Computers & Industrial Engineering 91 (2016) 197–204

Contents lists available at ScienceDirect

Computers & Industrial Engineering

journal homepage: www.elsevier.com/locate/caie

An improved mathematical model and a hybrid metaheuristic for truck scheduling in cross-dock problems

M. Keshtzari [a], B. Naderi [b,*], E. Mehdizadeh [c]

[a] Department of Industrial Engineering, Texas Tech University, Lubbock, TX 79409, USA
[b] Department of Industrial Engineering, Faculty of Engineering, Kharazmi University, Tehran, Iran
[c] Department of Industrial Engineering, Qazvin Branch, Islamic Azad University, Qazvin, Iran

ARTICLE INFO

Article history:
Received 26 January 2015
Received in revised form 6 November 2015
Accepted 7 November 2015
Available online 17 November 2015

Keywords:
Supply chain management
Cross-dock
Mixed integer programming
Particle swarm optimization algorithm
Metaheuristic

ABSTRACT

Recently, several papers study the problem of scheduling inbound and outbound trucks in cross-dock systems. There is only one attempt to mathematically model the problem. Unfortunately this model is ineffective due to its size complexity. Moreover, although different metaheuristics are proposed in the literature, they are not tailored-made for this problem. In this paper, a new mixed integer programming model is formulated for the problem. Using commercial optimization solvers, the performance of the proposed model is compared with the available model to solve small instances. To solve larger instances, a particle swarm optimization hybridized with a simulated annealing is proposed. Using Taguchi method, the proposed algorithm is tuned. Then, it is evaluated against two other available metaheuristics (genetic algorithm and electromagnetism-like metaheuristic) in the literature.

1. Introduction

Today's distribution networks focus on making operations more efficient. Companies attempt to reduce costs through inventory reduction at each level of operations, including distribution costs. One of the storage strategies for controlling distribution costs and increasing the level of customer satisfaction is the cross-docking system. In a distribution system, the main operations are receiving, storing, retrieving, and shipping. If any of these five operations is improved or eliminated, costs can be decreased; as a result, the productivity can be increased. The cross-docking system has the potential to eliminate two costly operations of storing and retrieving. Thereby, decreasing the total cost of the distribution center.

Cross-docking is a conception of storage management in which products move from the receiving dock to the shipping dock without being stored for a long time in a warehouse or a distribution center (Yu & Egbelu, 2008). In general, the cross-docking system works efficiently for companies that distribute large quantities of products or service a large number of stores. The advantages of cross-docking system include the improvement of customer response time in supply chains and reductions in labor costs, lead time, and warehouse space needs (Boloori Arabani, Zandieh, & Fatemi Ghomi, 2011). Many companies have benefitted from cross-docking, including global companies like Walmart (Gue, 2002) and Toyota (Liao, Egbelu, & Chang, 2012).

There are problems in cross-docking that include assignment problems, and have been examined in the literature. Tsui and Chang (1992) propose a bilinear program model to assign trucks to docks, with the objective function of minimizing weighted distances between inbound and outbound trucks. Rohrer (1995), without discussing any implementation cases, investigates the effects of using simulation on the success of a cross-docking system design. Mosheiov (1998) surveys the pickup and delivery problem with a special type of vehicle routing problem, and proposes a mathematical model to minimize transportation costs and maximize vehicle efficiency. He further applies two heuristics to determine appropriate solutions in a reasonable time. Apte and Viswanathan (2000) propose a framework to understand and design cross-docking systems. Lau, Sim, and Teo (2003) propose a tabu search algorithm for a vehicle routing problem with a particular number of trucks to minimize transportation costs during a specified time window. Bartholdi and Gue (2004) investigate the best shape for a cross-docking system. By conducting computational experiments and evaluating each shape by different characteristics of the system, Bartholdi and Gue conclude that the best shape depends on the sizes of facilities and the pattern of products flow.

Chen and Song (2006) study the cross-docking scheduling problem, by considering the total completion time for just-in-time logistics. They propose a number of heuristic branch and bound

* Corresponding author. Tel.: +98 910 2152804.
E-mail address: bahman.naderi@aut.ac.ir (B. Naderi).

http://dx.doi.org/10.1016/j.cie.2015.11.006
0360-8352/

algorithms that incorporate different characteristics of the problem. Ma and Chen (2007) study cross-docking scheduling problem, in which the objective function is to minimize total completion time, and propose a dynamic programming algorithm to solve the problem. Yu and Egbelu (2008) consider a cross-docking system with one receiving dock and a temporary storage unit in front of a single shipping dock. The objective function is to find an appropriate sequence of inbound and outbound trucks to minimize total completion time. Additionally, this model simultaneously determines product assignments from inbound trucks to outbound trucks. For the first time, Yu and Egbelu propose a mixed-integer model for the problem that solves small-size instances, but not large-size instances. They develop heuristic algorithms to increase solution efficiency.

Chen and Song (2009) characterize the cross-docking scheduling problem as a two-stage flow-shop problem with parallel machines. Each stage is either a receiving or a shipping dock. Receiving or shipping docks and inbound or outbound trucks are machines and a set of jobs, respectively. They propose a mixed-integer program to minimize makespan, and use CPLEX to solve small-size instances. Subsequently, they develop four heuristic algorithms to solve the problem. Similar to Yu and Egbelu (2008), Vahdani and Zandieh (2010) consider another type of cross-docking system in which temporary storage is not allowed. They propose several metaheuristic algorithms for the cross-docking scheduling problem in which the objective is the minimization of makespan and also develop three metaheuristic algorithms to solve the problem. Boloori Arabani, Fatemi Ghomi, and Zandieh (2010) study a just-in-time cross-docking scheduling problem, similar to that of Yu and Egbelu (2008). In this multi-criteria problem, they consider earliness and tardiness as two main criteria. To solve the model, they establish three metaheuristic algorithms. Boloori Arabani, Fatemi Ghomi, and Zandieh (2011) succeed the survey of Yu and Egbelu (2008), proposing five metaheuristic algorithms and comparing them with the heuristic method proposed by Yu and Egbelu (2008).

As reviewed in the literature, this problem is not well-studied and there has only been one attempt to formulate the problem (Yu & Egbelu, 2008). All the preceding papers studying the problem assume the same mathematical model and only develop solution algorithms. Although there are different metaheuristics in the literature, none are tailored-made for this problem; instead, they all use general operators. In this study, we first develop an efficient mixed-integer programming model and compare its performance with the existing model. Subsequently, we propose a hybrid metaheuristic algorithm to solve the problem in large-sizes. This algorithm is based on particle swarm optimization and simulated annealing. The performance of the proposed algorithm is then compared with two available metaheuristics, the electromagnetism algorithm (Vahdani & Zandieh, 2010) and the genetic algorithm (Boloori Arabani et al., 2011). These two algorithms outperform several other algorithms in the studies in which they are proposed.

The remainder of this paper is organized as follows: in Section 2, we propose a new mixed-integer programming model for the problem. We present and explain metaheuristic algorithms in Section 3. In Section 4, we discuss parameter setting. We provide computational results in Section 5. In Section 6, we summarize our findings and suggest possible trends for future studies.

2. Problem description and mixed-integer programming formulation

Developing mathematical programing models is a key step to studying an optimization problem. In this section, we develop a more effective mathematical model than the available model created by Yu and Egbelu (2008). In this problem, the cross-dock has one receiving dock and one shipping dock. A temporary storage is located in front of the shipping dock. If a product that arrives at the shipping dock is not needed by the outbound truck already located at the dock, the product can be temporarily stored in the storage until an outbound truck that needs this product arrives to the shipping dock. Both inbound and outbound trucks must stay in the docks until they finish loading or unloading the products (this is called the dock holding pattern). As a result, an inbound truck cannot leave the receiving dock before unloading all of its products. Similarly, an outbound truck cannot leave the shipping dock before loading all of the required products.

The objective function of this model is to determine the best sequence of inbound and outbound trucks to minimize the maximum completion time (makespan). Simultaneously, the product routing (i.e., the assignment of products from inbound trucks to outbound trucks) is also obtained.

The following notations are used for the mathematical model:

n	The number of products
q	The number of inbound trucks
u	The number of outbound trucks
j	Index for products where $j = \{1, 2, \ldots, n\}$
$i,\ l$	Index for inbound trucks where $i = \{1, 2, \ldots, q\}$
$t,\ k$	Index for outbound trucks where $t = \{1, 2, \ldots, u\}$
r_{ij}	The number of units of product j loaded in inbound truck i
s_{tj}	The number of units of product j needed for outbound truck t
d	The truck changeover time
v	The transportation time of products from inbound dock to outbound dock

The continuous variables applied in the model are:

$E_{1,l}$	Continuous variable for starting time of unloading the inbound truck in position l
$E_{2,l}$	Continuous variable for finishing time of unloading the inbound truck in position l
$F_{1,k}$	Continuous variable for starting time of loading the outbound truck in position k
$F_{2,k}$	Continuous variable for finishing time of loading the outbound truck in position k
C_{max}	Continuous variable for makespan

Integer and binary variables used in the model are respectively as follows:

$X_{j,l,k}$	Integer variable for the number of units of product j transferred from inbound position l to outbound position k
$Y_{l,k}$	Binary variable taking value 1 if any product transferred from inbound position l to outbound position k, and 0 otherwise
$Z_{i,l}$	Binary variable taking value 1 if inbound truck i is assigned to position l, and 0 otherwise
$W_{t,k}$	Binary variable taking value 1 if outbound truck t is assigned to position k, and 0 otherwise