A deterministic model for the transshipment problem of a fast fashion retailer under capacity constraints

Siamak Naderi [a,*], Kemal Kilic [a], Abdullah Dasci [b]

[a] *Faculty of Engineering and Natural Sciences, Sabanci University, Orta Mahalle, Tuzla, Istanbul, Turkey*
[b] *Sabanci Business School, Sabanci University, Orta Mahalle, Tuzla, Istanbul, Turkey*

ARTICLE INFO

ABSTRACT

In this paper we present a novel transshipment problem for a large apparel retailer that operates an extensive retail network. Our problem is inspired by the logistics operations of a very large fast fashion retailer in Turkey, LC Waikiki, with over 475 retail branches and thousands of products. The purpose of transshipments is to rebalance stocks across the retail network to better match supply with demand. We formulate this problem as a large mixed integer linear program and develop a Lagrangian relaxation with a primal–dual approach to find upper bounds and a simulated annealing based metaheuristic to find promising solutions, both of which have proven to be quite effective. While our metaheuristic does not always produce better solutions than a commercial optimizer, it has consistently produced solutions with optimality gaps lower than 7% while the commercial optimizer may produce very poor solutions with optimality gaps as high as almost 300%. We have also conducted a set of numerical experiments to uncover implications of various operational practices of LC Waikiki on its system's performance and important managerial insights.

1. Introduction

Due to its impact on revenues, costs, and more importantly, on service levels, logistics management has become increasingly critical in the apparel industry (Kiesmüller and Minner, 2009). As consumers demand greater product variety and higher levels of responsiveness at lower prices, effective management of logistics activities arises as a key competitive advantage for the retailers in this industry. The main challenges faced by these retailers are short selling seasons and unpredictable demands. Since forecasts are mostly inaccurate, firms usually have either excess inventories that are sold at markdown prices or stock-outs that lead to lost sales. The problem is exacerbated with short selling seasons which prevent firms to replenish their stocks. Therefore, an effective logistics strategy is key to avoid both of these undesirable outcomes.

Logistics decisions of apparel retailers include initial ordering before the season begins, allocation to the branches at the beginning of the season, and eventually phasing-out of the products at the end of the selling season. Increasingly, however, retailers are also practicing what is called "transshipment" or "transfer" policies, which involve the reallocation of products among retail branches in mid-season (Li et al., 2013). These policies help retailers to reduce stock-outs as well as excess inventories. This is the issue that is addressed in this paper.

The problem that we consider here is inspired by the logistics operations at the largest apparel retailer in Turkey, LC Waikiki, which has positioned itself as a "fast fashion" retailer. The term fast fashion used to refer to inexpensive designs that appeared on catwalks and were quickly moved to store shelves. Fast fashion items based on the most recent trends have shaped mass-merchandized clothing collections. Therefore, mass-merchandise retailers compete to introduce latest fashion trends in their collections. Although the term was first used in the US in the 1980s, the expression did not receive worldwide adoption until popularized by the Spanish-based apparel giant Zara. The crucial issue in fast fashion is providing inexpensive collections that also respond to fast changing consumer tastes and trends. Therefore, the entire fast fashion supply chain must be sufficiently agile to operate with products for which life cycles are measured not in months but rather in weeks.

On the plus side, the speed at which fast fashion moves tends to help retailers avoid markdowns. Typically, these retailers do not place very large orders months before the actual selling season, but rather work with smaller initial orders and renew collections more frequently. On the negative side, however, the fast-paced environment calls for higher turnover and more frequent introduction of new designs, a setting that necessitates shorter design and production lead times. As a result, companies need to rely on more expensive local sources and

---

accommodate large design teams. This fast-paced environment also creates new logistics challenges for retailers: When will these products be replaced? Should they be completely removed from the stores or kept at display at select stores? What will happen to the leftover items; reintroduced elsewhere, sold at discount, or simply written-off? Fast fashion companies need to deal with these issues much more frequently than traditional retailers.

Facing such challenges, leading fast fashion companies such as Zara and another Spanish company, Mango, the Japanese World Co., and Swedish H&M have built supply chains aiming at quickly responding to consumers' changing demands while decreasing the excess inventories at branches and hence, lowering costs (Caro and Gallien, 2007). For instance, Zara developed a decision support system featuring demand updating and a dynamic optimization module for initial shipment decisions to avoid stock-outs as well as excess inventories (Gallien et al., 2015). In addition to correct initial shipment decisions, the transfer or transshipment decisions among retail locations are also instrumental to reduce stock-outs and excess inventories.

The main benefit of transfer actions is better matching inventory and demand at different locations. It uses up-to-date sales information and inventory data and redistribute available inventory among the retail locations. Due to socio-economic and geographical differences among retailer locations, it is possible that a product sells very well in some stores while less so in others. Transfer actions can be adopted as a tool to increase the inventory levels at receiver stores while providing extra shelf space at sender stores. This action can be adopted by bypassing the central depot to facilitate the quick movement of merchandise. As a result, the revenues are increased while costs are reduced as compared to a system where no transshipment is utilized (Tagaras, 1989). There are a number works in the literature that describe how retailers take advantage of transfers to improve their performances. For example, Archibald et al. (2009, 2010) address transshipment issues at a tire retailer that has a network of 50 stores. In another work, Hu and Yu (2014) present a proactive transshipment problem for a famous fashion brand in China that has network for 43 retailers in Shanghai. The problem that we introduce here is motivated by the largest apparel retailer in Turkey.

In the next section, we provide a detailed background of our problem that includes the transfer practices at the company that motivated this work and a detailed description of the problem setting. In Section 3 we give a brief literature review. Section 4 presents the progressive development of the mathematical model. Section 5 presents our solution methods that include a Lagrangian relaxation based upper bounding method and simulated annealing based metaheuristic to find good feasible solutions. Section 6 reports on our numerical experiments followed by a few concluding remarks in Section 7.

2. Background

Textile is one of the key sectors in the Turkish economy in terms of GDP, domestic employment, and exports. Textile accounts for 10% of the Turkish GDP and 20% of employment in the manufacturing sector.[1] In 2016 Turkey exported around 15 Billion USD, mainly to the European Union countries and was ranked as the 6th biggest textile exporting country (see Fig. 1).[2] LC Waikiki, which has provided motivation to this work, is the largest textile retailer in Turkey with significant international presence.

LC Waikiki was founded in 1988 in France by a French designer and his friend. The LC Waikiki brand name is created by adding the word Waikiki, a famous beach in Hawaii, to LC, the abbreviation of the French word "Les Copains" meaning "friends". TEMA, a Turkey based group which was then a major supplier of the company, bought the LC Waikiki brand in 1997 undertook a major restructuring that included focusing on domestic market. In the same year, the group entered the Turkish fashion retail market with 21 stores. In 2009, it opened its first international store in Romania since the TEMA group had purchased the brand. Over the years, the group has followed an aggressive expansion strategy both domestically and internationally. Today, LC Waikiki has more than 500 stores in 46 countries in Asia, Africa, and Europe, in addition to over 475 stores in Turkey. In 2011, LC Waikiki became the leader of the "Ready-to-Wear" market in Turkey and remains as the largest apparel retailer in terms of sales as well as the number of stores. Fig. 1 depicts LC Waikiki's phenomenal growth in terms of the total number of stores over the years.

LC Waikiki has a highly centralized order planning and logistics system in which all initial orders and subsequent distribution decisions are made by the headquarters. New merchandise is received at a single central warehouse located in Istanbul, which then distributes essentially the entire amount to the retail branches (there are varying practices for international stores which are excluded from the consideration in this study). The retail practice at LC Waikiki can be considered as fast fashion in that it aims to keep items in stores only for about six to eight weeks. During this period, if the sales realize below expectations, they may reduce prices or if the sales display disparities across the stores, they may utilize transfers among the stores. Finally, at the end of their shelf-life, products are returned to the central warehouse and later sent to outlet stores (about 40 of the 475 stores are designated as outlet stores) or simply given away to charities. Stock-outs and excess inventories are critical issues at LC Waikiki as in any fast fashion company due to forecast errors. Since LC Waikiki initially distributes all of the items to stores, transfer remains essentially as the only tool to deal with these issues by rebalancing inventories across the retail network. It is these transfer decisions that is the subject of this paper.

Currently, a group at the headquarters manages transfer decisions. This group utilizes a mathematical model accompanied with some pre- and post-processing activities. However, we cannot disclose the precise nature of the model and the activities due to proprietary nature of these information. After transfer solutions are obtained, orders are automatically generated and transmitted to the stores. Store employees collect the products that have been chosen for transfer from the shelves and move them to a storage room. In the storage room, products are put in the boxes, each destined to a specific store without any re-assortments. Since the storage room capacities are limited, stores cannot to transfer more than what they can hold at their storage room. Once boxing is finished, the logistics company picks up the boxes and delivers them to their destinations. The boxes are ideally delivered before the weekend so that the transferred items can be put on shelves for the weekend sales.

Although our work is motivated by LC Waikiki's logistics operations, we believe many of the features of our model would resonate with issues fast fashion retailers need to consider. In our model, we maximize a measure of the total profit which is the total revenue less the total logistics cost that includes transportation, handling, and inventory holding costs. We also include a number of operational constraints that represent the real practice of the company. For example, we consider a centrally managed system where stores have no control over the decisions, i.e., they may not refuse the transfer decisions. This is valid particularly for firms that own their stores and manage them centrally. We also restrict the total number of items and the total number of stores to which each store can make shipments. Both of these constraints are justified by the limited number of employees in the stores and sizes of the storage rooms. Furthermore, in our model once a product is decided to be transferred from one store to the other, the entire stock (all the available sizes) is sent to the same store. LC Waikiki justifies this practice by the simplicity of the picking operations, which otherwise would be too labor intensive. Here, we will also investigate the effects of these restrictions on system performance.

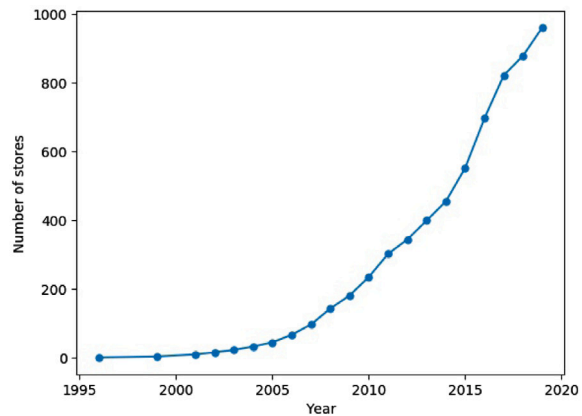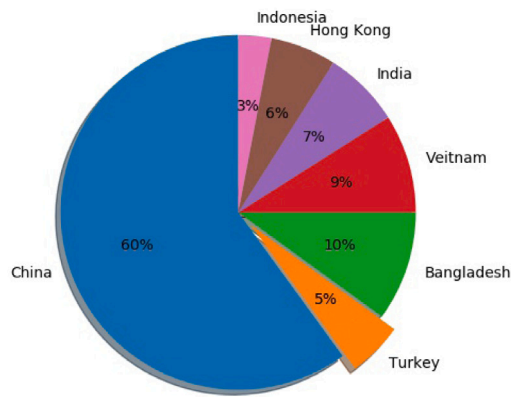[1] blog.tcp.gov.tr.
[2] www.wikipedia.com.

Fig. 1. Left: Global export market share, Right: Growth in the total number of stores.

There are a number of issues relevant to the fashion logistics decisions that we leave out of the scope of this work: (i) Initial allocation decisions, (ii) Uncertainty in demand, and (iii) Dynamic nature of the decision making process. At LC Waikiki the initial shipment decisions are made after a pilot sales experiment in which they obtain sales information from about 30 stores. They then make initial allocations in which they essentially distribute the entire stock to the stores. Certainly, the option of transfers might impact the initial allocation. However, we believe that the impact is small due to two aspects in this logistics system. First, the company has a policy to allocate almost all of the available inventory to the stores keeping none at the central depot. Therefore, the firm cannot use central depot for reallocation of products. Second, since the company has flat transportation cost rate independent of origin–destination pair, regional risk pooling effect becomes irrelevant. Therefore, the impact of subsequent transfer practice on initial allocation decisions has lessened. Demand uncertainty is always a concern particularly in the fashion industry, and in fact, it is the demand uncertainty that makes the transfer problem relevant. However, at the time a product is considered for transfer, there is demand information for at least for a couple of weekends. Therefore, the company is able to make much more accurate demand forecasts after this initial sales information, as compared to the time the initial allocation decisions are made. Finally, the transfer problem ideally should consider the fact that transfer decisions are made every week and hence, there are subsequent recourse opportunities. However, considering a multi-stage decision environment under demand uncertainty is simply beyond analysis for the sizes that we envision, particularly with complicating operational constraints. Instead, we envision a setting where the firm makes demand forecast until a product is planned to stay on shelves and the transfer problem is solved on a rolling-horizon basis. This setting, we believe, is a reasonable compromise given the other complexities of the system. Similarly, initial replenishment decisions are also important and they would be impacted by subsequent transfer options. However, considering transshipment and replenishment decisions jointly would also be extremely difficult considering the scale of our problem and particular operational constraints.

We have also assumed that each product's shelf life is known. This assumption is justified by the practice of the company where they keep merchandize for about six to eight weeks. Decision to continue displaying products on the shelves or removing them involves a number of other factors to consider. It requires information on new product designs as well as space considerations at the stores for different merchandize groups. These issues are also rather involved and therefore, kept out of the current study, but certainly worthwhile to consider in the future.

3. Related literature

There is a vast literature as far back as the 1950's on lateral transshipment or, as we call here, transfer issues. Although both terms commonly describe the decisions considered here and we use them interchangeably, the term transshipment has a wider meaning and usage. Time and again, various studies have shown that transfer option between retailers improves supply chain performance in terms of costs, revenues, and service levels. For example, Tagaras (1989) shows that utilizing transfer in a system with two retail locations leads to significant cost reductions. Although transfers considerably increase transportation cost, systems with these options are superior to systems without them (Banerjee et al., 2003). Furthermore, transfers enhance customer service levels without the burden of carrying extra safety stock at retail locations (Burton and Banerjee, 2005).

There are essentially two types of transfers: emergency or reactive transfers and preventive or proactive transfers, which are differentiated mainly with respect to their timing (Lee et al., 2007; Paterson et al., 2011; Seidscher and Minner, 2013; Ahmadi et al., 2016). Reactive transfer refers to responding to realized stock-outs at a retail location by using available inventory at another location whereas proactive transfer refers to redistribution of inventories among locations before the actual demand is realized. The literature can be classified primarily along this dimension, although there are also works that consider them jointly.

Perhaps the earliest work that considers reactive transfers is by Krishnan and Rao (1965) who study a centralized one-echelon inventory system with the objective of minimizing the total cost through transfers. One of the main motivations for reactive transfer models comes from spare parts distribution systems for repairable items, as exemplified by one of the more notable earlier works by Lee (1987) who studies a single-echelon model, which is then extended by Axsäter (1990) to a two-echelon system. More recent works on spare parts systems can be attributed to van Wijk et al. (2019) who consider a two-location system with lateral transshipment as well as an outside emergency option and Boucherie et al. (2018) who consider a complex two-echelon inventory system with multiple local warehouses.

Models with reactive transfer policies have also been studied for non-repairable items. A notable contribution is due to Robinson (1990), who provides structural results for a two-retailer system and develops a heuristic for the initial ordering decisions considering the subsequent transshipments. Herer et al. (2006) extend this work by considering more general cost structures and Özdemir et al. (2013) extend it considering capacity constraints on the transportation network. More recently, transshipment policies in systems with perishable items have also attracted research (see for example, Nakandala et al., 2017; Dehghani and Abbasi, 2018 for such recent works).

Proactive transfer is based on the concept of inventory rebalancing and is mostly utilized in periodic review inventory control framework. Allen (1958) provides perhaps the earliest model that considers proactive transfers in a single-period setting, which is then generalized by Das (1975) who also considers the initial replenishment decision. There are also models that include the timing of the transshipment as decision in a dynamic setting (Agrawal et al., 2004; Tiacci and Saetta, 2011) as well as in a static setting (Kiesmüller and Minner, 2009).

Although the type of transfers may be dictated by operational conditions of the setting, proactive transfer policies are found to be superior to purely reactive policies both in terms of costs and stock-out levels (see for example, Banerjee et al., 2003; Burton and Banerjee, 2005). In some settings, however, companies may also have opportunities to implement these policies jointly (see for example, Lee et al., 2007 for such a model). Finally, although all of the works mentioned above and majority of research on the transshipment issues, assume that the systems are centrally operated decentralized systems where retailers might refuse transshipment requests have also attracted research recently (see for example, Çömez et al., 2012; Li et al., 2013).

As we have noted earlier, the literature on transshipment issues is vast with considerable growth in the last two decades. Since reviewing this voluminous literature is not possible here, we have only offered a very selective review. Aside from the types of the transfer (i.e., reactive vs. proactive), the literature on transshipment is also divided along two other important dimensions: Whether the models consider only transshipment decisions or jointly with replenishment decisions and whether they consider multiple locations or just two locations. We have classified aforementioned works and few others with respect to these characteristics as shown in Table 1. We choose to put some classic and some more recent ones, but it is still far from portraying a complete picture. We refer the reader to a somewhat older, but an excellent review by Paterson et al. (2011) who also provide a more thorough classification and a comprehensive review up to its publication date.

Despite many simplification attempts, solving transfer problem to optimality remains a challenge. Even in the presence of many simplifying assumptions such as single product, single-period, limited number of retail locations, static transfer timing and so on, past works can only provide approximate solutions. The problem that we present here considers proactive transfers, but since it is motivated by the actual logistics operations at a large fashion retailer, it has many complexities that would be quite challenging to resolve under demand uncertainty or in a dynamic fashion. Therefore, we need to make restrictive assumptions along these dimensions. Certainly we are not alone in this respect; there are numerous other works that consider deterministic demand for transshipment models. Not surprisingly, these works also contain other complicating factors. For example, Herer and Tzur (2001, 2003) in their multi-period model consider fixed ordering costs in transshipments; Lim et al. (2005) and Ma et al. (2011) study transshipment decisions via cross-docking locations under time windows; Qi (2006) considers transshipment and production scheduling decisions jointly; Lee (2015) considers concave production and transportation costs; Coelho et al. (2012), Mirzapour Al-e hashem and Rekik (2014), and Peres et al. (2017) consider routing issues alongside transshipments; Rahmouni et al. (2015) and Feng et al. (2017) develop EOQ-based delivery scheduling models with transshipment while considering multiple products and resource constraints. Our setting too has a few operational practices that force us to model a static and deterministic problem.

4. Problem description and model formulation

We consider a retail logistics system that consists of a number of retail stores, each carrying a set of products of different sizes (SKUs). The firm has the precise stock information; that is, how many of each SKU the stores have and the projected demands of each SKU at each location during the remainder of the sales period. The problem is how

to reallocate (some of) the products to maximize a profit measure that is total revenue less transfer, handling, and inventory holding costs.

The firm has a single price policy in that the same price is applied to a product at all locations, which is indeed the practice of many retail chains and particularly of LC Waikiki. Each product also has a fixed transfer cost regardless of the origin–destination pair. This assumption is also motivated by the practice at LC Waikiki which has outsourced the transportation operations to a logistics company. The transfers are made by standard sized boxes for which LC Waikiki pays a fixed amount regardless of its contents and the locations of the sender and receiver stores. Since the number of products that fit in a box depends on the volume of the product, transportation cost differs for each product, but not on origin–destination pairs. The transfer cost can be estimated by adding the handling cost to the transportation cost for each product. However, none of these assumptions are really essential either for modeling or for our solution method and they can easily be relaxed.

We also assume that transfer time has no effect on the sales. The main purpose is to finalize the delivery of transfer items before the weekend where the bulk of the sales materialize. Hence, delivering a day earlier or later presumably does not make much difference, as long as the products arrive for the weekend. Furthermore, the geography of Turkey does not allow wide variations in transfer times, but we also recognize that considering transfer time effects would be valuable in some settings. Finally, we assume that there are no replenishment opportunities from the central warehouse at the time of the transfer decisions. Since the company has a policy to allocate the entire inventory of a product to the stores at the beginning rather than keeping some at the warehouse, this assumption is well justified. As another operational practice, they do not consider a second replenishment option. This is a common practice among the fast-fashion retailers whose business practices involve speedy turnover of designs as exemplified by Zara's practice (see for example, Gallien et al., 2015).

In addition to these requirements, we assume a single-period setting and deterministic demand. At LC Waikiki, most of the sales occur at weekends and therefore, the inventory levels of each product are updated at the beginning of each week. Likewise, demand forecasts are also revised after observing weekend sales. As a result, LC Waikiki, solves the transshipment problem once a week which allows us to consider single-period assumption to decrease the complexity of the problem. Deterministic demand is assumed since the forecast from the company is fairly accurate. After two or three weekend sales, the company can have a fairly good idea about the demand in the rest of the products' shelf lives. It is stated that their forecast error is below 15%. Many papers related to fast-fashion also state that forecast errors are considerably smaller towards the end of shelf lives of products (see for example, Caro and Gallien, 2010). Finally, as mentioned earlier, the company has a few operational practices that we include in our model: If a product is transferred from a store, its entire available inventory (all SKUs) is shipped to a single store. Also, there are limits on the total number of SKUs that can be transferred from a store and the number of different destinations to which a store can make transfers. All these assumptions could be relaxed or generalized, but we choose to stay with the company practices as much as possible.

As we will see shortly, without the aforementioned operational constraints, the problem can simply be formulated as a profit-maximizing transportation problem, which can easily be solved as a linear program. We are also ensured integer solutions if the demand and inventory values are integers. When we add the restriction on the total number of SKUs that can be transferred from a store, the problem can still be solved as a linear program. When we further add the restriction on the number of stores that a store can ship to, however, we need to introduce binary variables to keep track of whether a shipment is made from one store to another. Finally, when we include the single-destination

Table 1
Main characteristics of reviewed transfer-related papers.

	Replenishment	Single period		Multiple periods	
		2 retailers	Multiple retailers	2 retailers	Multiple retailers
Proactive	**Yes**	Das (1975) Tagaras and Vlachos (2002)	Karmarkar and Patel (1977) Hoadley and Heyman (1977)	Tiacci and Saetta (2011) Abouee-Mehrizi et al. (2015)	Diks and De Kok (1996) Ahmadi et al. (2016) Feng et al. (2017)
	No	Kiesmüller and Minner (2009) Li et al. (2013)	Allen (1958) Agrawal et al. (2004)	Dan et al. (2016)	Bertrand and Bookbinder (1998) Banerjee et al. (2003) Burton and Banerjee (2005) Acimovic and Graves (2014) Peres et al. (2017)
Reactive	**Yes**	Herer and Rashit (1999) Minner and Silver (2005) Liao et al. (2014) Olsson (2015) Dehghani and Abbasi (2018)	Lee (1987) Axsäter (1990) Herer et al. (2006) Johansson and Olsson (2018) Boucherie et al. (2018)	Archibald et al. (1997) Herer and Tzur (2001) van Wijk et al. (2019)	Archibald et al. (2009) van Wijk et al. (2012) Özdemir et al. (2013)
	No	Herer and Rashit (1995) Shao et al. (2011) Liao et al. (2014)	Nonås and Jörnsten (2007) Hu and Yu (2014) Patriarca et al. (2016) Bhatnagar and Lin (2019)	Tagaras (1989) Çömez et al. (2012) Shao (2018)	Robinson (1990) Banerjee et al. (2003) Burton and Banerjee (2005) Dijkstra et al. (2017)

constraint (i.e., when a product is shipped from one store to another, all the SKUs of the product must be shipped) the problem becomes much more difficult because we now also need to define a much larger set of binary decision variables to keep track of shipments between stores.

We now give the preliminary definitions, followed by the formulation of the model. We start with the base model without considering the operational requirements of the company and progressively extend the model by adding each of these constraints. We call two stores as "connected" if at least one product is transferred from one store to the other.

Sets and indices

$i, j \in I$: Set of stores,

$p \in P$: Set of products,

$k \in K_p$: Set of sizes for each product $p \in P$.

Parameters

s_{ipk} : Stock level of size k of product p at store i,

d_{ipk} : Demand of size k of product p at store i,

r_p : Unit net revenue of product p,

c_p : Unit transfer cost of product p,

h_p : Holding cost of product p.

Decision variables

x_{ijpk} : Amount of size k of product p transferred from store i to store j,

z_{ipk} : Sales of size k of product p at store i,

w_{ipk} : Amount of size k of product p store i has after the transfers.

Relaxed model

$$\max \Pi = \sum_{j \in I} \sum_{k \in k_p} \sum_{p \in P} r_p z_{jpk} - \sum_{i \in I} \sum_{\substack{j \in I \\ j \neq i}} \sum_{p \in P} \sum_{k \in K_p} c_p x_{ijpk}$$
$$- \sum_{i \in I} \sum_{p \in P} \sum_{k \in K_p} h_p(w_{ipk} - z_{ipk}) \tag{1a}$$

$$\text{s.t.} \quad w_{ipk} = \sum_{j \in I} x_{jipk}, \text{ for all } i \in I, p \in P \text{ and } k \in K_p, \tag{1b}$$

$$z_{ipk} \leq w_{ipk}, \text{ for all } i \in I, p \in P \text{ and } k \in K_p, \tag{1c}$$

$$z_{ipk} \leq d_{ipk}, \text{ for all } i \in I, p \in P \text{ and } k \in K_p, \tag{1d}$$

$$\sum_{j \in I} x_{ijpk} \leq s_{ipk}, \text{ for all } i \in I, p \in P \text{ and } k \in K_p, \tag{1e}$$

$$x_{ijpk} \geq 0, \text{ for all } i, j \in I, p \in P \text{ and } k \in K_p, \tag{1f}$$

$$z_{ipk} \geq 0, \text{ for all } i \in I, p \in P \text{ and } k \in K_p. \tag{1g}$$

The objective function (1a) maximizes the total profit where the first term represents the total revenue obtained from sales, the second term is the total transfer cost, and the last term is the total holding cost. Constraints (1b) define the stock level of each SKU after the transfers are completed. Constraints (1c) and (1d) ensure that sales are less than or equal to demand or the available stock of SKUs after the transfers are made. Constraints (1e) guarantee that a store may not transfer more than its inventory. Constraints (1f) and (1g) define the decision variables.

As mentioned earlier, above problem is simply a profit maximizing transportation problem and can be easily solved by commercial optimizers. Now we extend the problem (1a)–(1g) by adding one of the capacity constraints:

$$\max \text{ (1a)} \tag{2a}$$

$$\text{s.t.} \quad \sum_{\substack{j \in I \\ j \neq i}} \sum_{p \in P} \sum_{k \in K_p} x_{ijpk} \leq A_i, \text{ for all } i \in I, \tag{2b}$$

$$\text{(1b)–(1g)}. \tag{2c}$$

where Constraints (2b) ensure that a store does not transfer more SKUs than it is allowed. This constraint does not pose a challenge as the problem is still a linear program.

Next, we add the second capacity constraint to the current model. To do so, however, we need to introduce a binary decision variable y_{ij} that represents if stores i and j are connected. The extended model is formulated as follows:

$$\max \ (1a) \tag{3a}$$

$$\text{s.t.} \quad \sum_{\substack{j \in I \\ j \neq i}} \sum_{p \in P} \sum_{k \in K_p} x_{ijpk} \leq A_i, \text{ for all } i \in I, \tag{3b}$$

$$\sum_{\substack{j \in I \\ j \neq i}} y_{ij} \leq B_i, \text{ for all } i \in I, \tag{3c}$$

$$\sum_{j \in I} x_{ijpk} \leq s_{ipk} y_{ij}, \text{ for all } i \in I, p \in P \text{ and } k \in K_p, \tag{3d}$$

$$(1b)-(1d), (1f), \text{ and } (1g). \tag{3e}$$

where Constraints (3c) do not allow a particular store to transfer to more than a given number of stores. Constraints (3d) allow transfer between two stores only if they are connected; these constraints essentially replace Constraints (1e).

Finally, single-destination constraint is added to the model. This constraint requires a change in one decision variable set that represents the SKU flow. We now define a binary decision variable x_{ijp} that represents if product p is transferred from store i to store j, or not. Then, $x_{ijpk} = s_{ipk} x_{ijp}$, which allows us to drop the original flow variables from the formulation. The final model is given below.

The final model

$$\max \Pi = \sum_{i \in I} \sum_{k \in k_p} \sum_{p \in P} r_p z_{ipk} - \sum_{i \in I} \sum_{\substack{j \in I \\ j \neq i}} \sum_{p \in P} \sum_{k \in K_p} c_p s_{ipk} x_{ijp}$$
$$- \sum_{i \in I} \sum_{p \in P} \sum_{k \in K_p} h_p(w_{ipk} - z_{ipk}) \tag{4a}$$

$$\text{s.t.} \quad z_{ipk} \leq \sum_{j \in I} s_{jpk} x_{jip}, \text{ for all } i \in I, p \in P \text{ and } k \in K_p, \tag{4b}$$

$$z_{ipk} \leq d_{ipk}, \text{ for all } i \in I, p \in P \text{ and } k \in K_p, \tag{4c}$$

$$w_{ipk} = \sum_{j \in I} x_{jip} s_{jpk}, \text{ for all } i \in I, p \in P \text{ and } k \in K_p, \tag{4d}$$

$$\sum_{j \in J} x_{ijp} = 1, \text{ for all } i \in I, p \in P, \tag{4e}$$

$$\sum_{\substack{j \in I \\ j \neq i}} \sum_{p \in P} \sum_{k \in K_p} s_{ipk} x_{ijp} \leq A_i, \text{ for all } i \in I, \tag{4f}$$

$$\sum_{\substack{j \in I \\ j \neq i}} y_{ij} \leq B_i, \text{ for all } i \in I, \tag{4g}$$

$$x_{ijp} \leq y_{ij}, \text{ for all } i, j \in I, j \neq i \text{ and } p \in P, \tag{4h}$$

$$x_{ijp} \in \{0, 1\}, \text{ for all } i, j \in I \text{ and } p \in P, \tag{4i}$$

$$y_{ij} \in \{0, 1\}, \text{ for all } i, j \in I, \tag{4j}$$

$$z_{ipk} \geq 0, \text{ for all } i \in I, p \in P \text{ and } k \in K_p. \tag{4k}$$

where Constraints (4e) ensure that if a product is transferred from a store, its entire inventory is moved to exactly one store. As a result, assignment to multiple stores is not allowed and similarly, a store may not also keep a portion of the inventory. As we have elaborated before, this "single-destination" practice is rather peculiar, but nonetheless it is the case at LC Waikiki. The company justify this practice on the grounds that without this they would have to devote too much of their sales personnels' times for collection, which they are not willing to do. Clearly, this assumption may have a substantial impact on the profit, as

it may severely restrict options to better match demand with the supply. Indeed, in our numerical experiments we try to give a sense of the implications of this assumption. As we have also noted, this assumption also complicates the problem substantially, without which the problem can be solved much more effectively.

Before we move to the analysis of the problem, we like to point out that the final model is indeed quite difficult. The following proposition shows that the problem is NP-hard.

Proposition 1. *Problem* (4a)–(4k) *is NP-hard.*

Proof. We will prove the proposition by reduction. Assume that there is only one product ($P = \{1\}$), no holding cost, ($h = 0$) and the product has only one size ($K_1 = \{1\}$). Furthermore, assume that the unit net revenue of the product is zero, ($r = 0$), and there is no limitation on the number of stores to which each store can be connected (unlimited B_i). Since $r = 0$, Constraints (4b) and (4c) become redundant. Moreover, if B_i is unlimited, Constraints (4g) become redundant. Consequently, since any y_{ij} can be one, Constraints (4h) are also redundant. Now the problem reduces to:

$$\min \Phi = \sum_{\substack{i,j \in I \\ j \neq i}} c x_{ij} \tag{5a}$$

$$\text{s.t.} \quad \sum_{j \in I} x_{ij} = 1, \text{ for all } i \in I, \tag{5b}$$

$$\sum_{\substack{j \in I \\ j \neq i}} s_i x_{ij} \leq A_i, \text{ for all } i \in I, \tag{5c}$$

$$x_{ij} \in \{0, 1\}, \text{ for all } i, j \in I. \tag{5d}$$

Problem (5a)–(5d) is the well-known generalized assignment problem which belongs to class of NP-hard problems (Savelsbergh, 1997). □

As the proposition shows, our problem (4a)–(4k) is a very difficult mixed integer linear problem. As we will present later, our experiments with a commercial optimizer demonstrated that this problem could not be solved effectively. At LC Waikiki, the number of products that are considered for transfer is about 2000, on average. On the other hand, the number of stores is approximately 475 nationwide. Therefore, the proposed mixed integer linear program can be huge and a heuristic approach seems to be a reasonable way to proceed. The next section describes such a heuristic method.

5. Solution approach

We have developed a Lagrangian Relaxation (LR) based approach to obtain good upper bounds in reasonable time. LR has shown exceptional success in solving large scale combinatorial optimization problems (Fisher, 1981). LR is also used in the context of transshipment and it is shown that it can provide acceptable bounds to the optimal solution (Wong et al., 2005, 2006). A solution of the Lagrangian dual provides an upper bound on the optimal solution of the problem (4a)–(4k). To obtain a lower bound (i.e., a feasible solution), we have developed a two-stage heuristic that consists of a construction heuristic and simulated annealing based metaheuristic to improve the solution. Different heuristic and metaheuristic methods are applied to transshipment problems. For example, Patriarca et al. (2016) and Peres et al. (2017) develop metaheuristics to solve transshipment in inventory-routing problems. The latter applied a variable neighborhood search based algorithm, while the former developed a genetic algorithm. Moreover, local search based methods are utilized in transshipment problems. For instance, Wong et al. (2005, 2006) developed a simulated annealing based metaheuristic to find promising feasible solutions. Therefore, we have also opted for such metaheuristic. In the rest of this section, we describe these methods in detail.

5.1. Obtaining upper bounds

Note that in the formulation, Constraints (4c), (4g), and (4f) are similar to knapsack constraints and Constraints (4e) are basic assignment constraints, all of which are well-known in the literature. On the other hand, Constraints (4b) and (4h) complicate the problem because they connect "z" variables to "x" variables and "x" variables to "y" variables, respectively. Thus, problem (4a)–(4k) can be decomposed in well-known problems by relaxing these complicating constraints.

Let $\overline{\alpha} = \{\alpha_{ipk} \in \mathbb{R}^+ : i \in I, p \in P, k \in K_p\}$ and $\overline{\beta} = \{\beta_{ijp} \in \mathbb{R}^+ : i, j \in I, p \in P\}$ represent vectors of Lagrangian multipliers associated with Constraints (4b) and (4h), respectively. Then the relaxed problem can be written as

$$\max \Pi_{LR}(\overline{\alpha}, \overline{\beta}) = \sum_{i \in I} \sum_{p \in P} \sum_{k \in K_p} r_p z_{ipk} - \sum_{\substack{i \in I \\ j \neq i}} \sum_{j \in I} \sum_{p \in P} \sum_{k \in K_p} c_p s_{ipk} x_{ijp}$$

$$- \sum_{i \in I} \sum_{p \in P} \sum_{k \in K_p} h_p(w_{ipk} - z_{ipk})$$

$$- \sum_{i \in I} \sum_{p \in P} \sum_{k \in K_p} \alpha_{ipk}(z_{ipk} - \sum_{j \in I} s_{jpk} x_{jip})$$

$$- \sum_{i \in I} \sum_{\substack{j \in I \\ j \neq I}} \sum_{p \in P} \beta_{ijp}(x_{ijp} - y_{ij}) \tag{6a}$$

$$\text{s.t.} \quad \text{(4c)–(4g) and (4i)–(4k).} \tag{6b}$$

This problem can be decomposed into three subproblems, which are given as Eqs. (7a)–(9c), (8a)–(8e), and (9a)–(9c) in Box I.

Among these problems, Problem (7a)–(7c) is solvable by inspection. Problem (9a)–(9c) can be decomposed into knapsack problems for each store. Problem (8a)–(8e) seems to be computationally the most challenging of the three since this problem is similar to the generalized assignment problem. However, it is also separable for each store, which allows us to efficiently solve it. The subproblems for each $i \in I$ can be written as

$$\max \Pi_{LR}^{x^i}(\overline{\alpha}, \overline{\beta}) = \sum_{\substack{j \in I \\ j \neq i}} \sum_{p \in P} x_{ijp}((\sum_{k \in K_p} (-c_p + \alpha_{jpk} - h_p)s_{ipk}) - \beta_{ijp})$$

$$+ \sum_{p \in P} \sum_{k \in K_p} (\alpha_{ipk} - h_p)s_{ipk} x_{iip} + \sum_{j \in I} \sum_{p \in P} h_p w_{ipk} \tag{10a}$$

$$\text{s.t.} \quad \sum_{j \in J} x_{ijp} = 1, \text{ for all } p \in P, \tag{10b}$$

$$\sum_{\substack{j \in I \\ j \neq i}} \sum_{p \in P} s_{ipk} x_{ijp} \leq A_i, \tag{10c}$$

$$w_{ipk} = \sum_{j \in I} x_{jip} s_{jpk}, \text{ for all } p \in P \text{ and } k \in K_p, \tag{10d}$$

$$x_{ijp} \in \{0, 1\}, \text{ for all } j \in I \text{ and } p \in P. \tag{10e}$$

Suppose that Lagrangian multipliers α_{ipk} and β_{ijp} are set to some values. Then, let us define $\hat{z} = \{\hat{z}_{ipk} : i \in I, p \in P, k \in K_p\}$, $\hat{x} = \{\hat{x}_{ijp} : i, j \in I, p \in P\}$, and $\hat{y} = \{\hat{y}_{ij} : i, j \in I\}$ as the corresponding optimal solutions to the subproblems (7a)–(7c), (8a)–(8e) and (9a)–(9c), respectively. We can then improve the Lagrangian bounds for a given solution by revising these Lagrangian multipliers. We achieve this by solving the Lagrangian dual while retaining the primal solutions. Interested reader can refer to Litvinchev (2007) for a detailed account of this approach. The Lagrangian dual can be formulated as Eq. (11a)–(11h) in Box II.

The objective function (11a) is the objective function of the dual problem. Since the solutions $\hat{x}$, $\hat{y}$, and $\hat{z}$ are known, Constraints (11b)–(11h) are added to modify the Lagrangian multipliers while retaining the primal solutions. Constraints (11b) ensure that if a product p is sent from store i to any other store j, then the coefficient of $\hat{x}_{ijp}$ in the objective function must be less than the coefficient of $\hat{x}_{iip}$. Similarly, the coefficient of $\hat{x}_{ijp}$ must be less than the coefficient of any other $\hat{x}_{ij^*p}$, which is guaranteed by Constraints (11c). On the other hand, if $\hat{x}_{iip} = 1$, that is, product p remains at its original location, then $\hat{x}_{ijp} = 0$,

which is ensured by Constraints (11d). Constraints (11e) guarantee that multipliers α_{ipk} are not greater than corresponding unit revenues plus holding cost to prevent $\hat{z}_{ipk}$ to be zero. Likewise, Constraints (11f) set upper bounds on β_{ijp}. Finally, multipliers α_{ipk} and β_{ijp} must be non-negative which are ensured by Constraints (11g) and (11h), respectively.

The optimal solution to this problem is a tighter upper bound as compared to the solution obtained from the relaxed problem. Naturally, this solution provides an upper bound to the optimal solution of the original problem as well.

5.2. Obtaining lower bounds

As mentioned earlier, we obtain lower bounds, i.e., feasible solutions, via a construction heuristic followed by an improvement metaheuristic. The construction heuristic consists of two steps, in the first of which we iteratively connect stores until there is no improvement. We start by dividing all store-product combinations into two groups as sender and receiver based on their stock and demand levels without considering the sizes. For each product, if the stock level in a store is more than its demand, the store is classified as sender; otherwise, it is classified as a receiver. Note that, a store can be either in the sender group or in the receiver group for a product (or, in none of the groups in case the stock and demand levels are equal). We then sequentially connect senders to receivers by selecting products randomly. For each store in the sender group, we find a candidate store from the receiver group that creates the highest profit, i.e., revenue less implied costs. A transfer decision is made if Constraints (4f) and (4g) remain feasible. After all products are selected, we update the sender and receiver groups considering the current transfers. That is, a store that was initially in the sender group and sends its entire inventory to another store may be included in the receiver group in the next iteration. Moreover, a store in the receiver group can continue to stay in the same group, if it still has needs. Otherwise, it will not be considered as a sender or a receiver. This procedure is repeated until there is no improvement in the solution. In the second step, we further investigate profitable transfers that were not made in the previous step due to Constraints (4g). Now, we search for beneficial transfers by choosing among the destinations that a store is already connected, so that the constraint remains feasible, while the solution is improved.

At the improvement stage, we have developed a simulated annealing based metaheuristic. The proposed metaheuristic essentially destroys the current feasible solution by removing a transfer and then repairing it by inserting another transfer. It removes transfers according to three rules that are applied randomly. In the first rule, the transfer to be removed is also selected randomly. The other two rules use the "residual demand" information for each store-product pair, i.e., the difference between the demand and the transfer it receives in the current solution. That is, those with the negative residual demand are the ones that receive more than their demand. The second rule randomly chooses a store-product pair among those that have negative residual demands. And finally, for the third rule we first list all store-product pairs that have negative residual demand. Then we select the product that appears the most in the list and then choose from the stores that also appears the most in the list and paired with this product. After selecting a store-product pair, that transfer is removed and another transfer is inserted while maintaining the feasibility of Constraints (4e). The destination store is chosen randomly among the ones that have *positive* residual demand. The purpose of these rules is to enable moving to worse as well as better solutions than the current one.

The algorithm allows non-improving moves to include diversity as in the simulated annealing (SA) approach. It is adopted as follows: If the profit of the new transfer is greater than or equal to the profit of the removed one, the transfer is accepted. Otherwise, we accept it with probability $e^{\frac{-(currentProfit-newProfit)}{temperature}}$, where $currentProfit$ and $newProfit$ denote the profits of the removed transfer and the newly added one,

$$\text{Subproblem 1}: \quad \max \Pi^z_{LR}(\overline{\alpha}) = \sum_{i \in I} \sum_{p \in P} \sum_{k \in K_p} z_{ipk}(r_p - \alpha_{ipk} + h_p) \tag{7a}$$

$$\text{s.t.} \quad z_{ipk} \le d_{ipk}, \text{ for all } i \in I, p \in P \text{ and } k \in K_p, \tag{7b}$$

$$z_{ipk} \ge 0, \text{ for all } i \in I, p \in P \text{ and } k \in K_p. \tag{7c}$$

$$\text{Subproblem 2}: \quad \max \Pi^x_{LR}(\overline{\alpha}, \overline{\beta}) = \sum_{i \in I} \sum_{\substack{j \in I \\ j \ne i}} \sum_{p \in P} x_{ijp} \left(\sum_{k \in K_p} ((-c_p + \alpha_{jpk} - h_p) s_{ipk}) - \beta_{ijp} \right)$$

$$+ \sum_{i \in I} \sum_{p \in P} \sum_{k \in K_p} (\alpha_{iok} - h_p) s_{iok} x_{iip} + \sum_{i \in I} \sum_{j \in I} \sum_{p \in P} h_p w_{ipk} \tag{8a}$$

$$\text{s.t.} \quad \sum_{j \in J} x_{ijp} = 1, \text{ for all } i \in I, p \in P, \tag{8b}$$

$$\sum_{\substack{j \in I \\ j \ne i}} \sum_{p \in P} \sum_{k \in K_p} s_{ipk} x_{ijp} \le A_i, \text{ for all } i \in I, \tag{8c}$$

$$w_{ipk} = \sum_{j \in I} x_{jip} s_{jpk}, \text{ for all } i \in I, p \in P \text{ and } k \in K_p, \tag{8d}$$

$$x_{ijp} \in \{0, 1\}, \text{ for all } i, j \in I \text{ and } p \in P. \tag{8e}$$

$$\text{Subproblem 3}: \quad \max \Pi^y_{LR}(\overline{\beta}) = \sum_{i \in I} \sum_{\substack{j \in I \\ j \ne I}} \sum_{p \in P} \beta_{ijp} y_{ij} \tag{9a}$$

$$\text{s.t.} \quad \sum_{\substack{j \in I \\ j \ne i}} y_{ij} \le B_i, \text{ for all } i \in I, \tag{9b}$$

$$y_{ij} \in \{0, 1\}, \text{ for all } i, j \in I. \tag{9c}$$

Box I.

$$\min_{\overline{\alpha}, \overline{\beta}} \max \Delta(\hat{z}, \hat{x}, \hat{y}) = \sum_{i \in I} \sum_{p \in P} \sum_{k \in K_p} \hat{z}_{ipk}(r_p - \alpha_{ipk} + h_p) + \sum_{i \in I} \sum_{\substack{j \in I \\ j \ne I}} \sum_{p \in P} \beta_{ijp} \hat{y}_{ij} + \sum_{i \in I} \sum_{p \in P} \sum_{k \in K_p} (\alpha_{ipk} - h_p) s_{ipk} \hat{x}_{iip}$$

$$+ \sum_{i \in I} \sum_{\substack{j \in I \\ j \ne i}} \sum_{p \in P} \hat{x}_{ijp} \left(\left(\sum_{k \in K_p} (-c_p + \alpha_{jpk} - h_p) s_{ipk} \right) - \beta_{ijp} \right) \tag{11a}$$

$$\text{s.t.} \quad \sum_{k \in K_p} (\alpha_{jpk} - c_p - h_p) s_{ipk} - \beta_{ijp} \le \sum_{k \in K_p} (\alpha_{ipk} - h_p) s_{ipk}, \text{ for all } i \ne j \in I, p \in P, k \in K_p \text{ if } \hat{x}_{ijp} = 1, i = j, \tag{11b}$$

$$\sum_{k \in K_p} (\alpha_{jpk} - c_p - h_p) s_{ipk} - \beta_{ijp} \le \sum_{k \in K_p} (\alpha_{j^*pk} - c_o - h_p) s_{ipk} - \beta_{ij^*p}, \text{ for all } i, j \in I, p \in P, k \in K_p \text{ if } \hat{x}_{ijp} = 1, i \ne j, j^*, \tag{11c}$$

$$\sum_{k \in K_p} (\alpha_{ipk} - h_p) s_{ipk} \le \sum_{k \in K_p} (\alpha_{j^*pk} - c_p - h_p) s_{ipk} - \beta_{ij^*p}, \text{ for all } i, j \in I, p \in P, k \in K_p \text{ if } \hat{x}_{ijp} = 1, i \ne j \ne j^*, \tag{11d}$$

$$\alpha_{ipk} \le r_p + h_p, \text{ for all } i \in I, p \in P \text{ and } k \in K_p, \tag{11e}$$

$$\beta_{ijp} \le \sum_{k \in K_p} ((-c_p + r_p - h_p) s_{ipk}), \text{ for all } i, j \in I \text{ and } p \in P, \tag{11f}$$

$$\alpha_{ipk} \ge 0, \text{ for all } i \in I, p \in P \text{ and } k \in K_p, \tag{11g}$$

$$\beta_{ijp} \ge 0, \text{ for all } i \in I, j \in I, \text{ and } p \in P. \tag{11h}$$

Box II.

respectively and *temperature* is the current temperature, which is a parameter of SA. Initially, *temperature* is equal to the total profit of current solution so that the *probability* becomes high and the chance of accepting a worse solution is high. The *temperature* is decreased at each iteration using the formula *temperature* = *temperature* × *τ*, where $0 < \tau < 1$ is the *cooling rate*. A *counter* keeps the number of worse solutions accepted. The *cooling rate* is calculated by *cooling rate* = $1/counter$. The best solution is kept and updated whenever a better solution is

found. To avoid being trapped in local optima, the algorithm continues to search from either the best solution or second best solution if there is no improvement in a predefined number of iterations. The algorithm stops if either the total number of iterations reaches to its upper bound or the time limit hast been reached.

6. Computational results

In this section, we report on our computational experiments that consist of two main parts. In the first part, our purpose is to compare the effectiveness of our solution method to that of a commercial solver. Towards this purpose, the generated instances that are first solved by Gurobi 8.1 and then by our algorithm, which is implemented in Python 3.6 and the subproblems are also solved with Gurobi. All experiments are conducted on a High-Performance Computing (HPC) cluster with Linux system, 44 GB RAM, and 2.40 GHz processors with two cores. In the second part, our purpose is to develop insights into the effects of the operational constraints on the system performance and the impact of initial replenishment levels on the benefits of transshipment opportunities. Towards this end, we develop another set of instances, all of which are solved by Gurobi.

LC Waikiki has about 475 stores (excluding the outlet ones) and about 2000 items considered for transfer at any week. Unfortunately, problems of this scale cannot be directly handled by Gurobi. Therefore, we have targeted 50 and 100 as the number of stores and 100, 200, 500, and 1000 as the number of products. The number of sizes varies according to the product. However, most of the products have five to 10 different sizes (e.g., S, M, L, XL, and XXL or 28, 30, 32, 34, 36, 38, 40, 42, 44, and 46 for two different products). Thus, in the test problems, the number of sizes is set to either five or 10. Detailed information on the combination of sizes of the instances are given in the first columns of Tables 2, 3, and 4. In total, we have solved instances of 14 different size combinations.

We randomly generated demands (d_{ipk}) and initial stock levels (s_{ipk}) from a discrete uniform distribution that is defined between 0 and 10. The prices of the products are set between 20 and 50 Turkish Lira (TL) while the transfer costs for these products are set between 0.4 to 1.5 TL. We generated the selling prices and transfer costs randomly from uniform distributions with the bounds given above. To set the holding cost rate we should have also drawn unit costs but in the interest of simplicity we used the unit revenues. The holding cost per week is taken as the 0.5% of the unit revenue, which corresponds to about 30% or less, annually. Although we have not used any real data from LC Waikiki to develop these instances, we have decided on these values upon our conversations with the group that deals with the transfers; hence, we believe our instances are quite realistic.

In order to find the parameters of two of the operational constraints, first we solved each instance by ignoring all three restrictions. We then found the number of items each store sends and the number of stores it is connected. These are essentially, the maximum values when there are no operational constraints. Based on these numbers, we then set three levels of A_i and B_i for each store as low (1/3 of the maximum), medium (1/2 of the maximum), and high (2/3 of the maximum).

For each size and (A_i, B_i) combinations, we have randomly generated 10 instances. We have reached at this number through a small numerical experiment. We took three instance sizes, generated 100 random instances of each, solved them with our algorithm, and computed the average gaps progressively. It turned out that after 10th replication the progressive average of gaps becomes nearly constant for all three sets of instances as illustrated in Fig. 2. Therefore, we concluded that 10 instances would be enough to have a reliable performance metric in terms of average gaps. As a result, we have generated and tested a total of $14 \times 3 \times 10 = 420$ instances.

The results are illustrated in Tables 2, 3, and 4. The first column depicts the size of each instance with respect to the number of stores, products, and sizes. The Gurobi column reports the average of best
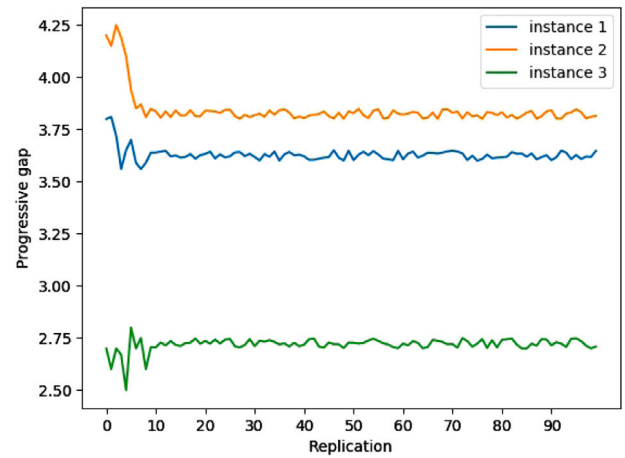

Fig. 2. Sensitivity of progressive optimality gap to the number of replications.

feasible solutions, the minimum, maximum, and average optimality gaps of 10 replications that Gurobi achieved and the time limit that we set. We have given a one-hour time limit for smaller sized problems, two-hour time limit for medium sized ones, and six-hour time limit to the larger sized instances, in addition to the problem loading times to Gurobi. The metaheuristic column also reports the average of best feasible solutions, the minimum, maximum, and average optimality gaps of 10 replications and the time our algorithms spent to find the upper and lower bounds. We calculated the optimality gaps as $\frac{UB - LB}{LB}$ where UB and LB represent the upper and lower bounds found by Gurobi and our method.

The results show that our algorithm is comparable to, and in some cases much more effective than, Gurobi. First of all, our algorithm spends about one-third to one-half of the time that we give to Gurobi excluding the time it takes to load the problem to Gurobi, which could be rather substantial in larger instances. In terms of the optimality gaps, the results are somewhat mixed. There are many instance sets, for which Gurobi found better solutions (lower bounds) than our method did. However, Gurobi's solutions deteriorate faster than our method with increasing problem size. Although our approach also suffers, it can solve most medium-sized problems with around 1% optimality gaps and large-sized problems with a maximum gap of about 7%.

The most important problem with Gurobi, however, is that it is rather unreliable. In some cases the solutions it found were just terrible, with optimality gaps hovering around 300%. Upon close inspection, we noticed that these poor results belong to the instances where there are five different sizes, whereas the instances with the same number of stores and products but 10 different sizes for each products, the behavior was quite the opposite. This was rather puzzling; after all, the latter instances are of larger size, but with closer inspection we were able to conclude that it was the combination of several factors that led to this unexpected results. First of all, since the transportation costs are quite low as compared to the revenues, as long as it is revenue-improving the optimal solution tends to have large number of transfers. Secondly, when there are 10 sizes for each product, there are fewer profitable opportunities for transfers as compared to the same number of stores and products with five sizes for each product. This might seem unclear at first, but single-destination constraint is mainly responsible for these results. For example, if there is only one size, then there would be many profitable opportunities for transfer. When there are two sizes, the opportunities would diminish because there would be more sales opportunities at the original sources and there would be fewer alternative stores that would have demand for both sizes. This would be even more prominent with increasing the number of sizes. This is indeed what we have observed in a simple experiment that we have conducted with 20 stores, 100 products and no operational

Table 2
Results for low level of A_i and B_i.

| Instance | Gurobi | | | | | Metaheuristic | | | | |
| | Average | Gap | | | Average | Average | Gap | | | Average |
I-P-K	Lower bound	Min	Average	Max	Runtime	Lower bound	Min	Average	Max	Runtime
50-100-5	848,013	0.71	0.79	0.89	3600	841,322	1.04	2.07	3.67	1285
50-100-10	307,259	0.00	0.00	0.01	7.04	302,956	0.50	2.41	5.33	1017
50-200-5	1,519,767	0.78	0.87	1.01	3600	1,500,891	1.06	3.00	5.38	1424
50-200-10	935,098	0.79	1.09	1.37	3600	924,970	1.77	3.39	8.32	1318
50-500-5	2,317,320	282.10	288.27	296.74	3600	6,244,111	4.22	4.48	4.76	2001
50-500-10	2,417,042	0.34	0.40	0.45	3600	2,398,677	0.98	2.40	3.65	1960
50-1000-5	4,663,613	262.45	284.38	302.65	3600	11,074,571	2.50	3.76	4.29	1966
50-1000-10	3,507,723	0.09	0.14	0.22	3600	3,487,356	0.76	1.04	1.44	2082
100-100-5	950,596	283.04	287.71	293.21	7200	2,695,145	3.47	5.30	6.20	1492
100-100-10	1,890,957	0.22	0.31	0.41	7200	1,882,898	0.57	1.58	3.29	3707
100-500-5	4,666,991	284.15	287.76	294.74	7200	12,141,278	6.10	6.57	6.96	4313
100-500-10	7,809,377	0.71	0.91	1.11	7200	7,714,316	0.85	3.52	5.16	4414
100-1000-5	9,303,094	284.71	288.34	291.90	21600	18,948,867	0.49	0.94	1.24	9245
100-1000-10	12,997,254	0.39	0.99	1.28	21600	12,906,313	1.33	2.45	3.22	6475

Table 3
Results for medium level of A_i and B_i.

| Instance | Gurobi | | | | | Metaheuristic | | | | |
| | Average | Gap | | | Average | Average | Gap | | | Average |
I-P-K	Lower bound	Min	Average	Max	Runtime	Lower bound	Min	Average	Max	Runtime
50-100-5	1,023,543	0.42	0.49	0.56	3600	1,015,396	0.82	2.57	4.95	1274
50-100-10	547,922	0.01	0.01	0.01	169	542,032	0.63	1.74	3.05	1181
50-200-5	1,794,245	0.72	0.83	0.95	3600	1,760,781	1.60	3.58	5.37	1404
50-200-10	1,249,620	0.47	0.56	0.66	3600	1,224,849	1.09	3.36	6.11	1366
50-500-5	5,085,082	6.65	90.41	287.53	3600	6,501,742	3.70	3.96	4.23	2000
50-500-10	3,192,027	0.08	0.10	0.13	3600	3,149,863	0.73	3.53	4.67	1982
50-1000-5	4,663,613	283.04	287.71	293.21	3600	12,553,494	1.12	1.80	2.93	1974
50-1000-10	6,070,388	0.10	0.14	0.19	3600	5,984,386	1.08	1.75	2.56	2075
100-100-5	3,008,779	0.52	0.82	1.18	7200	2,997,312	2.57	4.49	5.21	1475
100-100-10	1,890,938	0.07	0.08	0.10	7200	1,879,030	0.34	2.79	4.13	3774
100-500-5	4,636,738	284.96	289.79	294.74	7200	12,713,377	4.81	5.30	5.76	4754
100-500-10	9,447,082	0.94	1.06	1.17	7200	9,363,384	2.09	2.74	3.43	4514
100-1000-5	9,325,113	284.71	287.41	289.97	21600	20,854,343	1.83	2.50	2.92	9028
100-1000-10	15,871,661	0.01	0.42	1.22	21600	15,780,238	1.11	1.55	2.52	6109

Table 4
Results for high level of A_i and B_i.

| Instance | Gurobi | | | | | Metaheuristic | | | | |
| | Average | Gap | | | Average | Average | Gap | | | Average |
I-P-K	Lower bound	Min	Average	Max	Runtime	Lower bound	Min	Average	Max	Runtime
50-100-5	1,171,178	0.33	0.39	0.48	15	1,154,752	1.79	2.96	5.17	1290
50-100-10	819,047	0.01	0.02	0.06	2845	805,510	0.37	2.13	3.10	1333
50-200-5	2,042,125	0.58	0.79	1.37	3600	2,017,624	1.52	3.36	5.10	1448
50-200-10	1,507,772	0.31	0.37	0.46	3600	1,496,739	1.31	2.63	3.88	1423
50-500-5	7,151,269	0.12	0.28	0.41	3600	7,120,120	0.72	1.06	1.97	1778
50-500-10	4,456,310	0.01	0.01	0.01	154	4,296,653	3.37	4.28	4.77	2131
50-1000-5	12,891,943	0.07	0.12	0.16	3600	12,839,782	0.46	0.75	1.13	1971
50-1000-10	8,840,304	0.01	0.01	0.02	1963	8,462,620	1.46	2.12	2.98	2064
100-100-5	3,021,758	0.20	0.30	0.46	7200	3,010,490	3.76	4.55	5.64	1464
100-100-10	1,891,297	0.53	0.56	0.63	7200	1,872,998	4.41	5.23	5.96	3647
100-500-5	14,830,650	0.49	1.27	2.16	7200	14,685,243	2.66	3.87	5.58	4201
100-500-10	9,492,130	0.02	0.14	0.29	7200	9,373,087	1.10	2.19	3.02	4512
100-1000-5	29,909,041	0.56	0.82	1.56	21600	29,313,762	1.55	3.39	5.72	9014
100-1000-10	19,301,509	0.00	0.01	0.01	11479	19,217,184	0.65	1.09	1.43	6004

constraints. We then set the number of sizes as 2, 5, 8, 10, and 15 and have randomly drawn 10 instances for each size. Table 5 reports the results, which confirm our intuition. Hence, since there are far fewer transfers in the optimal solution as the number of sizes increases most stores do not perform any transfers but keep at the source. As a result, Gurobi can eliminate a substantial number of potential transfers across the stores and finds solutions easier in instances where there are 10 sizes as compared to five. To conclude, while our approach does not produce better solutions than Gurobi all the time, due to its robustness to problem characteristics, it is a much better alternative of the two.

In the second part of our experiments, our purpose is to shed some light into the effects of particular operational restrictions used by LC

Waikiki. The restrictions include the two capacity constraints and the single-destination policy. Towards this end, we considered six problem sizes as illustrated in Fig. 3. We randomly generated 10 instances for each of these problem sizes. As we have done in the first part, we have created further instances based on how tight the capacity constraints are. Similarly, we first solved each instance by ignoring all three restrictions, found the maximum values A_i and B_i can take for each store, and then created four combinations of (A_i, B_i) by setting them to either "low" (1/3 of the maximum) or "high" (2/3 of the maximum). Therefore, altogether we have solved a total of 240 instances.

The unconstrained version of the problem, i.e., (1a)–(1g), is solved first, and then the problem with the transfer capacity constraint,

Table 5
Effect of the number of sizes on the number of transfers.

K	Instance 1	Instance 2	Instance 3	Instance 4	Instance 5	Instance 6	Instance 7	Instance 8	Instance 9	Instance 10
2	1038	1074	1009	1013	1031	1030	1069	1011	998	1038
5	616	626	624	650	630	625	634	653	612	621
8	327	371	334	366	352	353	315	335	315	354
10	208	226	220	220	206	231	204	244	207	237
15	42	55	65	57	48	53	59	57	78	50

i.e., (2a)–(2c), which is followed by the problem with both capacity constraints, i.e., (3a)–(3e), and finally, the full problem (4a)–(4k) is solved. All these problems are solved with Gurobi. Although not all problems are solved to optimality, the gaps are rather small, so the results are quite reliable.

Fig. 3 depicts the summary results. Each graph in the figure contains results based on the combinations of (A_i, B_i) pairs. In the figure, optimum solution of each unconstrained instance is normalized to 100 and the objective functions of each instance's constrained versions are found as percentage of the optimal value of the unconstrained version. The graphs report the averages of these percentages over 10 instances. The effect of the constraint on the number of SKUs that can be transferred is quite clear. The optimal values for low A_i instances reduce to roughly 50%–70% of the maximum possible under unconstrained values (top two graphs). On the other hand, in most of the instances with high A_i's the optimal values do not decrease significantly. Although in some instances they may drop to less than 85% of the maximum possible, in most instances they drop to around mid-90% of the maximum possible. The addition of the second capacity constraint that restricts the number of stores deteriorates optimal values particularly when A_i's are high, but not when they are low. When A_i's are already low, further drop on the optimal value is around 5% and most drop happens in the instances with 10 sizes for each product. This result is intuitive because when there are 10 sizes, there are simply more opportunities to match the demand and supply as there are more sizes (because there is no single-destination constraint yet). Therefore, restricting the number of stores eliminates more of those opportunities. When the A_i's are high, however, the impact of the addition of the second capacity constraint has a much more detrimental effect across all instances, but particularly more so again in the 10-size instances due to the same reason. In the end, however, the constraint that restricts the number of items is more "constraining" than the constraint that restricts the number of stores. As one can see from the graphs, while in low A_i and high B_i cases (top-right graph) the optimal value drops to 40%–65% of the maximum, in high A_i and low B_i cases (bottom-left graph) the optimal value drops to around 65%–85% of the maximum possible.

We can observe that the single-destination constraint, after the capacity constraints, has a modest deteriorating impact on the optimal value. Depending on the cases, it has roughly an additional 5%–15% negative impact on the maximum possible values. This impact gets somewhat stronger when there are fewer opportunities for transfers, that is when there are fewer stores and products. For example, the largest impact of this constraint can be seen for the instances of 20 stores, 100 products, and 10 sizes (i.e., 20–100–10). This result is also quite intuitive because when there are more stores and/or more products, there are potentially larger number of attractive single-destination transfer options and therefore, the impact of this constraint is lessened. However, when the alternatives are already scarce, the single-destination constraint leads to yet fewer transshipment moves, deteriorating the objective function further.

In the last part of our experiments, we investigate the effect of initial replenishment levels on transshipment benefits under the particular operational constraints and practices. We must remark that we do not attempt to solve a joint replenishment-transshipment problem; such would be a daunting undertaking for the environment we are considering. Not only the problem sizes are enormous with hundreds of stores

and thousands of SKUs, but also the particular operational characteristics and practices and the need to include demand uncertainty make the task almost impossible. Therefore, what we try to demonstrate here is how our model can be used to quantify the transshipment benefits under a few replenishment levels. Towards this end we conducted a limited set of experiments with 10 instances of the problem with 20 stores, 100 products, and five sizes. Although other instances show a similar pattern, since these results ultimately depend on the particular way the problem parameters are generated, our results should be considered illustrative rather than suggestive.

To find the transshipment benefits, we first calculated the objective function value for each instance without any transshipment; that is, each store can only sell what it has and incur inventory holding cost for the remaining SKUs. We then solved six transshipment problems for each instance. The first one is the problem (1a)–(1g) where there is no constraint on transshipments, which gives the maximum possible benefit of the transshipment option. We then solved the main problem (4a)–(4k) with five capacity settings; four of which are the high-low capacity combinations as in the previous experiment and the fifth one is the unlimited capacity case, which is included to investigate the impact of the single-destination constraint alone. We then found the ratio of the objective function values under each transshipment scenario to that of the no transshipment case and then took the average of the ratios over 10 instances, which we use as a measure of the transshipment benefits under different conditions. In other words, we measure the "relative" benefit of the transshipment cases with respect to the no transshipment option.

The way we include different replenishment levels into the analysis is through modification of the current SKU inventories by some amount. For example, for a product where the current inventories of the five sizes are (2,3,1,0,2), a reduced replenishment level can be approximated with the modified inventories of (1,2,0,0,1), i.e., each SKU inventory is reduced by one. Similarly, an increased replenishment level can be approximated with (3,4,2,1,3), i.e., each is increased by one. This type of modification is a rough approximation of a decision to decrease or increase the initial replenishment order by one for each SKU and for each store. Such a setting is admittedly rough; firstly, because it involves some assumptions on what happens to the SKUs with zero inventories and secondly, it assumes that the company treats all SKUs and stores the same. However, dealing with such complexities require many other assumptions and instances with different settings. Therefore, we set aside those complexities and focus only on these rough approximations of replenishment level decisions.

Fig. 4 depicts our results, where "supply" case refers to the instances with the original inventory values, while the others are the instances in which the inventories are reduced or increased by one or two. Firstly, we can observe that as the constraints on transshipments are more relaxed, relative transshipment benefits increase, which is expected. For example, in the original instances, while the average ratio could be higher than 3.00 for the unconstrained transshipment case, the same could be as low as 1.8 for the instances with low A_i's, and somewhere in between for the other cases. Secondly, as the inventories decrease, relative transshipment benefits increase slightly or stay roughly the same. This observation suggests that even if the company tends to reduce the initial replenishment quantities, the transshipment benefits continue to be substantial. On the other hand, when the inventories are increased there is a sharp decrease in the relative transshipment
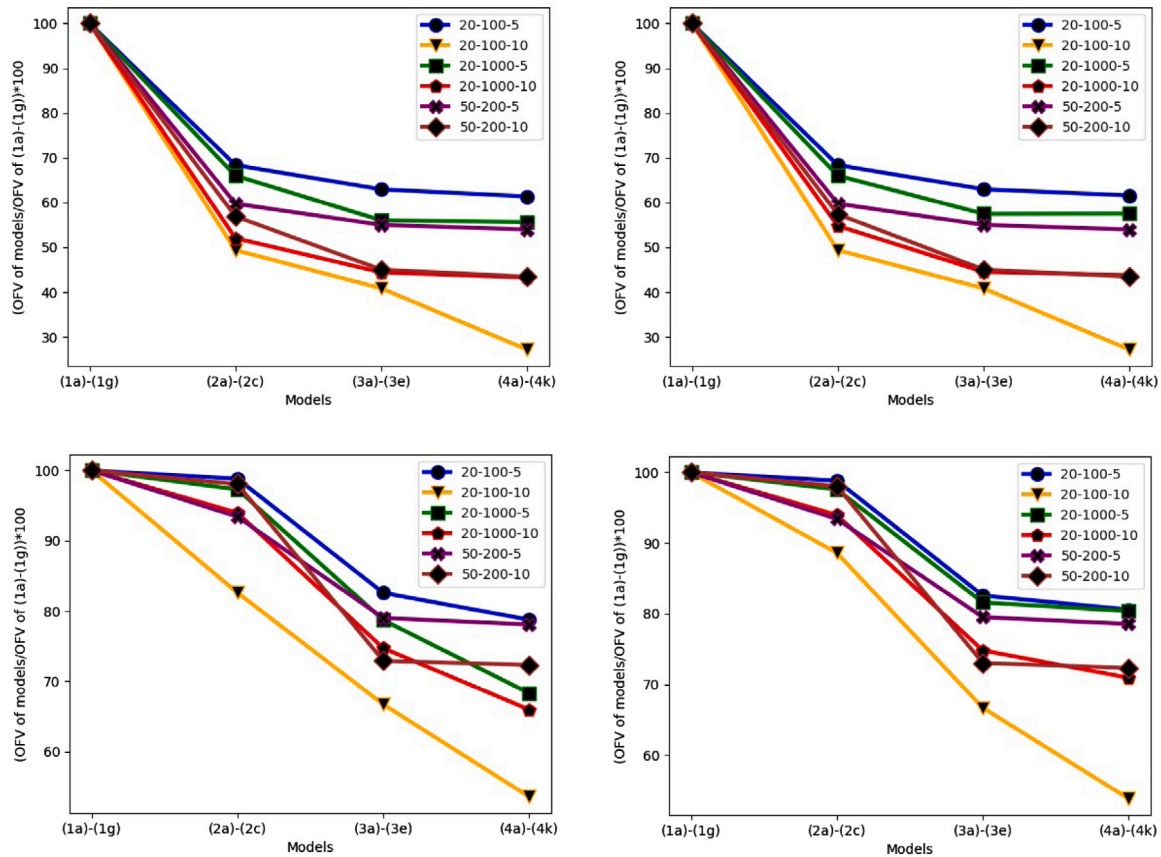
Fig. 3. Effect of adding capacity and single-destination constraints on the objective function value (OFV). Top-left: low A_i and low B_i, top-right: low A_i and high B_i, bottom-left: high A_i and low B_i, and bottom-right: high A_i and high B_i.

benefits in most cases. This result is also expected because as the inventories at the branches are increased, there are much fewer stores and SKUs that would need extra items. The only exception to this result is the cases with low A_i's when there is only modest increase in the inventories. In these cases a slight increase in the inventories may actually improve the relative transshipment benefits or deteriorate them less. The reason for this result is that when the inventories get slightly larger, the model may find more attractive cases under tight capacity constraints. However, as the inventories increase further (e.g., the "supply+2" case), relative benefits of transshipment decrease for these cases as well. Eventually, the differences between all the cases tend to reduce as the inventories increase. This result is also intuitive because when inventories are increased substantially, fewer stores and SKUs would have additional needs, which in turn reduces the needs for transshipments in the system and therefore, the negative effects of operational constraints on transshipment lessen as well.

To summarize the managerial implications of the first part of the experiments, we can conclude that in general, (i) it is the restriction on the total number of SKUs that can be transferred rather than the restrictions on the number of destinations that has a more negative effect, (ii) single-destination constraint has a more detrimental effect when the capacities are less restrictive, and finally, (iii) all the negative effects are usually more pronounced when there are larger number of sizes per product. Therefore, if a firm wishes to relax the single-destination constraint, it should start from products with larger number of sizes and accompany it relaxing restrictions on the transfer capacities. From the second part of the experiments, we can conclude that while reduced replenishment levels almost never lessen the relative benefits of transshipments, increased replenishment levels do lessen the relative benefits particularly when the constraints on transshipments are more relaxed. As a side result, we can also conclude that if the
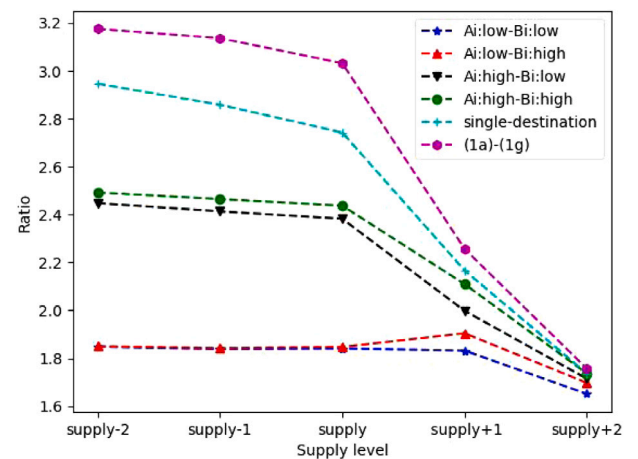

Fig. 4. Effect of initial replenishment level on transshipment benefit.

company considers relaxing some of the constraints on transshipments (for example, single-destination) at least for some products, it should consider lower inventory items first.

7. Concluding remarks

In this paper we introduce a novel proactive transshipment problem motivated by the practice at the largest fast fashion retailer in Turkey, LC Waikiki. The company, after allocating the initial inventory to over 475 stores and observing sales for a few weeks, engages in lateral transshipments among the stores. When a product has different sales

performances across the stores, lateral transshipments can improve the overall system performance. Not only such a practice helps the company better match supply with demand but also eliminates additional handling and transportation operations at its central depot.

Its large scale and particular operational restrictions necessitate the development of a novel model. We formulate the transfer problem of LC Waikiki as a mixed integer linear programming problem. With around 475 stores, 2000 products, and a variety of operational constraints, this problem becomes a very large mixed integer program and solving it optimally becomes a challenge. Therefore, we have developed a simulated annealing based metaheuristic to solve the problem. We also applied Lagrangian relaxation with a primal–dual approach to obtain sharp upper bounds on the optimal solution of the original problem. We generated 420 problem instances of varying sizes to evaluate the performance of the proposed algorithm against the commercial optimizer Gurobi. Each instance is solved by the proposed algorithm and Gurobi. The results show that although the solutions prescribed by Gurobi are better than ours in small-size instances and those with loose capacities, the proposed algorithm outperforms Gurobi in instances that are characterized by having a large number of potentially beneficial transfers and tighter capacity constraints. These instances are the ones where the combinatorial nature of the problem becomes the most challenging. Gurobi fails spectacularly in these instances, while our algorithm performs without a significant loss in its performance. Hence, our algorithm is quite robust to changing program characteristics. Finally, our algorithm is also quicker in finding solutions, spending only about one-third to one-half of the time spent by Gurobi. This feature makes our approach particularly attractive when companies need speedy solutions to these problems.

We have also conducted a carefully designed numerical experiment to uncover the effect of the particular operational constraints of the company. First, we have solved the instances without any of those constraints and found the maximum potential gross revenue (i.e., the base gross revenue) that can be obtained with transshipments. We have then added those constraints one by one to observe their effects. We observe that constraints on the total number of products a store can send may have a significant impact and depending on how tight those restrictions are, it may reduce the gross revenue to as low as 50% of its base level. The second capacity restriction, i.e., the number stores that a store can make shipments, usually has very little negative impact if the first constraint is already tight, but its impacts increase otherwise. Depending on the cases, it can result in an additional 5–20% decrease of the base revenue We have also measured the effect of the single-destination practice and have found that it may reduce the revenues by another 5–15% of the base revenue depending on how tight the capacity constraints are. When the capacity constraints are already tight, the negative impact of this practice is quite small, but where the capacities are loose the negative impact of this practice is quite significant. We have also observed that the number of sizes also plays a significant role in these results. In our sample instances we have used five and 10 as the number of sizes. Naturally, when there are 10 sizes of products, single-destination practice renders much fewer number of transfers as potentially beneficial and therefore, this practice becomes more detrimental to the base revenue when the number of sizes increases. Finally, we have investigated the effect of initial replenishment level on relative transshipment benefits. The results show that, while the reduced replenishment levels usually does not reduce the relative benefits of transshipments, increased replenishment levels may do so, particularly if the constraints in the transshipments are more relaxed.

We had to make a number of simplifying assumptions to effectively deal with this very large problem that has complicating operational constraints. Therefore, there are a number of avenues for future research. While one may have accurate demand forecasts as in this case, there are always forecast errors, and therefore, considering demand uncertainty is naturally an important extension to this study. In a similar vein, initial shipment decisions under demand uncertainty may also be considered jointly with the transfer decisions. Another potentially important avenue is to develop integrated models that include transfer decisions as well as markdown decisions, an avenue that we are currently pursuing. Finally, the frequency at which the collections are renewed can also be made jointly with transfer as well as other logistical decisions.

References

Abouee-Mehrizi, H., Berman, O., Sharma, S., 2015. Optimal joint replenishment and transshipment policies in a multi-period inventory system with lost sales. Oper. Res. 63, 342–350.

Acimovic, J., Graves, S.C., 2014. Making better fulfillment decisions on the fly in an online retail environment. Manuf. Serv. Oper. Manage. 17, 34–51.

Agrawal, V., Chao, X., Seshadri, S., 2004. Dynamic balancing of inventory in supply chains. European J. Oper. Res. 159, 296–317.

Ahmadi, G., Torabi, S.A., Tavakkoli-Moghaddam, R., 2016. A bi-objective location-inventory model with capacitated transportation and lateral transshipments. Int. J. Prod. Res. 54, 2035–2056.

Allen, S.C., 1958. Redistribution of total stock over several user locations. Nav. Res. Logist. 5, 337–345.

Archibald, T.W., Black, D., Glazebrook, K.D., 2009. An index heuristic for transshipment decisions in multi-location inventory systems based on a pairwise decomposition. European J. Oper. Res. 192, 69–78.

Archibald, T.W., Black, D.P., Glazebrook, K.D., 2010. The use of simple calibrations of individual locations in making transshipment decisions in a multi-location inventory network. J. Oper. Res. Soc. 61, 294–305.

Archibald, T.W., Sassen, S., Thomas, L., 1997. An optimal policy for a two depot inventory problem with stock transfer. Manage. Sci. 43, 173–183.

Axsäter, S., 1990. Modelling emergency lateral transshipments in inventory systems. Manage. Sci. 36, 1329–1338.

Banerjee, A., Burton, J., Banerjee, S., 2003. A simulation study of lateral shipments in single supplier, multiple buyers supply chain networks. Int. J. Prod. Econ. 81, 103–114.

Bertrand, L.P., Bookbinder, J.H., 1998. Stock redistribution in two-echelon logistics systems. J. Oper. Res. Soc. 49, 966–975.

Bhatnagar, R., Lin, B., 2019. The joint transshipment and production control policies for multi-location production/inventory systems. European J. Oper. Res. 275, 957–970.

Boucherie, R.J., van Houtum, G.-J., Timmer, J., van Ommeren, J.-K., 2018. A two-echelon spare parts network with lateral and emergency shipments: A product-form approximation. Probab. Engrg. Inform. Sci. 32, 536–555.

Burton, J., Banerjee, A., 2005. Cost-parametric analysis of lateral transshipment policies in two-echelon supply chains. Int. J. Prod. Econ. 93, 169–178.

Caro, F., Gallien, J., 2007. Dynamic assortment with demand learning for seasonal consumer goods. Manage. Sci. 53, 276–292.

Caro, F., Gallien, J., 2010. Inventory management of a fast-fashion retail network. Oper. Res. 58, 257–273.

Çömez, N., Stecke, K.E., Çakanyıldırım, M., 2012. In-season transshipments among competitive retailers. Manuf. Serv. Oper. Manage. 14, 290–300.

Coelho, L.C., Cordeau, J.-F., Laporte, G., 2012. The inventory-routing problem with transshipment. Comput. Oper. Res. 39, 2537–2548.

Dan, B., He, Q., Zheng, K., Liu, R., 2016. Ordering and pricing model of retailers' preventive transshipment dominated by manufacturer with conditional return. Comput. Ind. Eng. 100, 24–33.

Das, C., 1975. Supply and redistribution rules for two-location inventory systems: One-period analysis. Manage. Sci. 21, 765–776.

Dehghani, M., Abbasi, B., 2018. An age-based lateral-transshipment policy for perishable items. Int. J. Prod. Econ. 198, 93–103.

Dijkstra, A.S., Van der Heide, G., Roodbergen, K.J., 2017. Transshipments of cross-channel returned products. Int. J. Prod. Econ. 1–8.

Diks, E.B., De Kok, A., 1996. Controlling a divergent 2-echelon network with transshipments using the consistent appropriate share rationing policy. Int. J. Prod. Econ. 45, 369–379.

Feng, P., Fung, R.Y., Wu, F., 2017. Preventive transshipment decisions in a multi-location inventory system with dynamic approach. Comput. Ind. Eng. 104, 1–8.

Fisher, M.L., 1981. The Lagrangian relaxation method for solving integer programming problems. Manage. Sci. 27, 1–18.

Gallien, J., Mersereau, A.J., Garro, A., Mora, A.D., Vidal, M.N., 2015. Initial shipment decisions for new products at Zara. Oper. Res. 63, 269–286.

Mirzapour Al-e hashem, S., Rekik, Y., 2014. Multi-product multi-period inventory routing problem with a transshipment option: A green approach. Int. J. Prod. Econ. 157, 80–88.

Herer, Y., Rashit, A., 1995. Lateral stock transshipments in a two location inventory system. In: Proceedings of the 13th International Conference on Production Research, pp. 593–595.

Herer, Y.T., Rashit, A., 1999. Lateral stock transshipments in a two-location inventory system with fixed and joint replenishment costs. Nav. Res. Logist. 46, 525–547.

Herer, Y.T., Tzur, M., 2001. The dynamic transshipment problem. Nav. Res. Logist. 48, 386–408.

Herer, Y.T., Tzur, M., 2003. Optimal and heuristic algorithms for the multi-location dynamic transshipment problem with fixed transshipment costs. IIE Trans. 35, 419–432.

Herer, Y.T., Tzur, M., Yücesan, E., 2006. The multilocation transshipment problem. IIE Trans. 38, 185–200.

Hoadley, B., Heyman, D., 1977. A two-echelon inventory model with purchases, dispositions, shipments, returns and transshipments. Nav. Res. Logist. Q. 24, 1–19.

Hu, Z.-H., Yu, X.-K., 2014. Optimization of fast-fashion apparel transshipment among retailers. Text. Res. J. 84, 2127–2139.

Johansson, L., Olsson, F., 2018. Age-based inventory control in a multi-echelon system with emergency replenishments. European J. Oper. Res. 265, 951–961.

Karmarkar, U.S., Patel, N.R., 1977. The one-period, N-location distribution problem. Nav. Res. Logist. 24, 559–575.

Kiesmüller, G., Minner, S., 2009. Inventory Redistribution for Fashion Products under Demand Parameter Update. Technical Report Beta Report 275. Eindhoven, The Netherlands: Eindhoven University of Technology.

Krishnan, K., Rao, V., 1965. Inventory control in N warehouses. J. Ind. Eng. 16, 212.

Lee, H.L., 1987. A multi-echelon inventory model for repairable items with emergency lateral transshipments. Manage. Sci. 33, 1302–1316.

Lee, J.H., 2015. Modeling the two-region, two-product capacitated production and transportation problem. Int. J. Econ. Finance Manage. Sci. 3, 460–464.

Lee, Y.H., Jung, J.W., Jeon, Y.S., 2007. An effective lateral transshipment policy to improve service level in the supply chain. Int. J. Prod. Econ. 106, 115–126.

Li, X., Sun, L., Gao, J., 2013. Coordinating preventive lateral transshipment between two locations. Comput. Ind. Eng. 66, 933–943.

Liao, Y., Shen, W., Hu, X., Yang, S., 2014. Optimal responses to stockouts: Lateral transshipment versus emergency order policies. Omega 49, 79–92.

Lim, A., Miao, Z., Rodrigues, B., Xu, Z., 2005. Transshipment through crossdocks with inventory and time windows. Nav. Res. Logist. 52, 724–733.

Litvinchev, I.S., 2007. Refinement of lagrangian bounds in optimization problems. Comput. Math. Math. Phys. 47, 1101–1107.

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

Minner, S., Silver, E.A., 2005. Evaluation of two simple extreme transshipment strategies. Int. J. Prod. Econ. 93, 1–11.

Nakandala, D., Lau, H., Shum, P.K., 2017. A lateral transshipment model for perishable inventory management. Int. J. Prod. Res. 55, 5341–5354.

Nonås, L.M., Jörnsten, K., 2007. Optimal solutions in the multi-location inventory system with transshipments. J. Math. Model. Algorithms 6, 47–75.

Olsson, F., 2015. Emergency lateral transshipments in a two-location inventory system with positive transshipment leadtimes. European J. Oper. Res. 242, 424–433.

Özdemir, D., Yücesan, E., Herer, Y.T., 2013. Multi-location transshipment problem with capacitated production. European J. Oper. Res. 226, 425–435.

Paterson, C., Kiesmüller, G., Teunter, R., Glazebrook, K., 2011. Inventory models with lateral transshipments: A review. European J. Oper. Res. 210, 125–136.

Patriarca, R., Costantino, F., Di Gravio, G., 2016. Inventory model for a multi-echelon system with unidirectional lateral transshipment. Expert Syst. Appl. 65, 372–382.

Peres, I.T., Repolho, H.M., Martinelli, R., Monteiro, N.J., 2017. Optimization in inventory-routing problem with planned transshipment: A case study in the retail industry. Int. J. Prod. Econ. 193, 748–756.

Qi, X., 2006. A logistics scheduling model: scheduling and transshipment for two processing centers. IIE Trans. 38, 537–546.

Rahmouni, M., Hennet, J.-C., Fnaiech, F., 2015. Multi-product joint delivery planning in a retailers' network under resource constraints. Int. J. Syst. Sci.: Oper. Logist. 2, 125–134.

Robinson, L.W., 1990. Optimal and approximate policies in multiperiod, multilocation inventory models with transshipments. Oper. Res. 38, 278–295.

Savelsbergh, M., 1997. A branch-and-price algorithm for the generalized assignment problem. Oper. Res. 45, 831–841.

Seidscher, A., Minner, S., 2013. A semi-markov decision problem for proactive and reactive transshipments between multiple warehouses. European J. Oper. Res. 230, 42–52.

Shao, X.-F., 2018. Production disruption, compensation, and transshipment policies. Omega 74, 37–49.

Shao, J., Krishnan, H., McCormick, S.T., 2011. Incentives for transshipment in a supply chain with decentralized retailers. Manuf. Serv. Oper. Manage. 13, 361–372.

Tagaras, G., 1989. Effects of pooling on the optimization and service levels of two-location inventory systems. IIE Trans. 21, 250–257.

Tagaras, G., Vlachos, D., 2002. Effectiveness of stock transshipment under various demand distributions and nonnegligible transshipment times. Prod. Oper. Manage. 11, 183–198.

Tiacci, L., Saetta, S., 2011. A heuristic for balancing the inventory level of different locations through lateral shipments. Int. J. Prod. Econ. 131, 87–95.

van Wijk, A., Adan, I.J., van Houtum, G.-J., 2012. Approximate evaluation of multi-location inventory models with lateral transshipments and hold back levels. European J. Oper. Res. 218, 624–635.

van Wijk, A., Adan, I.J., van Houtum, G., 2019. Optimal lateral transshipment policies for a two location inventory problem with multiple demand classes. European J. Oper. Res. 272, 481–495.

Wong, H., van Houtum, G.-J., Cattrysse, D., Van Oudheusden, D., 2005. Simple, efficient heuristics for multi-item multi-location spare parts systems with lateral transshipments and waiting time constraints. J. Oper. Res. Soc. 56, 1419–1430.

Wong, H., van Houtum, G.-J., Cattrysse, D., Van Oudheusden, D., 2006. Multi-item spare parts systems with lateral transshipments and waiting time constraints. European J. Oper. Res. 171, 1071–1093.