from inbound to outbound trucks. The problem is solved on a mid-term horizon, so the origins and destinations have to be assigned to dock doors, not the trucks itself. The designation of doors as strip or stack doors is fixed. The assignment problem is formulated as a bilinear programming problem that tries to minimize the travel

[10]This includes that the cross-dock operates in exclusive service mode.

	Shape	No. of doors	Internal transport	Temporary storage	Service mode	Interchange-ability
Peck [42]	I	*	manually	yes	exclusive	truck
Tsui and Chang [43, 44]	I	*	manually	no	exclusive	destination
Bermúdez and Cole [45]	*	*	manually	ns	exclusive	destination
Cohen and Keren [46]	I	*	manually	ns	exclusive	destination
Oh et al. [47]	I	*	manually	ns	exclusive	destination
Bartholdi III and Gue [48]	I	*	*	yes	exclusive	destination
Gue [33]	I	*	manually	ns	exclusive	destination
Brown [49] (semi-permanent)	*	*	manually	yes	exclusive	destination
(dynamic)	*	*	manually	yes	mixed	truck
Bozer and Carlo [50] (semi-permanent)	*	*	manually	ns	exclusive	destination
(dynamic)	*	*	manually	ns	mixed	truck
Yu et al. [51]	*	*	manually	ns	exclusive	destination

Table 1: Characteristics of the articles discussed in Section 4.3. A '*' indicates that not a single value of the characteristic is valid, but that all values can be used, 'ns' indicates that a characteristic is not specified.

distance[11] the forklifts (the number of forklift trips required to carry a certain load is assumed to be known). To solve it, the authors propose a simple heuristic method to find a local optimum. The authors do not provide test results, but conclude that the found solution can serve as a good starting point for the cross-dock manager.

There exist exact algorithms to solve bilinear optimization problems, but these are not very suited for this problem as the same authors mention in [44]. In this article, a branch-and-bound algorithm is proposed to solve the dock door assignment problem exactly. The numerical tests show that this algorithm is however computational expensive.

Bermúdez and Cole [45] deal with a very similar problem, but now there is no fixed designation for the doors. All doors can have assigned either an origin or a destination. The mathematical model of Tsui and Chang [43] is adapted to take this into account. The objective function minimizes the total weighted travel distance instead of the real travel distance. Because the problem is NP-hard, a genetic algorithm (GA) is proposed to solve it. Based on data from an LTL carrier, the authors study the impact of different GA parameters on the solution and compare the results of the genetic algorithm to the results obtained with a pair-wise exchange technique (2-opt). The genetic algorithm seems to give comparable or slightly better results.

Cohen and Keren [46] also extend the approach of Tsui and Chang [43]. The mathematical model is adapted to allow that freight for a certain destination can be split and delivered to multiple doors assigned to that destination (the capacity of the outbound trucks is taken into account). The proposed formulation is a non-linear MIP model that is impractical for real size problems. So, the authors propose a heuristic algorithm to solve it. Because of its simplicity, the heuristic can be easily recalculated to adapt to small changes in the freight flow pattern. It is however not clear how well this heuristic performs.

A different assignment problem is considered by Oh et al. [47]. This article deals with cross-docking in a mail distribution center in which the different doors (and corresponding destinations) are clustered into groups. Each group has a shipping area located at the center of its stack doors. Arriving products are transported from the inbound trucks to these shipping areas, sorted according to their destination and loaded into outbound trucks. When a large amount of freight has to be shipped to a destination, this destination can be assigned to several stack doors. The objective is to find an assignment of destinations to stack doors and a clustering of destinations in groups that minimizes the total travel distance. So, the assignment of strip doors is not considered, and the assignment of stack door is solved on a mid-term horizon. The authors present a non-linear programming model of the problem and propose two heuristic methods to solve it: a decomposition heuristic and a genetic algorithm. Based on

[11] Here and in the following pages, the travel distance is the distance travelled (by workers, forklifts, ...) in order to transfer the goods internally from the inbound to the outbound truck.

data obtained from the mail distribution center, the computational results indicate that both heuristics can reduce the travel distance compared with the current situation (about 13 % for the decomposition heuristic and about 9 % for the genetic algorithm).

Bartholdi III and Gue [48] define the *layout* of a cross-dock as the specification of doors as either strip or stack doors and the assignment of destinations to stack doors. It is assumed that the assignment of inbound trucks to strip doors happens in real time by a dock supervisor using a first-come, first-served (FCFS) policy. The flow through each strip door then tends, over time, to resemble the aggregate flow through the terminal, so each inbound trailer is modeled as an 'average trailer'. The objective of this article is to determine an 'optimal' layout. So, this article deals with the mid-term assignment of outbound trucks, while the short-term assignment of inbound trucks is not considered and there is no fixed designation for the doors. In the previous approaches, the objective is the minimization of travel distance. According to the authors however, approaches to determine an optimal layout based on travel distance are inaccurate. The travel time should be taken into account, and the travel distance is not a good measure of travel time. The actual travel time also depends on the type of freight, the used material handling system and congestion. Minimizing the travel distance can even worsen congestion. In this article, a (non-linear) mathematical model is described which can take different types of material handling systems into account and which uses models of different types of congestion. The model tries to minimize the total labor cost, which accounts for both travel costs (based on travel time) and congestion costs (based on waiting times due to congestion). The authors use a simulated annealing procedure that swaps pairs sequentially to solve the assignment problem. Based on results obtained with the developed model, the authors formulate some guidelines for efficient layouts. For instance, it is interesting to alternate high-flow stack doors with strip doors at the center of the cross-dock to reduce travel time and congestion. The proposed method was used to improve the layout of an existing cross-dock and the authors report that labor productivity increased 11.7 % according to the company's measurements.

In the article, it is assumed that the freight flows from strip doors to destinations are known and independent of the layout. This is modeled by placing an 'average trailer' at each strip door. However, when the cross-dock supervisor assigns incoming trailers to doors in real time based on the contents of the trailers and the location of the doors ('look-ahead scheduling' instead of FCFS), the material flows are altered and dependent on the layout. In [33], Gue examines the effect of look-ahead scheduling on the material flows and the layout of the cross-dock. To determine the layout with the lowest labor cost, the author proposes to search the solution space of all layouts with a local search algorithm (that swaps pairs of trucks). For a given layout, the labor costs can be determined if the resulting material flows are known (only travel costs are considered, no congestion costs). To model these flows, 'biased trailers' are constructed by solving a linear programming problem. Such a trailer contains freight that is biased toward the destinations that are closest to the strip doors to which it is assigned. The author proposes a specific look-ahead algorithm[12] to test the solutions using simulation. The simulation results indicate that it is possible to save 15-20 % in labor costs by using this look-ahead scheduling policy for the inbound trucks. Extra costs can be saved by constructing the layout of the terminal based on the altered flows (at least if the average number of destinations per inbound truck is low).

So, Gue determines the layout (that changes only periodically) and assumes that the inbound trucks will be scheduled using a real time policy. Another possibility is to determine the layout together with the short-term assignment of inbound trucks, i.e. the assignment of the inbound trucks itself and not the origins to dock doors. This is what Brown calls a *semi-permanent layout*. In her master thesis, Brown [49] studies the problem of assigning trucks to dock doors (trailer-to-door assignment problem or hub layout problem) and how to unload the inbound trucks (freight sequencing problem). For the *trailer-to-door assignment problem*, the objective is to minimize the total travel distance. A semi-permanent layout is constructed in two phases. The first phase allocates dock doors as strip or stack door and also assigns destinations to the stack doors. Similar to Bartholdi III and Gue [48], 'average trailers' are used as inbound trailers. Starting from an initial assignment, a local search is performed with pair-wise exchanges of trucks to generate a final solution. In the second phase, the inbound trucks are assigned to strip doors. Pair-wise exchanges of inbound trucks are used to improve an initial assignment. Brown also considers a *dynamic layout* in which both the inbound and outbound trucks are assigned on a short-term horizon. Again, an initial assignment is improved by pair-wise exchanges of trucks. The experimental results (based on actual shipment data) indicate that the dynamic layout reduces the total travel distance significantly compared with the semi-permanent layout.

In [50], Bozer and Carlo also consider a semi-permanent and a dynamic layout. To determine the assignment of outbound trucks for the semi-permanent layout, the solution space of possible assignments is searched as done by Gue [33] and Brown [49], but simulated annealing is used instead of local search. Also different is that no 'average' or 'biased' inbound trucks are assumed, but actual data of several assignment periods is used. For a given outbound door assignment, the optimal inbound trailer-to-door assignment and the corresponding travel distance are then determined by solving a linear assignment problem (for which efficient algorithms exist). This is done multiple times, for each assignment period, and the sum of the resulting distances is taken. In this way, the variability in freight flow is taken into account. Once the optimal assignment of outbound trucks is determined, the short-term assignment of inbound trucks can be found by again solving the linear assignment problem. For the dynamic layout, the authors model the problem as a quadratic assignment problem (QAP) with rectilinear distances and present a MIP formulation. However, to solve large problem instances, the authors propose again to use simulated annealing,

[12] Wang and Regan [52] propose two alternative scheduling policies (see Section 4.4).

but now with the actual content of the inbound trucks. The proposed model tries to minimize travel distance, and congestion is not taken into account. The authors suggest that congestion can be avoided by not allowing solutions that have three outbound trucks assigned adjacently. The results of numerical experiments indicate that simulated annealing gives better results than a pairwise exchange steepest descent heuristic, which is known to perform well for solving a rectilinear QAP. Also, the dynamic layout seems to give slightly better results than the semi-permanent layout.

Yu et al. [51] also consider a semi-permanent layout. The objective is to minimize the total travel time. To determine the short-term assignment of inbound trucks, an online policy (different from FCFS) is proposed that assigns arriving inbound trucks on a real-time basis[13]. This policy is however myopic. It only guarantees to minimize the processing time of the considered inbound truck, but it may worsen the processing time of future arriving trucks. The designation of doors as strip or stack door and the mid-term assignment of the outbound trucks is found by solving the *destination-door allocation problem* (DDAP). The objective function of this problem is the expected value of the travel time with respect to several representative scenarios. A scenario describes the arrival times and the contents of the inbound trucks and is based on actual data instead of averages. In this way, the variability in freight flow is taken into account in a similar way as by Bozer and Carlo [50]. The applied on-line policy is also taken into account by the objective function. Two heuristics are provided to solve the DDAP: a local search heuristic and a genetic algorithm. The authors performed a computational study using simulated data patterned after actual data. The results show that both heuristics can reduce the total travel time with about 20 % compared to current practice.

4.4. Truck scheduling

In the previous section, the assignment of trucks to dock doors was discussed. Temporal constraints were not taken into account; it was not possible to assign multiple trucks to the same door sequentially. The *truck scheduling problem* on the other hand considers the dock doors as resources (used by the trucks) that have to be scheduled over time. The problem decides on the succession of inbound and outbound trucks at the dock doors of a cross-dock: *where and when* should the trucks be processed.

In fact, the assignment problem is part of the truck scheduling problem. As mentioned in the previous section, this assignment can be executed on a short-term or mid-term horizon. Usually, the truck scheduling problem assigns the trucks to dock doors on a short-term horizon. In this case, trucks with the same origin or destination can be assigned to different dock doors. However, for the mid-term assignment, it are the origins and destinations that are assigned

[13]Because time aspects are taken into account, this can in fact be considered as (dynamic) scheduling of the inbound trucks.

to doors and the truck scheduling problem reduces to sequencing all trucks of equal origin or destination.

This section discusses articles that deal with the truck scheduling problem. The characteristics of the cross-docks considered in the these articles are summarized in Table 2. The articles are also classified according to the classification scheme for deterministic truck scheduling problems proposed by Boysen and Fliedner [13] (see Table 3). The classification is based on three basic elements of any truck scheduling problem which are noted as a 'tuple': the 'door environment', operational characteristics and the objective. For each of these three main elements, several attributes are specified. 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), intermediate storage (allowed or not), etc.

4.4.1. Single strip and stack door

Several authors consider a simplified cross-dock with a single strip and a single stack door to study the truck scheduling problem. While this is not a realistic case, it can provide insights that are helpful for more complex cross-docks. Truck scheduling reduces in this case to the sequencing of the inbound and outbound trucks.

Chen and Lee [53] consider the so-called *two-machine cross-docking flow shop problem*. The objective is to sequence the inbound and outbound trucks in order to minimize the makespan, i.e. the time span from the start of the unloading of the first inbound truck until the end of the loading of the last outbound truck. The problem is modeled as a two-machine flow shop problem, but with additional precedence constraints to make sure that an outbound truck cannot be processed (on the second machine) before all its predecessor tasks have been completed (on the first machine). The load and unload times can be different for each truck (e.g. based on the actual content) and can possibly include the travel time. Pre-emption is not allowed and it is assumed that all trucks are available at the beginning of the planning horizon. Unloaded products can be temporarily put in storage (with infinite capacity) until the appropriate outbound truck is docked. The authors prove that this problem is strongly NP-hard and present a heuristic approach based on Johnson's rule (which solves the two-machine flow shop problem). A branch-and-bound algorithm to solve the problem optimally is also provided. Computational results show that the branch-and-bound algorithm can solve problems with up to 60 trucks in a reasonable amount of time.

Chen and Song [14] extend this problem to the *two-stage hybrid cross-docking scheduling problem*. Now multiple trucks can be loaded or unloaded at the same time by considering parallel machines at the inbound and outbound 'stage'. The travel time between the inbound and outbound docks is not taken into account. The authors provide a mixed integer programming model of this problem and propose several heuristics based on Johnson's rule to solve it.

Yu and Egbelu [27] also study a cross-dock with a single strip and a single stack door. Similar to the two-machine approach, the objective is to minimize

	Shape	No. of doors	Internal transport	Temporary storage	Service mode	Pre-emption	Arrival pattern	Departure time	Interchange-ability
Chen and Lee [53]	n/a	2	ns	yes	exclusive	no	concentrated	no	truck
Chen and Song [14]	*	*	ns	yes	exclusive	no	concentrated	no	truck
Yu and Egbelu [27] Vahdani and Zandieh [54] Arabani et al. [55]	n/a	2	automated	yes	exclusive	no	concentrated	no	allowed
Boysen et al. [56]	n/a	2	*	yes	exclusive	no	concentrated	no	allowed
Forouharfard and Zandieh [57]	n/a	2	ns	yes	exclusive	no	concentrated	no	allowed
Arabani et al. [58]	n/a	2	automated	yes	exclusive	no	concentrated	outbound	allowed
Vahdani et al. [59] Soltani and Sadjadi [60]	n/a	2	automated	no	exclusive	yes	concentrated	no	allowed
Larbi et al. [61]	n/a	2	ns	yes	exclusive	yes	scattered	no	destination
Alpan et al. [62]	*	*	ns	yes	exclusive	yes	scattered	no	destination
Boysen and Fliedner [13]	*	*	ns	yes	exclusive	no	concentrated	outbound	destination
Rosales et al. [63]	*	*	manually	yes	exclusive	no	concentrated	no	truck
Wang and Regan [52]	*	*	manually	no	exclusive	no	scattered	no	destination
Acar [64]	*	*	manually	no	exclusive	no	scattered	no	destination
McWilliams et al. [65, 66] McWilliams [67, 68, 69]	*	*	automated	no	exclusive	no	concentrated	no	destination
Chmielewski et al. [70]	*	*	manually	yes	exclusive	no	concentrated	both	destination
Lim et al. [71, 72] Miao et al. [73]	*	*	manually	yes	mixed	no	scattered	both	truck
Boysen [74]	*	*	manually	no	exclusive	no	concentrated	no	truck
Shakeri et al. [75] Li et al. [76]	*	*	manually	yes	mixed	no	concentrated	no	truck

Table 2: Characteristics of the articles discussed in Section 4.4. A '*' indicates that not a single value of the characteristic is valid, but that all values can be used, 'n/a' or 'ns' indicate that a characteristic is not applicable or not specified.

Article	Notation
Chen and Lee [53]	$[E2\|t_j = 0\|C_{max}]$
Chen and Song [14]	$[E\|t_j = 0\|C_{max}]$
Yu and Egbelu [27], Vahdani and Zandieh [54], Arabani et al. [55]	$[E2\|\text{change}\|C_{max}]$
Boysen et al. [56]	$[E2\|p_j = p, \text{change}\|C_{max}]$
Forouharfard and Zandieh [57]	$[E2\|\text{change}\|\sum S_p]$
Arabani et al. [58]	$[E2\|\text{change}\|*]$
Vahdani et al. [59], Soltani and Sadjadi [60]	$[E2\|\text{pmtn,no-wait,change}\|C_{max}]$
Larbi et al. [61]	$[E2\|\text{pmtn}\|*]$
Alpan et al. [62]	$[E\|\text{pmtn}\|*]$
Boysen and Fliedner [13]	$[E\|t_{io}, \text{fix}\|\sum w_s U_s]$
Rosales et al. [63]	$[E\|t_{io}\|*]$
Acar [64]	$[E\|r_j, \text{no-wait}\|*]$
McWilliams et al. [65]	$[E\|p_j = p, \text{no-wait}, t_{io}\|C_{max}]$
McWilliams et al. [66]	$[E\|\text{no-wait}, t_{io}\|C_{max}]$
McWilliams [67]	$[E\|p_j = p, \text{no-wait}\|*]$
McWilliams [68]	$[E\|\text{no-wait}\|*]$
Chmielewski et al. [70]	$[E\|r_j, \tilde{d}_j, \text{limit}, t_{io}\|*]$
Lim et al. [71]	$[M\|\text{limit}, t_j = 0\|*]$
Lim et al. [72], Miao et al. [73]	$[M\|\text{limit}, t_{io}\|*]$
Boysen [74]	$[E\|p_j = p, \text{no-wait}, t_j = 0\|\sum C_o]$ $[E\|p_j = p, \text{no-wait}, t_j = 0\|*]$ $[E\|p_j = p, \text{no-wait}, t_j = 0\|\sum T_o]$
Shakeri et al. [75], Li et al. [76]	$[M\|t_{io}\|C_{max}]$

Table 3: Classification of the articles discussed in Section 4.4 according to the classification scheme proposed by Boysen and Fliedner [13]. When a certain attribute is not applicable, the default value is assumed to be valid.

the makespan, but now products are assumed to be interchangeable. So, the product assignments from the inbound trucks to the outbound trucks have to be determined additionally. Also different is that a truck changeover time is considered and the travel time between the strip and stack doors has been fixed. It is assumed that the inbound trucks can be unloaded in any sequence. The problem is formulated as a mixed integer programming model. To solve large problem instances, a heuristic algorithm is proposed. The heuristic method is tested on several small problem instances and the results indicate that the solutions are close to the optimal solutions obtained by complete enumeration (percentage deviation between 0 and 11.13 %).

Vahdani and Zandieh [54] elaborate further on this problem and apply five metaheuristic algorithms to solve it: a genetic algorithm, tabu search, simulated annealing, an electromagnetism-like algorithm and variable neighborhood search. For these five metaheuristics, the solution obtained with the heuristic developed by Yu and Egbelu is used as an initial solution or as a member of the initial population. The computational experiments show that these metaheuristics can improve the solutions obtained by the heuristic of Yu and Egbelu at the expense of a slightly higher computation time. Arabani et al. [55] also present five metaheuristics to tackle this problem: a genetic algorithm, tabu search, particle swarm optimization, ant colony optimization and differential evolution.

In [56], Boysen et al. deal with a very similar problem, but on a more aggregate level. The time horizon is divided into discrete time slots and it is assumed that the trucks can be completely loaded or unloaded within such a time slot. The authors formulate the problem as an integer programming model and show that this problem is NP-hard in the strong sense. To solve it, a decomposition approach is proposed in which two subproblems are considered: given a fixed inbound sequence, determine the optimal outbound sequence and vice versa. By solving these two subproblems iteratively until a stopping criteria is met, a global solution is found. These subproblems can be solved suboptimally with a heuristic approach or exactly by a (bounded) dynamic programming approach.

Some other articles deal with very similar problems. In [57], Forouharfard and Zandieh try to sequence the inbound and outbound trucks in order to minimize the number of products that pass through temporary storage. The authors propose an imperialist competitive algorithm (ICA) to solve the problem. Arabani et al. [58] consider still another objective function. It is assumed that the outbound trucks have a due date and the objective is to minimize the total (weighted) earliness and tardiness of these trucks. Three metaheuristics are proposed to solve this problem: a genetic algorithm, particle swarm optimization and differential evolution. Vahdani et al. [59] consider also a similar problem, but now temporary storage is not allowed. To make it possible that the freight is directly shipped from inbound to outbound truck, the loading and unloading of the trucks can be halted and continued at a later point in time (pre-emption). The authors formulate the problem as an integer programming model and propose two metaheuristics to solve it: a genetic algorithm and an electromagnetism-like algorithm. In [60], Soltani and Sadjadi present two metaheuristics (hybrid simulated annealing and hybrid variable neighborhood search) to tackle the same problem.

Larbi et al. [61] consider only the scheduling of the outbound trucks in a cross-dock with a single strip and a single stack door. An arriving inbound truck is unloaded and the products with the destination of the current outbound truck are directly loaded. The other goods can be temporarily put in storage (with infinite capacity), or the outbound truck can be moved to a parking zone, liberating the stack door for another truck (pre-emption of the loading operation). It is assumed that the outbound trucks are available at any time and that the unloading can be done in any order. The loading, unloading and travel times are not considered. The objective is to find the best schedule of outbound trucks that minimizes the total cost (storage and pre-emption costs). The authors distinguish between three cases with different levels of information about the inbound trucks. In the first case, full information is assumed, i.e. the sequence of the inbound trucks and the content of all trucks is known. A graph based algorithm is proposed that can solve this case in polynomial time. In the second case, it is assumed that no information about the inbound trucks is available. Only the daily quantities to ship to each destination are known in advance. The content of an inbound truck and its arrival time is only known upon arrival. For this case, the authors propose a heuristic based on a probabilistic decision rule to determine which outbound truck should be loaded next. In the third case, partial information is available. When an inbound truck arrives, the content and the sequence of a certain number (Z) of inbound trucks that will arrive next is also revealed. Two heuristic methods are presented. For the first heuristic, the approach proposed for the full information case is adapted for a rolling horizon. The second heuristic combines the algorithms for the full information and the no information case. The first heuristic is recalculated every time a new truck arrives (so every piece of new information is taken into account), while the second heuristic only has to be recalculated when Z trucks have arrived. The performed numerical experiments indicate that the total cost increases significantly if no information is available. When only partial information is available, there is also an extra cost, but this extra cost quickly decreases as Z increases. The numerical results also suggest that in this case the second heuristic gives better results.

In [62], Alpan et al. extend this problem to a cross-dock with multiple strip and stack doors (for the case with full information). To solve the problem optimally, the authors propose a graph based dynamic programming approach. Because the number of nodes increases exponentially with the problem size, two strategies are examined to limit the number of nodes generated at each stage of the dynamic programming model.

4.4.2. Scheduling of inbound trucks

Other articles consider a more realistic cross-dock with multiple strip and stack doors, but deal only with the scheduling of the inbound trucks. It is assumed that the outbound trucks are already scheduled or are assigned on a mid-term horizon (i.e. the destinations are assigned to stack doors).

In addition to a classification scheme, Boysen and Fliedner present in [13] an optimization model for the case in which a fixed outbound schedule is used. The outbound trucks depart at predefined points in time, regardless of the loaded freight. For instance postal services usually apply fixed schedules. All shipments that arrive before the departure of the truck are loaded, the other shipments are postponed until the next departure to the same destination. The objective is then to schedule the inbound trucks in order to minimize the (weighted) number of delayed shipments. The model takes the travel time between the assigned inbound and outbound doors into account. The authors prove that this model is NP-hard in the strong sense.

Rosales et al. [63] study the scheduling of inbound trucks at a large cross-dock facility in Georgetown. The scheduling is performed for the period of one shift. The objective is to minimize the operational cost and to provide a balanced workload to all workers. The operational cost consists of two parts: the travel cost (proportional to the travel distance) and the labor cost. Because one worker is assigned to work at each dock, minimizing labor cost amounts to minimizing the number of docks required to handle the freight. Overtime is allowed, but it comes at an extra cost. The travel times are dependent on the door assignment of the inbound trucks, while the unload times are estimated based on the composition and volume of the freight. It is assumed that all trucks are available at the beginning of the shift and that pre-emption is not allowed. Goods can be temporarily stored near to the (scheduled) stack doors. The authors formulate the problem as a mixed integer programming model that includes constraints to enforce workload balancing. Computational experiments show that CPLEX is able to solve realistically sized problems in a reasonable time and outperforms the current (manual) approach. By explicitly including workload-balancing constraints, the number of used docks can be reduced with only little impact on the travel distance. The proposed model is also implemented at the Georgetown cross-dock and leads to a cost reduction and a better balanced workload.

Wang and Regan [52] also consider the scheduling of inbound trucks and propose some (dispatching) rules that are applicable in a dynamic environment. When a strip door becomes available, and multiple inbound trucks are waiting to be unloaded, one of these trucks has to be handled first. Usually, the next truck is chosen based on the FCFS policy. This is a fair rule with respect to the waiting time of the inbound trucks, but it may not lead to the most optimal result for the cross-dock as a whole. In a cross-dock, the travel time between the docks is usually small compared to the time the products have to wait inside the trucks or at the docks. So, the authors propose two time-based algorithms that are concerned with the impact of a new inbound truck on the total processing or total transfer time. The processing time of a product consists of the waiting time at the strip door (inside the truck), the travel time between strip and stack door and the waiting time inside the outbound truck. The transfer time considers also the waiting time before the inbound truck is docked. It is assumed that there is always an outbound truck available for each destination, so there is no temporary storage space needed. The unloading of the trucks cannot be interrupted (no pre-emption) and the arrival times are scattered throughout the day. The authors performed a simulation study to compare both algorithms with the FCFS rule and the look-ahead policy proposed by Gue [33]. They conclude that significant time savings can be obtained by using the proposed time-based rules, at least when the average number of waiting trucks is higher than 0.65.

Another approach to schedule only the incoming trucks is taken by Acar [64]. In his master thesis, the objective is not to minimize the travel distance inside the cross-dock, but to have an assignment that is robust against the variability in system parameters such as truck arrival times, service times (for loading, unloading and transferring freight) and the truck loads. The author formulates the problem as a mixed integer quadratic programming (MIQP) problem to minimize the variance associated with the distribution of the idle times of the docks. Indeed, an assignment with even distribution of idle times at the strip docks will tend to absorb the stochastic variability in the arrival and service times. It is assumed that there is always an outbound truck docked at each stack door, so there is no temporary storage space needed. The truck arrival times are taken into account and pre-emption is not allowed. For each inbound truck, there can be a different service time (to unload and move its content). Because of the computational complexity, a simple heuristic algorithm is proposed. Some experimental tests on small problem instances indicate that this heuristic gives results on average within 4.41 % of the optimal solution (but the maximum deviation is about 16 %). The author also proposes a dynamic heuristic to assign the trucks to docks at real time.

McWilliams et al. [65, 66] consider the truck scheduling of inbound trucks at a cross-dock used in the parcel delivery industry. In such a cross-dock, unloaded parcels are transported to outbound trucks by means of a fixed network of conveyors. Because of this stationary network, the designation of doors as either strip or stack doors is fixed and the route of a parcel is defined by its assigned strip and stack door. The travel time of a parcel is dependent on its route, but also on congestion of the conveyor network. The objective of this *parcel hub scheduling problem* (PHSP) is then to minimize the time interval from the unloading of the first parcel until the loading of the last parcel (makespan). It is assumed that all trucks are available at the beginning of the time horizon and that pre-emption is not allowed. As full outbound trucks are immediately replaced, goods do not have to be intermediately stored. In [65], it is assumed that the batch sizes (and the unload times) of the inbound trucks are equal, while this assumption is relaxed in [66]. Because a conveyor network is a queueing network, it is difficult to develop an analytical model of its behavior. So, the authors propose a simulation-based scheduling algorithm (SBSA) to solve the PHSP. This algorithm is a genetic algorithm that makes use of a detailed deterministic simulation model to evaluate the makespan for each candidate solution. Computational results show a significant reduction in the makespan (between 4.2 and 35.8 %) compared to arbitrary scheduling (as a representation of current practice).

Simulation optimization is however computationally expensive and requires excessive computing time to obtain solutions for large-scale problems. So, in [67, 68], McWilliams proposes a decomposition approach to tackle the PHSP. A combination of time-based and resource-based decomposition is applied. The time horizon is divided into several smaller sub-periods (time buckets) and the focus is on the bottleneck resources (the final sorters of the conveyor network). The objective is to minimize the maximum workload at the final sorters over all time buckets. This workload balancing problem is formulated as a (NP-hard) minimax programming model. The time between unloading and arriving at the bottleneck is assumed to be independent of the used strip door. In [67], it is assumed that the batch sizes (and the unload times) of the inbound trucks are equal and a genetic algorithm is used to solve the problem. In [68], this assumption is relaxed and the problem is solved by applying a local search algorithm and simulated annealing. Computational results indicate that the genetic algorithm finds solutions with a significant lower makespan than the SBSA while the computation time is more than a factor 10 lower. The local search and simulated annealing on its turn seem to improve the results of the genetic algorithm in a similar computation time.

In the previous approaches, the workload at the final sorters is balanced in a static way. In [69], McWilliams presents a dynamic load balancing algorithm (DLBA). Whenever an unload dock becomes idle, one of the waiting inbound trucks has to be assigned to the idle dock. The objective is to balance the flow of parcels through the conveyor network and to avoid flow congestion. The algorithm makes use of updated information on the availability of inbound trucks and the state of the cross-dock. Computational results show that the proposed algorithm (applied in a static context) gives, for large problem instances, significant better results than the static approach in [68], and this in a much shorter computation time.

Chmielewski et al. [70] also study the scheduling of inbound trucks, but the authors consider at the same time the assignment of the outbound trucks on a mid-term horizon (i.e. the destinations are assigned to stack doors). Unloaded goods are placed in a buffer area, from where they are transported to a buffer area for loading (at each stack door). The workers and resources needed to perform this transportation are limited, and also the size of the buffer areas is limited. Pre-emption is not allowed and an earliest arrival and latest departure time are defined for each truck. One objective is to find an optimal schedule that leads to minimal total distances and a minimal number of required resources. A second objective is the minimization of waiting times. Trucks should be allocated to a door as soon as possible after their arrival. The authors propose two solution approaches. In a first approach, the problem is modeled as a time-discrete, multicommodity flow problem with side constraints. The objective is to minimize the total cost (based on the travel distance). The costs increase slightly with time in order to take also the second objective (to minimize the waiting time) into account. To solve this mixed integer problem, the authors propose a decomposition-and-column-generation approach. The second approach makes use of a multi-objective evolutionary algorithm (EA) that results in a set of Pareto-optimal solutions. This is a real multi-criteria approach that tries to minimize the total travel distance and the total waiting time. Two variants are considered: (1+1)-EA with one offspring each iteration and ($\mu+\lambda$)-EA with multiple offspring each iteration. Computational results show that the decomposition-and-column-generation approach outperforms the standard algorithm for MIP (branch-and-bound with CPLEX) in terms of lower objective function values and better feasible solutions. However, this approach can only be used for a limited number of discrete time periods because otherwise the flow network becomes much too large. The computation times of the EA algorithms are much lower, but at the expense of solution quality; the total distance of the solutions is much higher. However, the waiting times are much better, due to the the multi-objective approach.

4.4.3. Scheduling of inbound and outbound trucks

The following articles deal with the scheduling of both inbound and outbound trucks.

In [71], Lim et al. consider a truck scheduling problem in which it is assumed that the trucks are loaded or unloaded during a fixed time window. This means that the scheduling problem is reduced to determining at which dock door the trucks have to be processed. The length of these time windows can be interpreted as the time needed to load or unload a truck. The objective of this so-called *truck dock assignment problem* is to minimize the total travel distance. The trucks can be assigned to any door (mixed service mode) and the capacity of the cross-dock is limited. Pre-emption is not allowed and trucks that cannot be served are penalized by adding an extra distance. A shortcoming of this approach is that the time to transport freight between the dock doors is not taken into account. The authors formulate the problem as an integer programming model and because the problem is NP-hard, they propose a tabu search and a genetic algorithm approach to solve it.

The same authors extend this approach by taking the travel time between the docks into account [72, 73]. The objective is now to minimize the operational cost (based on travel time) and the cost of unfulfilled shipments. A similar tabu search heuristic [73] and an adapted genetic algorithm [72, 73] to solve this truck scheduling problem are discussed. The experimental test results indicate that the genetic algorithm outperforms CPLEX in terms of solution quality and computing time. The tabu search approach on its turn seems to dominate the genetic algorithm.

In [74], Boysen deals with truck scheduling for a cross-dock in which products are not allowed to be intermediately stored. Such a zero-inventory policy is for instance used when frozen goods are transported and the cross-docking terminal is not cooled. To make sure that the cooling chain is not broken, goods are not allowed to be intermediately stored. This policy can be applied in several industrial sectors, but the article focuses on the food industry. As a result, products are dedicated to a specific outbound truck and are not interchangeable. In the food industry, standardized cargo carriers and trailers are used, so it is assumed that docking, unloading and undocking of trucks take a very similar amount of time. The author also assumes that the travel times of goods inside the cross-dock are negligible because of the small size of cross-dock terminals in the food industry. Each dock door is exclusively dedicated to inbound or outbound operations (exclusive service mode). The author presents a formalization of the truck scheduling problem that can take into account different operational objectives (minimization of flow time, processing time and tardiness of outbound trucks). To solve this problem optimally, a dynamic programming approach is proposed in which an acyclic directed graph is constructed. The shortest path in this graph then corresponds to the optimal solution. This approach can be extended by applying lower and upper bounds (bounded dynamic programming). The author also presents a simulated annealing procedure. A computational study shows that the (bounded) dynamic programming approach can be used to solve smaller problem instances (up to 25 inbound trucks) optimally within a few minutes. For realistic (larger) problem sizes, the simulated annealing approach is able to find near-optimal results in less than 1 second. The author also indicates how this method can be used as part of a rolling horizon approach.

Shakeri et al. [75] study the truck scheduling problem in a cross-dock where goods are exchanged between the trucks, i.e. each truck serves both as inbound and as outbound truck. The problem is modeled as a two-stage parallel-machine scheduling problem and the objective is to minimize the makespan. In the unloading stage, goods are unloaded and moved to the temporary storage (with infinite capacity) at the correct dock door. It is assumed that the different goods of a truck can be unloaded in parallel. The moving can only start after unloading and when the destination truck is docked. The travel time is based on the distance between the dock doors. In the loading stage, the goods are (sequentially) loaded into the trucks. The loading of a truck can only start if its own goods are unloaded and all products that have to be loaded are available in the storage area. It is also assumed that all trucks are available at the beginning of the planning horizon and that pre-emption is not allowed. Between two consecutive trucks, a setup time is taken into account. The authors provide a (non-linear) mixed integer programming formulation of the problem that can be used for small problem instances.

In [76], Li et al. present a heuristic method in order to solve larger problem instances. This dependency ranking search (DRS) heuristic consists of two parts. The first part builds a feasible sequence of jobs with respect to the number of dock doors. In the second part, these jobs are assigned to doors based on the distance between the doors. Some computational experiments were performed and the results show that, for the small instances, the CPLEX solver performs slightly better than the DRS heuristic. However, CPLEX is much slower. For medium and large problem instances, CPLEX is not able to find solutions (in a time limit of 2 hours) for most cases, while the heuristic finds a solution in more than 8 of the 10 instances (in a few minutes).

4.5. Temporary storage

Although the idea of cross-docking is to unload products from trucks and directly load the products into departing trucks, temporary storage is usually inevitable. Freight has to be staged because of the imperfect synchronization of inbound and outbound trucks and because the goods do not arrive in the sequence in which they must be loaded. The loading sequence is for instance determined by the need to build tightly packed loads or to place fragile products on top, or by the order in which the goods have to be delivered if there are multiple stops [3]. Usually, a dispatching rule is used to determine where the freight has to be staged, for instance in front of the dock door where the outbound truck is or will be docked. There are however articles that deal with the operational decision where to store incoming freight. The characteristics of the considered cross-docks are summarized in Table 4.

A first study is performed by [15]. In this article, the aim is to determine temporary storage locations for incoming freight such that the total travel distance of the goods is minimized. It is assumed that the dock door assignment and the travel distances are known. The authors show that this problem can be modeled as a minimum cost flow problem, for which several polynomial time algorithms exist. A storage location can however be used only once in this approach. Therefore, the authors propose to solve the problem multiple times, each time taking the freight for the corresponding period into account. As a result, storage locations can be used multiple times. Numerical experiments are performed to compare the proposed method with a situation in which the workers choose the storage locations and which usually results in loads stored at available locations nearest to the origins of the loads. The results show that the proposed algorithm can reduce the total travel distance up to about 40 %.

In his master thesis, Sandal [77] uses simulation to compare several staging strategies in order to support the optimal loading of the outbound trucks. The author distinguishes three cases that determine which freight is staged: no freight is staged (pure cross-docking), all freight is staged and the loading only starts when all goods are stored, or the goods that will (seriously) violate the scheduled loading sequence are staged while the other freight is loaded directly. When the freight is staged, two strategies can be distinguished. In the first strategy, the storage area before each stack door is treated as a single FCFS queue. In the second strategy, these storage areas are divided in three equal zones and freight is placed in one of these zones based on its ranking in the scheduled loading sequence.

4.6. Vehicle routing

Freight destined for a cross-dock needs in many cases to be picked up at various locations, and has to be delivered to multiple locations after consolidation at the cross-dock. Both the pickup and the delivery process can be seen as a vehicle routing problem and some studies consider cross-docking and vehicle routing simultaneously.

A first approach is taken by Lee et al. [78]. The aim is to find an optimal routing schedule for pickup and delivery (within the planning horizon) that minimizes the sum of transportation cost and fixed costs of the vehicles. It is assumed that split deliveries are not allowed and all pickup vehicles should arrive at the cross-dock simultaneously to prevent waiting times for the outbound trucks. While this can be a valid constraint for some cases (see Section 3.3), this is not generally true. The authors present an integer programming model of the problem, which however seems unsatisfactory to solve the described problem. A tabu search algorithm is proposed to find solutions. This approach corresponds to the solving of two vehicle routing problems (one for pickup and one for delivery). The second routing problem can only start when the first one is finished and the complete process has to be finished within a certain planning horizon. In [79], Liao et al. propose another tabu search algorithm to solve the same problem.

Wen et al. [80] study the so-called vehicle routing problem with cross-docking (VRPCD). In this problem, orders from suppliers have to be picked up by a homogeneous fleet of vehicles. These orders are then consolidated at a cross-dock and immediately delivered to customers by the same set of vehicles, without intermediate storage at the cross-dock. During the consolidation, goods are unloaded from the inbound vehicles and reloaded on outbound vehicles. The unloading must be completed before reloading starts. The authors assume that the duration of the unloading consists of a fixed time for preparation and a duration proportional to the load size. It is also assumed that if the delivery will be executed by the same vehicle as used for pickup, the unloading is not necessary (independent of the sequence in which the vehicle is loaded during the pickup tour). A time window is defined for all suppliers and customers and orders are not splittable. In the case without consolidation, the solution of this problem can be found by solving two vehicle routing problems (one for pickup and one for delivery). Because of the consolidation however, the pickup and delivery routes are not independent. Only trying to minimize the distance of the pickup and delivery routes is not sufficient, the exchanges of orders at the cross-dock also have to be taken into account. These two aspects usually conflict with each other. The authors present a mixed integer programming formulation of the problem in which the objective is to minimize the total travel time of all vehicles. This formulation contains many variables and constraints, so the authors propose to use tabu search embedded within an adaptive memory procedure. This method is tested on realistic data involving up to 200 supplier-customer pairs. Experimental results show that the algorithm can produce solutions less than 1 % away from the optimum within short computing times (less than 5 seconds) for small problem instances. For larger instances, the gap with a lower bound is less than 5 % while the computation time stays below 5 minutes.

4.7. Cross-docking networks

Some authors do not study problems concerning a single cross-dock, but consider a network that contains one or more cross-docks. The aim is to determine the flow of goods through such a network in order to reduce costs, while making supply meet demand.

The research of Lim et al. [84] extends the traditional transshipment problem. The transshipment problem consists of a number of supply, transshipment and demand nodes. The arcs between these nodes have different capacity limits and costs. The objective is to find a minimum cost flow that meets all demands

	Shape	No. of doors	Internal transport	Temporary storage	Service mode	Pre-emption	Arrival pattern	Departure time	Interchange-ability
Vis and Roodbergen [15]	I	*	manually	yes	exclusive	ns	ns	ns	truck
	*	*	manually	yes	exclusive	ns	ns	ns	truck
Sandal [77]	I	6	manually	yes	exclusive	no	concentrated	no	destination

Table 4: Characteristics of the articles discussed in Section 4.5. A '*' indicates that not a single value of the characteristic is valid, but that all values can be used, 'ns' indicates that a characteristic is not specified.

	Shape	No. of doors	Internal transport	Temporary storage	Service mode	Pre-emption	Arrival pattern	Departure time	Interchange-ability
Li et al. [5] Álvarez-Pérez et al. [81]	*	*	ns	yes	ns	no	scattered	both	truck
Stickel [82]	*	*	manually	yes	exclusive	no	scattered	outbound	destination
Magableh et al. [83]	*	*	manually	yes	exclusive	no	scattered	outbound	destination

Table 5: Characteristics of the articles discussed in Section 4.8. A '*' indicates that not a single value of the characteristic is valid, but that all values can be used, 'ns' indicates that a characteristic is not specified.

and the capacity constraints. In the extended transshipment problem, storage is allowed at the transshipment centers. These centers can be considered as cross-docks because the aim of the model is to minimize or eliminate holdover inventory. Moreover, this problem takes supplier and customer time windows into account and considers the capacity and holding costs of the cross-docks. All shipments have to pass via a cross-dock, so no direct shipments are considered. Similar to the original problem, the objective is to minimize the total cost (transportation costs and holding costs) while meeting demand and respecting the time windows and capacity constraints. If multiple departures and deliveries within a time window are allowed (multiple shipping–multiple delivery), the authors show that a time-expanded network can be used to formulate the problem as a minimum cost flow problem (MCFP) which can be solved in polynomial time. For other cases, the authors prove that the problem is NP-hard.

For the special case when only one delivery or departure is allowed within a time window and the departure and arrival times are fixed (single shipping–single delivery with fixed schedules), a genetic algorithm is developed by Miao et al. [85]. This heuristic gives better results (in terms of solution quality and computation time) than solving the integer programming formulation of the problem with CPLEX (with a time limit).

Chen et al. [86] study a similar problem which they call the multiple cross-dock problem. The major differences are that supplies and demands are not-splittable and that different products can be considered (multicommodity flow problem). Also, transportation time is in this approach not taken into account. An integer programming formulation of the problem is provided, together with a proof of its NP-completeness. The authors propose three heuristics (simulated annealing, tabu search and a combination of both) to solve the problem. These heuristics provide better solutions than those obtained by solving the integer programming formulation with CPLEX, within only less than 10 % the time used by CPLEX. Among the three heuristics, tabu search seems to give the best results.

The previous studies represent the shipment of goods as flows. Individual transportation units are not considered and the transportation cost is proportional to the quantity to ship. However, to take advantage of consolidation, the vehicle transportation cost should be taken into account. A first approach that does consider the transportation vehicles explicitly (and this is why the authors regard it as cross-docking) is taken by Donaldson et al. [35]. In the considered problem, the goal is to determine whether to route freight directly from suppliers to customers or via a cross-dock and how many vehicles should be scheduled on each transportation link in order to minimize the transportation costs. Compared to the previous approaches however, this problem is more simplified, e.g. storage at the cross-docks is not considered and the synchronization of inbound and outbound trucks is left out of the problem. The authors eliminate links with a large transportation time in an attempt to consider time windows. However, when the due dates at the destination nodes can vary for the different goods, it is possible that the vehicle allocation of an obtained solution violates the due dates in practice. The authors present an integer programming model of the problem. Because the problem is difficult to solve with branch-and-bound algorithms, an alternative approach is proposed. In this approach, an iterative procedure is used in which either the integrality restrictions on the links from origin nodes to the cross-docks or on the links from the cross-docks to the destination nodes are relaxed. This relaxation heuristic provides near optimal solutions in an acceptable time. The authors used this approach to compare several scenarios (with a different number of cross-docks at different places) for the network design of a postal service company.

The same problem is also studied by Musa et al. [36]. They propose an ant colony optimization (ACO) heuristic to solve the problem and show that this heuristic gives in a short time slightly better results than a branch-and-bound approach (with the optimization software package LINDO) that requires a much longer time.

The approach of Ma et al. [87] takes most of the above-mentioned concerns into account. The so-called *shipment consolidation problem* (SCP) considers supplier and customer time windows and also the transportation times between the network nodes. Moreover, storage at the transshipment centers (cross-docks) is taken into account, shipments can be transported directly to their destination or via a cross-dock and the transportation cost accounts for the number of trucks. However, only one type of products is considered (single commodity). Again, the objective is to minimize the total cost (transportation and inventory cost) while satisfying the constraints imposed by the time windows. The authors present an integer programming model of the problem and show that it is NP-complete in the strong sense. Therefore, the authors propose a (two-stage) heuristic algorithm to solve the problem. The basic idea of the algorithm is to consider first trucks that can be fully loaded and then to find solutions that combine several smaller loads that are not considered yet. In the first stage, a full truckload plan (TL plan) and an initial less-than-truckload plan (LTL plan) are constructed. In the second stage, this initial LTL plan is improved iteratively by using a metaheuristic (squeaky wheel optimization or genetic algorithm). The computational experiments indicate that the proposed heuristic gives competitive results compared to CPLEX (with a time limit) within a much shorter time.

4.8. Other issues

The following articles deal with still other cross-docking issues. The characteristics of the considered cross-docks (for the articles in which the internal details are described) are summarized in Table 5.

In [5], Li et al. consider the scheduling of internal resources for the loading and unloading of freight. The loading and unloading process is usually accomplished by teams of workers and equipment. Since the number of available teams is limited, these teams have to be scheduled efficiently. The objective is to complete the processing of each truck as close as possible to its due date (just-in-time). The authors model this problem as a two-phase parallel machine scheduling problem with earliness and tardiness penalties and formulate it as an integer programming model. This scheduling problem is NP-hard, so two heuristic approaches are proposed. Both approaches use a genetic algorithm and try to improve the best solution of each generation. The first approach applies a local search heuristic (squeaky wheel optimization), while the second approach solves the integer programming subproblem that results when the assignment of teams to trucks is fixed and only the start and end time of the loading and unloading processes can change. Experimental results indicate that both approaches find near-optimal solutions in a much shorter time than CPLEX. The second approach gives the best result, but at the expense of a longer computation time.

Álvarez-Pérez et al. [81] propose a different method to solve the same problem. This method is a combination of Reactive GRASP (greedy randomized adaptive search procedure) and tabu search. The Reactive GRASP procedure is used to construct initial solutions which are improved by the tabu search algorithm. The numerical experiments suggest that this method performs similar or slightly better compared with the heuristics of Li et al., but is in turn more time-consuming for the larger problem instances.

Stickel [82] deals with the problem in which not only the scheduling of the internal resources is considered, but also the vehicle routing problem that appears for the pick-up and delivery of goods and the scheduling of the inbound and outbound trucks. Compared to previous approaches, these three types of problems are integrated and solved simultaneously. The author proposes two solution approaches: a centralized-hierarchical and decentralized-heterarchical approach. The centralized-hierarchical approach corresponds to the situation in which a central instance (like a third-party logistics provider) has all relevant information and can take the necessary decisions. The problem is formulated as a mixed integer problem. However, because of its complexity, only small problem instances can be solved by applying branch-and-bound (with CPLEX). In the second approach, it is assumed that there is not a single entity that has all decision power, but several entities have to cooperate. The truck scheduling is interpreted as an interface between the vehicle routing and the scheduling of the internal resources and time slots at the dock doors are allocated among the cooperating entities by means of a combinatorial auction.

Yan and Tang [30] compare the costs of a traditional distribution center with the cost of pre-distribution and post-distribution cross-docking. In pre-distribution cross-docking, it is assumed that the goods are directly loaded into outbound trucks. The suppliers are responsible for the necessary preparation and sorting to facilitate immediate loading at the cross-dock. This requires that the suppliers know the order quantities for each destination. In post-distribution cross-docking, the preparation and sorting happens at the cross-dock itself. This incurs higher costs at the cross-dock, but allows to assign the goods to destinations upon arrival at the cross-dock. In this way, the influence of the fluctuating demand can be reduced by pooling the risk during the transportation period from the supplier to the cross-dock. The authors construct analytical models to perform a pair-wise comparison of the cost (including inventory, back order and operational costs) of the three systems. It is assumed that the demand is correlated between two adjacent periods, but independent between different destinations. The cost of the different systems depends on several parameters (e.g. delivery lead time, unit holding cost and variation in demand). Numerical experiments are performed to study the influence of these parameters on the preference of cross-docking. The results indicate that pre-distribution cross-docking is preferred when the demand is stable and the leadtime between supplier and cross-dock is short. However, when the demand is uncertain and the leadtime is long, the benefits of reallocating goods among stores outweights the higher operational costs and so post-distribution cross-docking is preferred. Post-distribution also seems to be preferable if the number of destinations or the unit holding cost increases.

In [88], the same authors compare in a similar way pre-distribution and post-distribution cross-docking when transshipments are allowed; goods can be shipped from an overstocked destination to a nearby understocked destination in order to avoid back orders. Post-distribution cross-docking has higher operational costs than pre-distribution cross-docking, but will need less transshipments due to the pooled demand. It is assumed that the demand is independent in time and between different destinations. An analytical formulation of the costs (including transshipment costs) for both systems is provided and the cost sensitive factors are analyzed. The results of numerical experiments suggest that a higher uncertainty of demand, a higher unit transshipment costs and a longer lead time from the supplier to the cross-dock make post-distribution cross-docking more preferred. Pre-distribution cross-docking is more preferable when the unit holding cost or unit back order cost is very high or very low.

Simulation is a general technique that also can be used to deal with several aspects of cross-docking. For instance, simulation allows to compare alternative cross-dock layouts or can be used to test various dock door assignment strategies, and this for one or more selected performance metrics. Some of the articles discussed above make use of simulation, e.g. [28, 42, 49, 52, 65, 77].

In [89], Rohrer explains that simulation on the one hand is useful to determine whether all the equipment will function together properly and to test different design alternatives. On the other hand, simulation can also be used to test alternative control algorithms before the actual implementation. The author lists also some issues that have to be taken into account while modeling a cross-dock and provides some useful performance metrics.

Magableh et al. [83] present a simulation model that represents the operations within a cross-dock, specifically the processing of inbound and outbound shipments. The authors tried to make the model generic so that it can easily be expanded to model other cross-docking facilities. The presented model can for instance be used to analyze the effect of an increased demand or to compare the performance of different dispatching rules.

5. Conclusion

As can be noted, a considerable amount of articles about cross-docking has been published, certainly during the recent years. Several articles deal with cross-docking in a more general way (e.g. suitability for cross-docking and the implementation of cross-docking), while other articles are concerned with a specific type of problem (on a strategic, tactical or operational level). Especially the problems of dock door assignment (Section 4.3) and truck scheduling (Section 4.4) have attracted the attention of many researchers. Despite this attention, we believe that there are still many opportunities to improve and extend the current research.

First of all, not all problems with which cross-docking practitioners are confronted are extensively discussed. For instance, only a few articles about cross-dock layout design (Section 4.2) are published. These articles deal with the shape of the cross-dock and the design of the storage area. Other aspects, like the dimension of the cross-dock and the dimension, shape and arrangement of the internal cross-dock areas, are however not considered. There are also not many articles that deal with temporary storage (Section 4.5), while a good strategy can improve the cross-docking operations, for instance by avoiding excessive travel distances and congestion.

In the second place, not all types of cross-docks are considered. As can be seen in Tables 1, 2, 4 and 5, the same characteristics are appearing and some characteristics do almost not occur. For instance, only a few articles deal with a conveyor network for the internal transportation. While the use of forklifts may indeed be more common in industry, there are also many cross-docks that make use of a conveyor network (e.g. parcel carriers). As the choice for an automated system imposes some restrictions (exclusive service mode, fixed routes) and gives rise to some other issues (congestion), it would be interesting to specifically consider this type of cross-docking. Also, most articles study cross-docks with an exclusive mode of service and without pre-emption. While this can simplify the planning and execution of the daily operations, it limits the flexibility of the cross-dock. So, future research could be performed to determine how pre-emption and a mixed service mode can be correctly applied in order to improve the cross-docking operations. Moreover, not many articles take restrictions on departure times (deadlines) for the trucks into account. Also, only a few articles assume that goods are interchangeable, while this is not an exceptional situation (e.g. inbound trucks arriving at the cross-dock of a retailer containing only one product type destined for several branches). So it would be interesting if future research also considers restricted departure times and interchangeable products.

Thirdly, many of the presented articles make simplifying assumptions that limit the real-world applicability. For instance, it is usually assumed that all trucks are available at the beginning of the time horizon, that the loading and unloading of a truck can be done in any order and can start immediately after docking (the required workforce and material is always available) and that the cross-dock has an infinite capacity to temporarily store freight. Other common simplifications are that internal congestion is not taken into account (except in e.g. [48, 65, 66]), value added activities like repacking or labeling are not considered and trucks are or inbound or outbound, but not both (except in [75, 76]). Future research should address these assumptions in order to make the proposed approaches more applicable in practice.

In the fourth place, also to improve the applicability, the approaches should be more robust and dynamic. In the presented articles, it is usually assumed that all necessary information, for instance about the incoming loads (e.g. the exact content and arrival time), is fixed and known on beforehand. However, discussions with cross-docking practitioners reveal that there are (serious) deviations between the predicted and actual information. For instance, it is not unusual that the weight of an arriving load is higher than indicated on the cargo documents, which can possibly cause an overloaded outbound truck. So, robustness against these kinds of deviations is required in order to be applied in an industrial context. Only one of the presented articles proposes a robust approach; in [64], the objective is to obtain a schedule of the inbound trucks that is robust against variability in the arrival and service times. Moreover, most of the presented articles are not appropriate for a dynamic environment, as the considered (operational) problems are assumed to be static. Of course, this is a simplification of reality. The control of a cross-dock is a going concern and so these problems are inherently dynamic (trucks arrive early or late, equipment fails, ...). Consequently, real time decisions are necessary. A few articles propose a (simple) dynamic approach (e.g. [51, 52, 64, 69]) or explain how the proposed static approach can be applied as part of a rolling horizon approach (e.g. [74]), but certainly more research in this direction is required.

Lastly, in practice, cross-dock practitioners have to deal with several problems together. While some of the presented articles tackle more than one problem (e.g. [49, 77, 82] and the articles discussed in Section 4.1), most articles are concerned with just one problem. Furthermore, as these problems are interdependent, improvements are expected when they can be solved together. So, future research is required that integrates several problems in one approach.

For instance, it would be interesting to combine truck scheduling with the routing of the trucks. On the one hand, the routing schedules of the inbound trucks determine the arrival times at the cross-dock, which in turn influence the scheduling of the trucks. Also, the routings of the outbound trucks influence the truck scheduling by setting deadlines for the trucks. On the other hand, if the truck schedule also determines which goods have to be combined in one truck, this influences the routing of the outbound trucks. As both problems are interdependent, it makes sense to combine them. This provides more alternatives to the decision maker, which allows better solutions but also makes the decision making problem more difficult.

The truck scheduling and the scheduling of the resources inside the cross-dock are also interdependent problems that can be combined. The scheduling of the trucks heavily influences the workload for the internal resources. For instance, the assignment of the trucks to dock doors determines the travel distance for the workers. Also, by not correctly spreading the workload (in space and time), congestion can occur inside the terminal. Conversely, the resource scheduling determines the time lag between the inbound and outbound operations and influences in this way the truck scheduling. So, solving both problems simultaneously can improve the cross-docking operations.

The scheduling of the trucks and the internal resources is also interdependent with the packing and unpacking of loads. The time at which a certain item can be unloaded is dependent on the way the inbound truck is packed, while the loading sequence of the outbound trucks determines if goods can be directly moved from inbound to outbound or have to be temporarily stored. Consequently, this influences the involved material handling and the time lag between the unloading and loading operations. There is a trade-off between the optimal packing of the trucks which involves more material handling, and less material handling but a worse packing of the trucks (and possibly more trucks are required to transport the same amount of freight). So, it could be interesting to combine the scheduling with the packing decisions. Moreover, the packing also influences the vehicle routing as it imposes restrictions on the sequence of customer visits. Other problems that are interdependent and for which benefits can arise by solving or considering them together are the scheduling of trucks and the unloading strategy for the work force (e.g. workers unload a truck completely before unloading another truck (trailer-at-a-time [49]), workers can unload a certain number of trucks together, ...), the layout design and the temporary storage strategy, and the vehicle routing and the routing of goods through a cross-docking network.

It is clear now that cross-docking poses complex and challenging problems, during the design phase as well as during operations. Cross-docks can be subject to different organizational and management approaches and different objectives can be aimed for. Particularly, cross-docks have to operate today in an uncertain and dynamic environment, among others due to a tough competition in the transport and logistic sector and ever-increasing traffic. Dealing with uncertainty is important and flexibility becomes a major topic. Unrealistic assumptions and too rigid approaches prevent an efficient cross-dock operation. As the operational control of a cross-dock is a going concern, 'one-shot optimization' is not sufficient. Because of these complicated problems, it is worthwhile to consider approaches that proved to be useful in other domains, e.g. manufacturing execution systems (MES) to control the operations in a factory in real time. More specific, the authors see opportunities to apply the concepts and principles of the holonic manufacturing paradigm [90–95] to coordinate and control cross-docks. A Holonic MES tries to improve the responsiveness, proactiveness, robustness and flexibility of a production system, properties that are also of high interest to manage cross-docking operations.

References

[1] 2008 Cross-Docking Trends Report, http://www.saddlecrk.com/whitepaper, 2008.

[2] 2011 Cross-Docking Trends Report, http://www.saddlecrk.com/whitepaper, 2011.

[3] J. J. Bartholdi III, K. R. Gue, The Best Shape for a Crossdock, Transportation Science 38 (2) (2004) 235–244.

[4] M. R. Galbreth, J. A. Hill, S. Handley, An Investigation of the Value of Cross-Docking for Supply Chain Management, Journal of Business Logistics 29 (1) (2008) 225–239.

[5] Y. Li, A. Lim, B. Rodrigues, Crossdocking—JIT Scheduling with Time Windows, The Journal of the Operational Research Society 55 (12) (2004) 1342–1351.

[6] B. Schaffer, Implementing a Successful Crossdocking Operation, IIE Solutions 29 (10) (1997) 34–36.

[7] E. Kinnear, Is there any magic in cross-docking?, Supply Chain Management: An International Journal 2 (2) (1997) 49–52.

[8] Glossary of the Material Handling Industry of America (MHIA), http://www.mhia.org/learning/glossary, consulted: June 2011., ????

[9] C. E. Witt, Crossdocking: Concepts demand choice, Material Handling Engineering 53 (7) (1998) 44–49.

[10] U. M. Apte, S. Viswanathan, Effective Cross Docking for Improving Distribution Efficiencies, International Journal of Logistics: Research and Applications 3 (3) (2000) 291–302.

[11] M. Aichlmayr, Cut The Crossdock Hype, Transportation & Distribution 43 (5) (2002) 56.

[12] R. L. Cook, B. Gibson, D. MacCurdy, A lean approach to cross docking, Supply Chain Management Review 9 (2) (2005) 54–59.

[13] N. Boysen, M. Fliedner, Cross dock scheduling: Classification, literature review and research agenda, Omega 38 (6) (2010) 413–422.

[14] F. Chen, K. Song, Minimizing makespan in two-stage hybrid cross docking scheduling problem, Computers & Operations Research 36 (6) (2009) 2066–2073.

[15] I. F. Vis, K. J. Roodbergen, Positioning of goods in a cross-docking environment, Computers & Industrial Engineering 54 (3) (2008) 677–689.

[16] G. Stalk, P. Evans, L. E. Shulman, Competing on Capabilities: The New Rules of Corporate Strategy, Harvard Business Review 70 (2) (1992) 57–69.

[17] M. Napolitano, Cross dock fuels growth at Dots, Logistics Management 50 (2) (2011) 30–34.

[18] F. Robusté, Centralized Hub-Terminal Geometric Concepts. I: Walking Distance, Journal of Transportation Engineering 117 (2) (1991) 143–158.

[19] F. Robusté, C. F. Daganzo, Centralized Hub-Terminal Geometric Concepts. II: Baggage and Extensions, Journal of Transportation Engineering 117 (2) (1991) 159–177.

[20] H. Ding, A. Lim, B. Rodrigues, Y. Zhu, New heuristics for over-constrained flight to gate assignments, Journal of the Operational Research Society 55 (7) (2004) 760–768.

[21] A. Haghani, M.-C. Chen, Optimizing gate assignments at airport terminals, Transportation Research Part A: Policy and Practice 32 (6) (1998) 437–45.

[22] D. Agustina, C. K. M. Lee, R. Piplani, A Review: Mathematical Modles for Cross Dock Planning, International Journal of Engineering Business Management 2 (2) (2010) 47–54.

[23] Z. Li, M. Y. H. Low, Y. G. Lim, B. Ma, Optimal Decision-making on Product Ranking For Crossdocking/Warehousing Operations, in: Proceedings of the 6th IEEE International Conference on Industrial Informatics (INDIN 2008), 871–876, 2008.

[24] H. L. Richardson, Cross Docking: Information flow saves space, Transportation & Distribution 40 (11) (1999) 51–54.

[25] V. B. Kreng, F.-T. Chen, The benefits of a cross-docking delivery strategy: a supply chain collaboration approach, Production Planning & Control 19 (3) (2008) 229–241.

[26] M. A. Waller, C. R. Cassady, J. Ozment, Impact of cross-docking on inventory in a decentralized retail supply chain, Transportation Research Part E: Logistics and Transportation Review 42 (5) (2006) 359–382.

[27] W. Yu, P. J. Egbelu, Scheduling of inbound and outbound trucks in cross docking systems with temporary storage, European Journal of Operational Research 184 (1) (2008) 377–396.

[28] K. R. Gue, K. Kang, Staging queues in material handling and transportation systems, in: Proceedings of the 33rd conference on Winter simulation, 1104–1108, 2001.

[29] S. Pearson Specter, How to crossdock successfully, Modern Materials Handling 59 (1) (2004) 42–46.

[30] H. Yan, S.-l. Tang, Pre-distribution and post-distribution cross-docking operations, Transportation Research Part E: Logistics and Transportation Review 45 (6) (2009) 843–859.

[31] S. Kumar, A study of the supermarket industry and its growing logistics capabilities, International Journal of Retail & Distribution Management 36 (3) (2008) 192–211.

[32] M. Napolitano, Making the move to cross docking: A practical guide to planning, designing, and implementing a cross dock operation, Warehousing Education and Research Council (WERC), 2000.

[33] K. R. Gue, The Effects of Trailer Scheduling on the Layout of Freight Terminals, Transportation Science 33 (4) (1999) 419–428.

[34] C. S. Sung, S. H. Song, Integrated service network design for a cross-docking supply chain network, Journal of the Operational Research Society 54 (12) (2003) 1283–1295.

[35] H. Donaldson, E. L. Johnson, H. D. Ratliff, M. Zhang, Schedule-Driven Cross-Docking Networks, Tech. Rep. 9904, Georgia Institute of Technology, Atlanta, Georgia, USA, 1999.

[36] R. Musa, J.-P. Arnaout, H. Jung, Ant colony optimization algorithm to solve for the transportation problem of cross-docking network, Computers & Industrial Engineering 59 (1) (2010) 85–92.

[37] C. S. Sung, W. Yang, An exact algorithm for a cross-docking supply chain network design problem, Journal of the Operational Research Society 59 (1) (2008) 119–136.

[38] M. Gümüs, J. H. Bookbinder, Cross-docking and its Implications in Location-Distribution Systems, Journal of Business Logistics 25 (2) (2004) 199–228.

[39] V. Jayaraman, A. Ross, A simulated annealing methodology to distribution network design and management, European Journal of Operational Research 144 (3) (2003) 629–645.

[40] A. Ross, V. Jayaraman, An evaluation of new heuristics for the location of cross-docks distribution centers in supply chain network design, Computers & Industrial Engineering 55 (1) (2008) 64–79.

[41] M. Bachlaus, M. K. Pandey, C. Mahajan, R. Shankar, M. K. Tiwari, Designing an integrated multi-echelon agile supply chain network: a hybrid taguchi-particle swarm optimization approach, Journal of Intelligent Manufacturing 19 (6) (2008) 747–761.

[42] K. E. Peck, Operational Analysis of Freight Terminals Handling Less Than Container Load Shipments, Ph.D. thesis, University of Illinois at Urbana-Champaign, 1983.

[43] L. Y. Tsui, C.-H. Chang, A microcomputer based decision support tool for assigning dock doors in freight yards, Computers & Industrial Engineering 19 (1-4) (1990) 309–312.

[44] L. Y. Tsui, C.-H. Chang, An optimal solution to a dock door assignment problem, Computers & Industrial Engineering 23 (1-4) (1992) 283–286.

[45] R. Bermúdez, M. H. Cole, A Genetic Algorithm Approach to Door Assignments in Breakbulk Terminals, Tech. Rep. MBTC 1084, Mack-Blackwell Rural Transportation Center, University of Arkansas, Fayettevilee, Arkansas, USA, 2001.

[46] Y. Cohen, B. Keren, Trailer to door assignment in a synchronous cross-dock operation, International Journal of Logistics Systems and Management 5 (5) (2009) 574–590.

[47] Y. Oh, H. Hwang, C. N. Cha, S. Lee, A dock-door assignment problem for the Korean mail distribution center, Computers & Industrial Engineering 51 (2) (2006) 288–296.

[48] J. J. Bartholdi III, K. R. Gue, Reducing Labor Costs in an LTL Crossdocking Terminal, Operations Research 48 (6) (2000) 823–832.

[49] A. M. Brown, Improving the Efficiency of Hub Operations in a Less-than-Truckload Distribution Network, Master's thesis, Virginia Polytechnic Institute and State University, 2003.

[50] Y. A. Bozer, H. J. Carlo, Optimizing inbound and outbound door assignments in less-than-truckload crossdocks, IIE Transactions 40 (11) (2008) 1007–1018.

[51] V. F. Yu, D. Sharma, K. G. Murty, Door Allocations to Origins and Destinations at Less-than-Truckload Trucking Terminals, Journal of Industrial and Systems Engineering 2 (1) (2008) 1–15.

[52] J.-F. Wang, A. Regan, Real-Time Trailer Scheduling for Crossdock Operations, Transportation Journal 47 (2) (2008) 5–20.

[53] F. Chen, C.-Y. Lee, Minimizing the makespan in a two-machine cross-docking flow shop problem, European Journal of Operational Research 193 (1) (2009) 59–72.

[54] B. Vahdani, M. Zandieh, Scheduling trucks in cross-docking systems: Robust meta-heuristics, Computers & Industrial Engineering 58 (1) (2010) 12–24.

[55] A. R. B. Arabani, S. M. T. F. Ghomi, M. Zandieh, Meta-heuristics implementation for scheduling of trucks in a cross-docking system with temporary storage, Expert Systems with Applications 38 (3) (2011) 1964–1979.

[56] N. Boysen, M. Fliedner, A. Scholl, Scheduling inbound and outbound trucks at cross docking terminals, OR Spectrum 32 (1) (2010) 135–161.

[57] S. Forouharfard, M. Zandieh, An imperialist competitive algorithm to schedule of receiving and shipping trucks in cross-docking systems, The International Journal of Advanced Manufacturing Technology 51 (9) (2010) 1179–1193.

[58] A. R. B. Arabani, S. M. T. F. Ghomi, M. Zandieh, A multi-criteria cross-docking scheduling with just-in-time approach, The International Journal of Advanced Manufacturing Technology 49 (5-8) (2010) 741–756.

[59] B. Vahdani, R. Soltani, M. Zandieh, Scheduling the truck holdover recurrent dock cross-dock problem using robust meta-heuristics, The International Journal of Advanced Manufacturing Technology 46 (5) (2010) 769–783.

[60] R. Soltani, S. J. Sadjadi, Scheduling trucks in cross-docking systems: A robust meta-heuristics approach, Transportation Research Part E: Logistics and Transportation Review 46 (5) (2010) 650–666.

[61] R. Larbi, G. Alpan, P. Baptiste, B. Penz, Scheduling cross docking operations under full, partial and no information on inbound arrivals, Computers & Operations Research 38 (6) (2011) 889–900.

[62] G. Alpan, R. Larbi, B. Penz, A bounded dynamic programming approach to schedule operations in a cross docking platform, Computers & Industrial Engineering 60 (3) (2011) 385–396.

[63] C. R. Rosales, M. J. Fry, R. Radhakrishnan, Transfreight Reduces Costs and Balances Workload at Georgetown Crossdock, Interfaces 39 (4) (2009) 316–328.

[64] M. K. Acar, Robust Dock Assignments at Less-Than-Truckload Terminals, Master's thesis, University of South Florida, 2004.

[65] D. L. McWilliams, P. M. Stanfield, C. D. Geiger, The parcel hub scheduling problem: A simulation-based solution approach, Computers & Industrial Engineering 49 (3) (2005) 393–412.

[66] D. L. McWilliams, P. M. Stanfield, C. D. Geiger, Minimizing the completion time of the transfer operations in a central parcel consolidation terminal with unequal-batch-size inbound trailers, Computers & Industrial Engineering 54 (4) (2008) 709–720.

[67] D. L. McWilliams, Genetic-based scheduling to solve the parcel hub scheduling problem, Computers & Industrial Engineering 56 (4) (2009) 1607–1616.

[68] D. L. McWilliams, Iterative improvement to solve the parcel hub scheduling problem, Computers & Industrial Engineering 59 (1) (2010) 136–144.

[69] D. L. McWilliams, A dynamic load-balancing scheme for the parcel hub-scheduling problem, Computers & Industrial Engineering 57 (3) (2009) 958–962.

[70] A. Chmielewski, B. Naujoks, M. Janas, U. Clausen, Optimizing the Door Assignment in LTL-Terminals, Transportation Science 43 (2) (2009) 198–210.

[71] A. Lim, H. Ma, Z. Miao, Truck Dock Assignment Problem with Time Windows and Capacity Constraint in Transshipment Network Through Crossdocks, in: Computational Science and Its Applications - ICCSA 2006, International Conference, Glasgow, UK, May 8-11, 2006, Proceedings, Part III, vol. 3982 of *Lecture Notes in Computer Science*, Springer, 688–697, 2006.

[72] A. Lim, H. Ma, Z. Miao, Truck Dock Assignment Problem with Operational Time Constraint Within Crossdocks, in: Advances in Applied Artificial Intelligence, 19th International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems, IEA/AIE 2006, Annecy, France, June 27-30, 2006. Proceedings, vol. 4031 of *Lecture Notes in Artificial Intelligence*, 262–271, 2006.

[73] Z. Miao, A. Lim, H. Ma, Truck dock assignment problem with operational time constraint within crossdocks, European Journal of Operational Research 192 (1) (2009) 105–115.

[74] N. Boysen, Truck scheduling at zero-inventory cross docking terminals, Computers & Operations Research 37 (1) (2010) 32–41.

[75] M. Shakeri, M. Y. H. Low, Z. Li, A Generic Model for Crossdock Truck Scheduling and Truck-to-Door Assignment Problems, in: Proceedings of the 6th IEEE International Conference on Industrial Informatics (INDIN 2008), 857–864, 2008.

[76] Z. P. Li, M. Y. H. Low, M. Shakeri, Y. G. Lim, Crossdocking planning and scheduling: Problems and algorithms, Tech. Rep. STR_V10_N3_06_POM, Singapore Institute of Manufacturing Technology, 2009.

[77] S. Sandal, Staging approaches to reduce overall cost in a crossdock environment, Master's thesis, University of Missouri, 2005.

[78] Y. H. Lee, J. W. Jung, K. M. Lee, Vehicle routing scheduling for crossdocking in the supply chain, Computers & Industrial Engineering 51 (2) (2006) 247–256.

[79] C.-J. Liao, Y. Lin, S. C. Shih, Vehicle routing with cross-docking in the supply chain, Expert Systems with Applications 37 (10) (2010) 6868–6873.

[80] M. Wen, J. Larsen, J. Clausen, J.-F. Cordeau, G. Laporte, Vehicle routing with cross-docking, Journal of the Operational Research Society 60 (12) (2009) 1708–1718.

[81] G. A. Álvarez-Pérez, J. L. González-Velarde, J. W. Fowler, Crossdocking—Just in Time scheduling: an alternative solution approach, Journal of the Operational Research Society 60 (4) (2009) 554–564.

[82] M. Stickel, Planung und Steuerung von Crossdocking-Zentren, Dissertation, Institut für Fördertechnik und Logistiksysteme, Fakultät für Maschinenbau, Universität Karlsruhe, 2006.

[83] G. M. Magableh, M. D. Rossetti, S. Mason, Modeling and analysis of a generic cross-docking facility, in: Proceedings of the 37th conference on Winter simulation, 1613–1620, 2005.

[84] A. Lim, Z. Miao, B. Rodrigues, Z. Xu, Transshipment through Crossdocks with Inventory and Time Windows, Naval Research Logistics 52 (8) (2005) 724–733.

[85] Z. Miao, K. Fu, Q. Fei, F. Wang, Meta-heuristic Algorithm for the Transshipment Problem with Fixed Transportation Schedules, in: New Frontiers in Applied Artificial Intelligence, 21st International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems, IEA/AIE 2008, Wroclaw, Poland, June 18-20, 2008, Proceedings, vol. 5027 of *Lecture Notes in Artificial Intelligence*, 601–610, 2008.

[86] P. Chen, Y. Guo, A. Lim, B. Rodrigues, Multiple crossdocks with inventory and time windows, Computers & Operations Research 33 (1) (2006) 43–63.

[87] H. Ma, Z. Miao, A. Lim, B. Rodrigues, Crossdocking distribution networks with setup cost and time window constraint, Omega 39 (1) (2011) 64–72.

[88] S.-L. Tang, H. Yan, Pre-distribution vs. post-distribution for cross-docking with transshipments, Omega 38 (3-4) (2010) 192–202.

[89] M. Rohrer, Simulation and cross docking, in: C. Alexopoulos, K. Kang, W. R. Lilegdon, D. Goldsman (Eds.), Proceedings of the 27th conference on Winter simulation, 846–849, 1995.

[90] R. F. Babiceanu, F. F. Chen, Development and applications of holonic manufacturing systems: a survey, Journal of Intelligent Manufacturing 17 (1) (2006) 111–131.

[91] P. Leitão, F. Restivo, ADACOR: A holonic architecture for agile and adaptive manufacturing control, Computers in Industry 57 (2) (2006) 121–130.

[92] D. C. McFarlane, S. Bussmann, Developments in Holonic Production Planning and Control, International Journal of Production Planning and Control 11 (6) (2000) 522–536.

[93] L. Monostori, J. Váncza, S. R. T. Kumara, Agent-Based Systems for Manufacturing, CIRP Annals - Manufacturing Technology 55 (2) (2006) 697–720, CIRP keynote paper.

[94] P. Valckenaers, H. Van Brussel, Holonic Manufacturing Execution Systems, CIRP Annals - Manufacturing Technology 54 (1) (2005) 427–432.

[95] H. Van Brussel, J. Wyns, P. Valckenaers, L. Bongaerts, P. Peeters, Reference Architecture for Holonic Manufacturing Systems: PROSA, Computers in Industry 37 (3) (1998) 255–274, special issue on intelligent manufacturing systems.