The following publication Zhen, L., Wu, Y., Wang, S., Hu, Y., & Yi, W. (2018). Capacitated closed-loop supply chain network design under uncertainty. Advanced Engineering Informatics, 38, 306-315 is available at https://doi.org/10.1016/j.aei.2018.07.007.

Capacitated closed-loop supply chain network design under uncertainty

Abstract: This study optimizes the design of a closed-loop supply chain network, which contains forward and reverse directions and is subject to uncertainty in demands for new & returned products. To address uncertainty in decision-making, we formulate a two-stage stochastic mixed-integer non-linear programming model to determine the distribution center locations and their corresponding capacity, and new & returned product flows in the supply chain network to minimize total design and expected operating costs. We convert our model to a conic quadratic programming model given the complexity of our problem. Then, the conic model is added with certain valid inequalities, such as polymatroid inequalities, and extended with respect to its cover cuts so as to improve computational efficiency. Furthermore, a tabu search algorithm is developed for large-scale problem instances. We also study the impact of inventory weight, transportation weight, and marginal value of time of returned products by the sensitivity analysis. Several computational experiments are conducted to validate the effectiveness of the proposed model and valid inequalities.

Keywords: capacitated closed-loop supply chain, conic quadratic programming, stochastic programming, valid inequalities, tabu search.

1. Introduction

Customers' required level of service has increasingly escalated with the improvement in a living standard. In terms of delivery time, the demand for new products is more and more urgent. Companies provide remanufactured products, which are in a looking-new condition and favorable quality to satisfy consumer demand. Many well-known enterprises, such as HP, Xerox, and Kodak, design and operate their supply chains by jointly considering forward and reverse supply chains. These enterprises incorporate their remanufacturing processes into their regular production lines and operations (Taleizadeh et al., 2018). The closed-loop supply chain (CLSC) has drawn considerable attention from both the academia and the practitioners.

The traditional supply chain management refers to decisions on efficient production and product transportation from suppliers to demand places through one or more distribution centers (Üster and Hwang, 2016). By contrast, a CLSC network consists of all necessary components to design, fabricate, sell, and recycle a product (Zhang et al., 2015). Thus, the CLSC network design should simultaneously consider operations of the forward and the reverse flows. The design of the CLSC network consists of several long-term and short-term decisions; the former determines whether a distribution center should be built and its corresponding capacity, whereas the latter determines the order assignment strategy (Haddadsisakht and Ryan, 2018). The CLSC design and operations management have attracted attention from the academia and industry over the last 20 years (Üster and Hwang, 2016). However, the increasing popularity of online shopping has improved the CLSC management to a highly profound level given the rapid development of internet technology. The retail platform of Alibaba Group in China announced that USD 25.3 billion of gross merchandise volume was settled through Alipay on November 11, 2017; this value reflects an increase of 39% compared to 2016. However, its return rate is 62.9% according to unofficial statistics. Thus, a well-designed CLSC network is increasingly significant for online shopping companies, such as Alibaba and Amazon.

This study is motivated by a real-world bottleneck problem encountered during online shopping. Now, with the advance of the internet technology, online shopping has become one of the most active ways for consumers to buy remanufactured products (Xu et al., 2017). The supply chain design and operations management is experiencing increasing competitive and regulatory pressures which also lead to new challenges. Hence, supply chain managers are concerned with a novel CLSC network which considers many realistic factors together simultaneously in order to obtain numerous financial benefits. This study investigates a three-tiered supply network that considers capacitated distribution centers, uncertainty demands of new & returned products, risk-pooling strategy to buffer random demands, savings from collocating of a joint distribution center, value loss related to inventory and transportation time, and a linear relationship between distribution center capacity and cost simultaneously. We formulate our problem as a two-stage stochastic programming model in which the first stage is responsible for long-term decisions, such as distribution center locations and their corresponding capacities. The second stage corresponds to certain short-term decisions, such as the product transportation assignment optimization under various scenarios. We then convert our original model to a conic quadratic mixed-integer programming (MIP) model and add certain valid inequalities, such as polymatroid inequalities considering the complexity of our problem. Then, we add extended cover cuts to strengthen the model formulation. Moreover, a tabu search algorithm is also suggested to solve the model with large-scale problem instances.

The remainder of this study is organized as follows. Section 2 introduces an overview of related works. Section 3 proposes a non-linear MIP model about the CLSC design and converts it to a conic quadratic MIP model. In addition, certain valid inequalities are developed to improve the computational efficiency. Section 4 details the parts of the tabu search algorithm, which we use for large-scale problem instances. Section 5 discusses the computational experiments under our optimization strategy. Section 6 presents the conclusions drawn from this study.

2. Related literature

Numerous studies have been conducted on the CLSC design and optimization given the increasing consumer demand for environmentally friendly products. We review three streams of related literature on supply chain management problems. Readers who are interested in this area can refer to Fleischmann et al. (2001), Savaskan et al. (2004), Listeş (2007), Guide and Wassenhove (2009), Zhang et al. (2015), and Wu et al. (2018) to obtain a comprehensive overview on supply chain network design and operations management.

The first research stream is related to capacity restrictions in the facility location problems. Facility location decisions significantly influence the strategic design of supply chain networks (Melo et al., 2009). Quantitative papers about the facility location problem have been published and the facility location problem has been a well-studied topic within operations research (Org, 2007; Ljubić and Moreno, 2018). Facility location problems with minimizing marginal revenue have been studied from various perspectives (Drenzner et al., 2015). In addition, capacity restrictions in the facility location problems are an extension of the original problem and play a critical role (Zhang et al., 2015). Fischetti et al. (2016) use Benders decomposition without separability to conduct computational experiments for solving capacitated facility location problems. Mota et al. (2018) present a multi-objective MIP model that integrates several decisions simultaneously, such as capacitated facility locations, supply chain network design, and technology allocation. From a number of related works, an increasing stream of research is aimed at integrating strategic and operational-level decisions in facility location problems. Furthermore, the role of facility locations in supply chain network management is becoming increasingly crucial in a realistic business environment; scholars are also required to develop further comprehensive decision models, which can jointly capture many realistic factors in the supply chain network management (Melo et al., 2009).

The second topic of related works is concerned with the reverse logistics management issues for remanufactured products. Recently, supply chains with returned products are receiving increased attention in the operations management discipline. Reverse logistics is responsible for taking back returned products and recovering them efficiently and economically (Senthil et al., 2018). The recovery had a significant economic impact on the industry and society, thereby causing an increase in the type of literature in reverse logistics management issues for remanufactured products (Savaskan et al., 2004). Reverse logistics differs from traditional logistics because the former has unique characteristics, such as coordination requirement of two markets (Srivastava, 2007). Fleischmann et al. (1997) provide a first review of quantitative models for reverse logistics. Numerous studies related to reverse logistics use game theory to build remanufacturing models. Li et al. (2017) present a Stackelberg game model for considering forward and reverse supply chains as an integrated problem. Govindan et al. (2015) propose that we should focus on multi-objective problems by using new approaches to realize green, sustainable, and environmental objectives, such as pollution prevention and life-cycle assessment.

The last stream of studies explores the CLSC design and optimization problem. In the modern structure of the supply chain network management, forward and reverse logistics are regarded as an integrated network rather than separate problems (Rezaei and Kheirkhah, 2017). A growing number of firms have realized the importance of CLSC optimization (Senthil et al., 2018). Sahyouni et al. (2007) build three types of uncapacitated CLSC network design models that are aimed at minimizing fixed location and transportation costs. Jabbarzadeh et al. (2018) propose a stochastic robust CLSC design and optimization model, which considers lateral transshipment as a reactive strategy to address disruption risks; the authors also develop a

Lagrangian relaxation algorithm to solve the problem efficiently. Moreover, determining whether uncapacitated or capacitated CLSC network faces the same problem that the sub-problems generated remain intractable and require several cutting-plane methods (Zhang et al., 2015).

In summary, many studies on CLSC optimization problems disregarded the uncertainty of new & returned products on network design. Many CLSC design and optimization problems excluded several realistic factors, such as integrated capacitated distribution center, risk pooling to buffer random demands, and value loss related to inventory and transportation time, although these studies have considered uncertainty problems. Several other factors were frequently ignored, such as the savings from constructing an integrated distribution center, and the relationship between distribution center capacity and cost given the economy of scale. However, these ignored factors are crucial to the real-world CLSC management.

This study conducts a comprehensive investigation of a CLSC network design and optimization problem by considering several realistic factors, such as uncertain scenarios and value loss; these factors are related to inventory and transportation time. In addition, certain valid inequalities are added to strengthen the model formulation. In comparison to the existing literature, the model proposed in this study can provide more reasonable CLSC management plans in real-world application.

3. Model formulation and reformulation

We use a relatively simple three-tiered supply chain network as an example to explain the problem background of this study. Figure 1 illustrates that the underlying strategic and operational setting of our problem consists of three types of facilities, namely, a supplier, several capacitated distribution centers (DCs), and retailers, in this network. The relationship among these facilities flows in two directions, that is, forward and reverse. Specifically, the former is the flow of the retailer's order for a product from the DC which is replenished from the supplier. By contrast, the latter is the flow of returned products from the retailer to the corresponding DC and then back to the supplier for remanufacturing. Clearly, DCs can hold stocks of both new & returned products. Thus, there are three types of DCs in the network: forward DCs which just store new products, reverse DCs which just store returned products, and joint DCs which can store both new & returned products. Since satisfying the needs of overall demand in the designed CLSC network generates a certain cost, the goal of our problem is to minimize the total cost, including the fixed construction cost of each DC, transportation cost, working inventory cost, safety stock inventory cost and the time value of returned products. Our problem adopts the risk-pooling strategy, denoting that inventories are maintained at the DCs rather than retailers' sites. Furthermore, in the real-world, a returned product could wait in excess of 3.5 months before remanufacturing. Products that are worth USD 1000 will lose nearly half of their original product value during this waiting time (Zhang et al., 2015). Hence, we also analyze the trade-off between reprocess efficiency and responsive costs when designing our bidirectional supply chain network. In summary, we first determine distribution center locations among alternative sites (denoted by $d, d \in D$) and capacity expansion of each DC. Then, we decide the service allocation of retailers (denoted by $r, r \in R$) under various scenarios (denoted by $s, s \in S$).

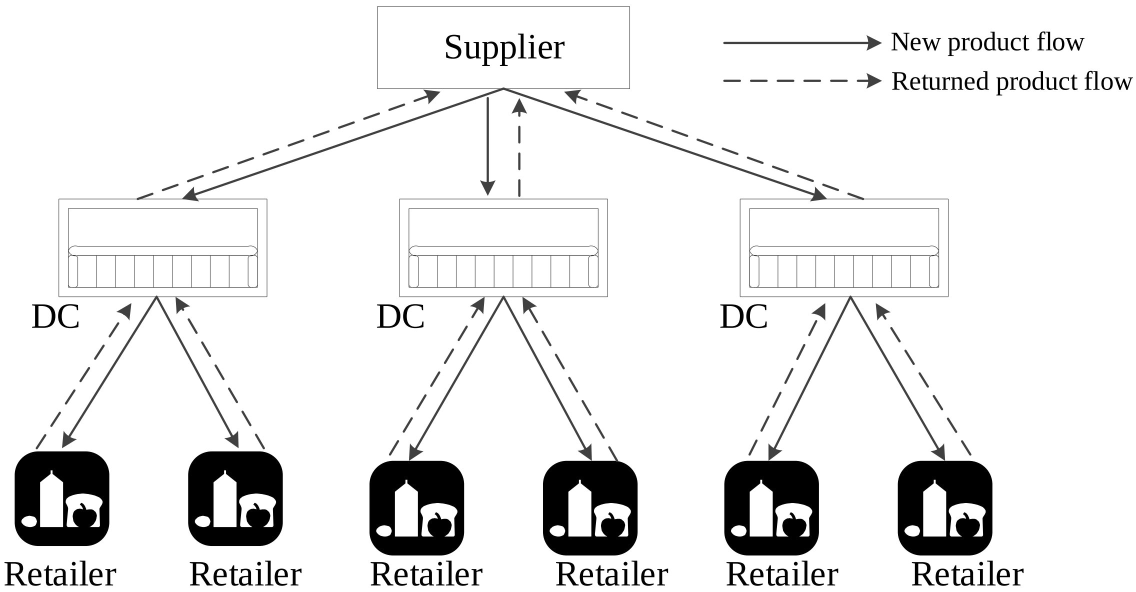


Figure 1: Structure of a three-tiered supply network

3.1. Mathematical model formulation

In this section, a non-linear MIP model is proposed for our problem. The following assumptions are considered in this study.

(1) The demand for new & returned products of each retailer is uncertain and should be fully

satisfied.

(2) The demand for new products is higher than the number of the returned products.

(3) All related costs are deterministic and known.

(4) Transportation capacity is sufficient.

Before formulating the mathematical model for this problem, we list the notations used in this paper as follows.

Indices and sets:

R set of all retailers, $r = 1,2,\dots,|R|$

D set of all candidate DC sites, $d = 1,2,\dots,|D|$

S set of all scenarios, $s = 1,2,\dots,|S|$

Parameters:

a_d^F, a_d^R cost per unit to ship between DC d and the supplier for new/ returned products

b_{rd} cost per unit to ship between DC d and retailer r in forward/ reverse flows

c_d^F, c_d^R base distribution/ collection capacity at DC d

d_{sr} demand (daily) of new products at retailer r under scenario s

e_d^F, e_d^R unit distribution/ collection capacity expansion cost at DC d

f initial price of returned products

g_d^F, g_d^R allowed distribution/ collection capacity expansion at DC d

h inventory holding cost per unit of products per year for each DC

i weight factor associated with the inventory cost in forward/ reverse flows

k daily transportation cost per unit of returned products

l_d lead time in days at a DC d

m returned products' (daily) marginal value of time

N number of days in a year

o_d^F, o_d^R fixed cost of placing an order of new/ returned products at DC d

P_s probability of scenario s

r_{sr} demand (daily) of returned products at retailer r under scenario s

s_d^F, s_d^R fixed (yearly) costs of locating a DC for forward/ reverse flow at DC d

s_d^C fixed location cost savings at joint DC d

t_d^F, t_d^R fixed transportation costs between supplier and DC d for new/returned products

t weight factor associated with the transportation cost in forward/ reverse flows

v weight factor associated with loss in value of returned products

$WI_{sd}^F(\cdot)$ total annual cost of working inventory at forward DC d under scenario s

$WI_{sd}^R(\cdot)$ total annual cost of working inventory at reverse DC d under scenario s

Z_α standard normal deviate such that $P(Z \leq Z_\alpha) = \alpha$

α desired percentage of retailer's order satisfied

Decision variables:

θ_{sd}^F shipment quantity of new products at DC d under scenario s

θ_{sd}^R shipment quantity of returned products at DC d under scenario s

τ_d^F binary, which equals one if candidate d is selected as a forward DC; zero, otherwise

τ_d^R binary, which equals one if candidate d is selected as a reverse DC; zero, otherwise

τ_d^C binary, which equals one if candidate d is selected as a joint DC; zero, otherwise

γ_{rds}^F binary, which equals one if demand of new products of retailer r is served by DC d under scenario s; zero, otherwise

γ_{rds}^R binary, which equals one if returned products of retailer r is served by DC d under scenario s; zero, otherwise

γ_{ds}^R $= (Y_{1ds}^R, Y_{2ds}^R, \dots, Y_{|R|ds}^R)^T$

β_d^F amount of distribution capacity expansion at DC d

β_d^R amount of collection capacity expansion at DC d

In our two-stage stochastic programming model, the first stage addresses design decisions to be made at present, whereas the second stage addresses decisions under uncertainty realized by a given set of scenarios. The location and capacity expansion decisions for the DCs belong to the first stage. Bidirectional network flow decisions belong to the second stage and are determined after a demand-and-return scenario is realized. Our model is designed to minimize the total cost of the first-stage design costs and expected second-stage costs over a given set of scenarios.

Mathematical model (P1):

$$\textit{Minimize} \quad \sum_{d\in D}(s_d^F\tau_d^F + s_d^R\tau_d^R + e_d^F\beta_d^F + e_d^R\beta_d^R - s_d^C\tau_d^C) +$$

$$\sum_{s\in S}\sum_{d\in D} P_s[\sum_{r\in R} tNb_{rd}d_{sr}\gamma_{rds}^F + ihZ_\alpha\sqrt{l_d\sum_{r\in R} d_{sr}\gamma_{rds}^F} + WI_{sd}^F(D_{sd}^F, \theta_{sd}^F) +$$

$\sum_{r\in R} tNb_{rd}r_{sr}\gamma_{rds}^R + WI_{sd}^R(D_{sd}^R,\theta_{sd}^R)] + \sum_{s\in S} P_s[v\sum_{d\in D} R(\gamma_{ds}^R,\theta_{sd}^R)]$

(1)

$$s.t.\ \sum_{d\in D}\gamma_{rds}^F = 1, \sum_{d\in D}\gamma_{rds}^R = 1 \qquad \forall r\in R, \forall s\in S \tag{2}$$

$$\gamma_{rds}^F \le \tau_d^F, \gamma_{rds}^R \le \tau_d^R \qquad \forall r\in R, \forall d\in D, \forall s\in S \tag{3}$$

$$\tau_d^C \le \tau_d^F, \tau_d^C \le \tau_d^R \qquad \forall d\in D \tag{4}$$

$$\theta_{sd}^F + Z_\alpha\sqrt{l_d\sum_{r\in R} d_{sr}\gamma_{rds}^F} + l_d\sum_{r\in R} d_{sr}\gamma_{rds}^F \le c_d^F\tau_d^F + \beta_d^F \qquad \forall d\in D, \forall s\in S \tag{5}$$

$$\theta_{sd}^R \le c_d^R\tau_d^R + \beta_d^R \qquad \forall d\in D, \forall s\in S \tag{6}$$

$$\beta_d^F \le g_d^F\tau_d^F \qquad \forall d\in D \tag{7}$$

$$\beta_d^R \le g_d^R\tau_d^R \qquad \forall d\in D \tag{8}$$

$$\tau_d^F, \tau_d^R, \tau_d^C \in \{0,1\} \qquad \forall d\in D \tag{9}$$

$$\gamma_{rds}^F, \gamma_{rds}^R \in \{0,1\} \qquad \forall r\in R, \forall d\in D, \forall s\in S \tag{10}$$

$$\theta_{sd}^F, \theta_{sd}^R \ge 0 \qquad \forall d\in D,\ \forall s\in S \tag{11}$$

$$\beta_d^F, \beta_d^R \ge 0 \qquad \forall d\in D \tag{12}$$

where :

$$WI_{sd}^F(D_{sd}^F,\theta_{sd}^F) = \begin{cases} o_d^F\frac{D_{sd}^F}{\theta_{sd}^F} + t(t_d^F + a_d^F\theta_{sd}^F)\frac{D_{sd}^F}{\theta_{sd}^F} + \frac{ih}{2}\theta_{sd}^F, & \theta_{sd}^F > 0 \\ 0, & \theta_{sd}^F = 0 \end{cases} \tag{13}$$

$$WI_{sd}^R(D_{sd}^R,\theta_{sd}^R) = \begin{cases} o_d^R\frac{D_{sd}^R}{\theta_{sd}^R} + t(t_d^R + a_d^R\theta_{sd}^R)\frac{D_{sd}^R}{\theta_{sd}^R} + \frac{ih}{2}\theta_{sd}^R, & \theta_{sd}^R > 0 \\ 0, & \theta_{sd}^R = 0 \end{cases} \tag{14}$$

$$D_{sd}^F = N\sum_{r\in R} d_{sr}\gamma_{rds}^F,\ D_{sd}^R = N\sum_{r\in R} r_{sr}\gamma_{rds}^R \qquad \forall d\in D, \forall s\in S \tag{15}$$

$$R(\gamma_{ds}^R,\ \theta_{sd}^R) = R_{inv}(\theta_{sd}^R) + R_{tr}(\gamma_{ds}^R) = mf\left[\frac{N\theta_{sd}^R}{2} + \sum_{r\in R}\left(\frac{b_{rd}+a_d^R}{k}\right)Nr_{sr}\gamma_{rds}^R\right]. \tag{16}$$

The objective function (1) minimizes the total cost, including the fixed and expansion cost of each DC, transportation cost, working inventory cost, safety stock inventory cost and time value of the returned products. We rename the five types of cost defined by using the above parameters and decision variables to facilitate reading:

i. Fixed and expansion cost : $\sum_{d\in D}(s_d^F\tau_d^F + s_d^R\tau_d^R + e_d^F\beta_d^F + e_d^R\beta_d^R - s_d^C\tau_d^C)$;

ii. Transportation cost : $\sum_{s\in S} P_s[\sum_{d\in D}\sum_{r\in R}(tNb_{rd}d_{sr}\gamma_{rds}^F + tNb_{rd}r_{sr}\gamma_{rds}^R)]$;

iii. Working inventory cost : $\sum_{d\in D}\sum_{s\in S} P_s[WI_{sd}^F(D_{sd}^F,\theta_{sd}^F) + WI_{sd}^R(D_{sd}^R,\theta_{sd}^R)]$;

iv. Safety stock inventory cost : $\sum_{d\in D}\sum_{s\in S} P_s(ihZ_\alpha\sqrt{l_d\sum_{r\in R} d_{sr}\gamma_{rds}^F})$;

v. Time value of returned products: $\sum_{s\in S} P_s[v\sum_{d\in D} R(\gamma_{ds}^R,\theta_{sd}^R)]$.

The first kind of cost is the sum of the fixed costs of building a new DC and expansion costs of each DC and the fixed cost savings caused by the joint DC. The cost of forward flows sums the transportation, working inventory, and safety stock inventory costs, whereas the cost of reverse flows has the same cost components, except for the safety stock inventory cost. Equation (13) represents the working inventory cost of new products, which consist of fixed costs for handling orders, supplier-to-DC shipping costs, and the average order holding costs per year. In addition, the working inventory cost of returned products is formulated as Equation (14). Equation (15) states the process for calculating D_{sd}^F and D_{sd}^R. We assume that D_{sd}^R is the total number of returned products of each DC under various scenarios. $R(\gamma_{ds}^R, \theta_{sd}^R)$, as expressed in Equation (16), is the average time value loss of returned products per year; this loss is related to the marginal value of time of returned products. A detailed description of Equations (13), (14), and (16) can be found in Shen et al. (2003), Geyer et al. (2007), and Blackburn et al. (2004), respectively.

Constraints (2) guarantee that each retailer is served by only one DC. Constraints (3) ensure that only open DCs can be assigned. Constraints (4) state that if a DC is assigned to provide both forward and reverse services, then such DC acts as a joint DC. Constraints (5) are the capacity restrictions of each forward DC which summarizes the new product, safety stock under the assumption of normal demands, and covers stock-outs that occur with a probability of α or less and average demand during lead times. Notably, Z_α is the 100αth percentile of a standard normal distribution, i.e., $P(Z \leq Z_\alpha) = \alpha$ where Z is a standard normal random variable. A detailed description of Constraints (5) can be found in Ozsen et al. (2010). Constraints (6) are the capacity restrictions of each reverse DC. Constraints (7)−(8) stipulate that the design expansion capacity is less than its corresponding expansion capacity limit. Lastly, Constraints (9)−(12) define the value range of decision variables.

3.2. Model reformulation

Our problem is formulated as a non-linear MIP model, and finding its optimal solutions in a reasonable amount of time is difficult. We note that our previous model belongs to a new version of the family of joint location-inventory models, which were first proposed by Shen et al. (2003). From Zhang et al. (2015), we can linearize our original model as an equivalent conic quadratic mixed-integer programming (CQMIP) model, which can be directly solved by the CPLEX.

Definition 1. A CQMIP model is an optimization problem of the form:

$Minimize \quad a'x$

$s.t. \quad ||B_i x + c_i||_2 \le d_i' x + e_i, \qquad i = 1, \dots, p$

where $x \in Z^n \times R^m, a \in R^{(n+m)}, B_i \in R^{n_i \times (n+m)}, c_i \in R^{n_i}, d_i \in R^{(n+m)}, e_i \in R, ||\cdot||_2$ is the Euclidean norm, and all parameters are rational.

Proposition 1. The following CQMIP model (P2) is equivalent to the previous non-linear MIP model (P1).

CQMIP (P2):

$$Minimize \quad \sum_{d\in D}(s_d^F \tau_d^F + s_d^R \tau_d^R + e_d^F \beta_d^F + e_d^R \beta_d^R - s_d^C \tau_d^C) + \sum_{s\in S} P_s \{\sum_{d\in D}\left(ihZ_\alpha x_{sd} + \frac{ih}{2} y_{sd}\right) + \sum_{d\in D}\sum_{r\in R}\left[tN(b_{rd} + a_d^F) d_{sr}\gamma_{rds}^F + (t + \frac{vmf}{k})(b_{rd} + a_d^R) N r_{sr} \gamma_{rds}^R + \frac{vmfN+ih}{2} z_{sd}\right]\} \tag{17}$$

$$s.t. \quad \sum_{d\in D}\gamma_{rds}^F = 1, \sum_{d\in D}\gamma_{rds}^R = 1 \qquad \forall r \in R, \forall s \in S \tag{18}$$

$$\gamma_{rds}^F \le \tau_d^F, \gamma_{rds}^R \le \tau_d^R \qquad \forall r \in R, \forall d \in D, \forall s \in S \tag{19}$$

$$\tau_d^C \le \tau_d^F, \tau_d^C \le \tau_d^R \qquad \forall d \in D \tag{20}$$

$$\beta_d^F \le g_d^F \tau_d^F \qquad \forall d \in D \tag{21}$$

$$\beta_d^R \le g_d^R \tau_d^R \qquad \forall d \in D \tag{22}$$

$$x_{sd}^2 \ge l_d \sum_{r\in R} d_{sr}\,(\gamma_{rds}^F)^2 \qquad \forall d \in D,\ \forall s \in S \tag{23}$$

$$\frac{1}{2}(y_{sd} + \theta_{sd}^F)^2 \ge \frac{2(o_d^F + tt_d^F)}{ih} N \sum_{r\in R} d_{sr}(\gamma_{rds}^F)^2 + \frac{3}{2}(\theta_{sd}^F)^2 + \frac{1}{2}y_{sd}^2 \quad \forall d \in D,\ \forall s \in S \tag{24}$$

$$\frac{1}{2}(z_{sd} + \theta_{sd}^R)^2 \ge \frac{2(o_d^R + tt_d^R)}{vmfN+ih} N \sum_{r\in R} r_{sr}(\gamma_{rds}^R)^2 + \frac{3}{2}(\theta_{sd}^R)^2 + \frac{1}{2}z_{sd}^2 \quad \forall d \in D,\ \forall s \in S \tag{25}$$

$$\theta_{sd}^F + Z_\alpha x_{sd} + l_d \sum_{r\in R} d_{sr}\gamma_{rds}^F \le c_d^F \tau_d^F + \beta_d^F \qquad \forall d \in D,\ \forall s \in S \tag{26}$$

$$\theta_{sd}^R \le c_d^R \tau_d^R + \beta_d^R \qquad \forall d \in D, \forall s \in S \tag{27}$$

$$\tau_d^F, \tau_d^R, \tau_d^C \in \{0,1\} \qquad \forall d \in D \tag{28}$$

$$\gamma_{rds}^F, \gamma_{rds}^R \in \{0,1\} \qquad \forall r \in R, \forall d \in D, \forall s \in S \tag{29}$$

$$\theta_{sd}^F, \theta_{sd}^R \ge 0 \qquad \forall d \in D,\ \forall s \in S \tag{30}$$

$$\beta_d^F, \beta_d^R \ge 0 \qquad \forall d \in D. \tag{31}$$

Proof: In order to convert our previous model P1 into the CQMIP model P2, we add a conic transformation. First, we introduce three sets of auxiliary variables, named x_{sd}, y_{sd}, z_{sd}, which satisfy the following inequalities:

$$x_{sd} \geq \sqrt{l_d \sum_{r\in R} d_{sr}\gamma_{rds}^F} \tag{32}$$

$$\frac{ih}{2} y_{sd} \geq (o_d^F + tt_d^F)\frac{D_{sd}^F}{\theta_{sd}^F} + \frac{ih}{2}\theta_{sd}^F \tag{33}$$

$$\left(\frac{vmfN}{2} + \frac{ih}{2}\right) z_{sd} \geq (o_d^R + tt_d^R)\frac{D_{sd}^R}{\theta_{sd}^R} + (\frac{vmfN}{2} + \frac{ih}{2})\theta_{sd}^R. \tag{34}$$

Because γ_{rds}^F is a binary variable and $\gamma_{rds}^F = (\gamma_{rds}^F)^2$, we can convert previous inequalities again as follows:

$$x_{sd}^2 \geq l_d \sum_{r\in R} d_{sr} (\gamma_{rds}^F)^2 \tag{35}$$

$$\frac{1}{2}(y_{sd} + \theta_{sd}^F)^2 \geq \frac{2(o_d^F + tt_d^F)}{ih} N \sum_{r\in R} d_{sr}(\gamma_{rds}^F)^2 + \frac{3}{2}(\theta_{sd}^F)^2 + \frac{1}{2} y_{sd}^2 \tag{36}$$

$$\frac{1}{2}(z_{sd} + \theta_{sd}^R)^2 \geq \frac{2(o_d^R + tt_d^R)}{vmfN + ih} N \sum_{r\in R} r_{sr}(\gamma_{rds}^R)^2 + \frac{3}{2}(\theta_{sd}^R)^2 + \frac{1}{2} z_{sd}^2. \tag{37}$$

After placing related parts with these three sets of auxiliary variables, we can obtain our equivalent CQMIP model, and it can be solved by the CPLEX directly because it is a linear model.

3.3. Valid inequalities

Generally, commercial software packages use a branch-and-bound algorithm for solving CQMIP models, and the performance of these packages can be significantly improved by strengthening the models with certain cuts (Atamtürk et al., 2012). This section therefore applies two types of structural cutting planes.

The first type is polymatroid inequalities, which utilize submodularity and reformulate Constraints (23)−(25) to strengthen the convex relaxation of our CQMIP model. Inequalities $l_d \sum_{r\in R} d_{sr}\gamma_{rds}^2 \leq x_{sd}^2$ and $\sqrt{l_d \sum_{r\in R} d_{sr}\gamma_{rds}} \leq x_{sd}$ are equivalent because $x_{sd} > 0, \forall s \in S, \forall d \in D$, and $\gamma_{rds}^2 = \gamma_{rds}, \forall r \in R, \forall d \in D, \forall s \in S$. The latter inequalities demonstrate a submodular form given the concavity and the non-negativity of the square root function. Several definitions are introduced before showing polymatroid inequalities. We drop the superscripts F and R in the following description to simplify the notation.

Definition 2. A set function $g: 2^I \to R$ is submodular if $g(M) + g(N) \geq g(M \cup N) + g(M \cap N)$ for all $M, N \in I$.

Definition 3. For a submodular function g on I, the polyhedron $EP_g := \{\pi \in R^I: \pi(M) \leq g(M), \forall M \subseteq I\}$ is regarded as the extended polymatroid related to g if $g(\emptyset) = 0, \pi(M) = \sum_{i \in M} \pi_i$.

For an extended polymatroid EP_g, Atamtürk and Narayanan (2008) present that the extended polymatroid inequalities $\pi y \leq w$ with $\pi \in EP_g$ are valid for the lower convex envelope of g: $Q_g := \text{conv}\{(\gamma, x) \in \{0,1\}^{|I|} \times R: g(\gamma) \leq x\}$. When the inequalities are defined by the extreme points of the extended polymatroid EP_g, they are called extremal extended polymatroid inequalities.

Proposition 2. Let Q_f denote the lower convex envelope of the sets of solutions that satisfy constraints (23): $Q_f = \text{conv}\{(\gamma_{ds}, x_{sd}) \in \{0,1\}^{|I|} \times R: x_{sd} \geq g(M) = \sqrt{l_d \sum_{i \in M} d_{sr}}\ \forall M \subseteq I\}$. So, the inequality $\sum_{r \in R} \pi_r \gamma_{rds} \leq x_{sd}$ is valid for Q_f, where $\pi_r = \sqrt{l_d \sum_{r \in M(r)} d_{sr}} - \sqrt{l_d \sum_{r \in M(r-1)} d_{sr}} \in EP_g$, $M = \{r | \gamma_{rds} = 1\}, M(r) = \{(1), (2), \ldots, (r)\}, 1 \leq r \leq |R|$ for some permutation. This valid inequality is an extremal extended polymatroid inequality of Q_f.

Proposition 3. Let Q_u denote the lower convex envelope of the sets of solutions that satisfy constraints (24)−(25). The inequality $\sum_{r \in R} \pi_r \gamma_{rds} \leq y_{sd} + \theta_d$ is valid for the lower convex envelope of the sets of solutions that satisfy constraints (24) − (25), where $\pi_r = \sqrt{8H_d N \sum_{r \in M(r)} d_{sr}} - \sqrt{8H_d N \sum_{r \in M(r-1)} d_{sr}}$, $(H_d^F = \frac{2(o_r^F + tt_r^F)}{ih}, H_d^R = \frac{2(o_r^R + tt_r^R)}{vmfN + ih})$, for $M = \{r | \gamma_{rds} = 1\}, M(r) = \{(1), (2), \ldots, (r)\}, 1 \leq r \leq |R|$ for some permutation. This valid inequalities is also an extremal extended polymatroid inequality of Q_u.

The detailed proof description of the abovementioned valid inequalities can be found in Zhang et al. (2015). Although there are exponentially many extremal extended polymatroid inequalities, only a small subset of them is needed in the branch-and-bound search tree. It is noted that, given a solution, a violated polymatroid cut can be found by employing the separation problem. Specifically, we use a greedy algorithm introduced by Edmonds (1971) and Atamtürk et al. (2012). Main steps of Edmond's greedy algorithm are described as follows. For each $d \in D, s \in S$, do:

1. Given $\gamma_{ds}^* \in [0,1]^{|R|}$ and x_{ds}^*, sort γ_{rds}^* in nonincreasing order $\gamma_{(1)ds}^* \geq \gamma_{(2)ds}^* \geq \cdots$.

2. For $r = 1, \ldots, |R|$, let $M_r = \{(1), (2), \ldots, (r)\}$ and $\pi_{rs} = \sqrt{l_d \sum_{k \in M(r)} d_{(ks)}} - \sqrt{l_d \sum_{k \in M(r-1)} d_{(ks)}}$.
3. If $\pi\gamma_{ds}^* > x_{ds}^*$, we add the extended polymatroid cut $\pi\gamma_{ds} \leq x_{sd}$ to the formulation.

Besides the abovementioned extended polymatroid inequalities, we also present some extended cover cuts which are derived from non-linear knapsack relaxations of the formulation. To this end, we relax the left-hand side of Constraints (26) by dropping θ_{sd}^F and F, and replace the right-hand side with C to obtain the 0-1 knapsack constraint as follows.

$$Z_\alpha \sqrt{l_d} \sqrt{\sum_{r \in R} d_{sr}\gamma_{rds}} + l_d \sum_{r \in R} d_{sr}\gamma_{rds} \leq C_d \tag{38}$$

In order to simplify the notation, we drop the subscripts d and s when defining the inequalities. For inequality (38), define the set function $g: 2^I \to R$, where $g(S) = Z_\alpha \sqrt{l}\sqrt{d(S)} + ld(S), d(S) := \sum_{r \in S} d_r$. Using submodularity of g, Atamtürk et al. (2012) present cover and extended cover cuts for the submodular knapsack set,

$\Gamma = \left\{\gamma \in \{0,1\}^{|R|}: g(\gamma) \leq C\right\} = \left\{\gamma \in \{0,1\}^R: Z_\alpha \sqrt{l}\sqrt{\sum_{r \in R} d_r\gamma_r} + l\sum_{r \in R} d_r\gamma_r \leq C\right\}$. And they show that given a subset of indices $S \subseteq I$ and the conic quadratic 0-1 knapsack set Y, we can find valid cover inequalities which depend on the cover set.

Definition 4. $S \subseteq I$ is called a cover for Y if $Z_\alpha \sqrt{l}\sqrt{d(S)} + ld(S) > C$.

The corresponding cover inequality of cover S: $\sum_{r \in S} \gamma_r \leq |S| - 1$ is valid for γ according to Atamtürk and Narayanan (2008). Besides, cover inequalities can be strengthened by extending them with non-cover variables. Before introducing extended cover inequalities, we first define the difference function and the notion of extension.

Definition 5. Given a set function g on I and $i \in I$, the difference function p is defined as $p_i(S) := g(S \cup i) - g(S)$ for $S \subseteq I \backslash i$.

Definition 6. Let $\pi = (k_{(1)}, \ldots, k_{(|I|-|S|)})$ be a permutation of the indices in $I \backslash S$. Define $S_l = S \cup \{k_{(1)}, \ldots, k_{(l)}\}$ for $l = 1, \ldots, |I| - |S|$, where $S_0 = S$. The extension of S corresponding to permutation π is $E_\pi(S) := S \cup U_\pi(S)$, where $U_\pi(S) = \{k_{(l)}: p_{k_{(l)}}(S_{l-1}) \geq p_i(\emptyset), \forall i \in S\}$.

Given cover S and permutation π, the corresponding extended cover inequality

$\sum_{r\in E_\pi(S)} \gamma_r \leq |S| - 1$ is valid for γ (Atamtürk et al., 2012). We will add abovementioned valid inequalities to our model to speed up the solution process.

4. Tabu search algorithm

Notably, the CLSC design and optimization problem can be classified as the class of NP-hard problem (Krarup and Pruzan, 1983; Schrijver, 2003), in which computational complexity increases exponentially with the growth in the number of retailers, DCs, and scenarios. Large-scale problem instances of our problem cannot be solved by the commercial solver, such as CPLEX, within a reasonable time. Occasionally, the solution is suspended by an "out-of-memory" error. From Atamtür et al. (2012), we find that our equivalent CQMIP model can be added some valid inequalities to speed up the solving process for small-scale instances while solving large-scale instances is still time-consuming. Hence, utilizing heuristics and metaheuristics in solving large-scale instances of the CLSC problem is unavoidable. Some studies related to the CLSC problem (Easwaran and Üster, 2009; Noham and Tzur, 2018; Punyim et al., 2018) find that the tabu search algorithm performs exceptionally well, and optimal solutions are obtained in almost all cases according to their computational results. Therefore, we design a tabu search algorithm for solving the proposed model.

Glover (1989) initially proposes the tabu search algorithm, which is an adaptive local iteration search within a search space, by for solving several combinatorial optimization problems. The core idea of tabu search algorithm is to prevent the search process from being trapped in the local optimum by avoiding cycling. The tabu attribute moves from one solution to another and diversifies the solutions iteratively to find an improved version (Vivaldini et al., 2016). In each iteration, the tabu search algorithm is applied to explore its neighborhood to avoid the local optimum and improve its quality. Additionally, the optimum admissible movement is that of the minimum evaluation in the neighborhood of the current solution (assuming that a minimizing problem is being solved). Several main steps in the tabu search algorithm are presented as follows.

4.1. Solution encoding and evaluation

An encoding scheme is necessary for a solution to be compatible with the tabu search representation. Our encoding scheme consists of two segments that control two decision variables, namely, τ_d^F and τ_d^R. Simultaneously, another decision variable, τ_d^C, equals one if both τ_d^F and τ_d^R equal one and zero otherwise. For the example in Figure 2, we assume that the number of total DCs is three. We first generate six random numbers with the range [0, 1] as genes (the first three values are responsible for τ_d^F, whereas the last three values correspond to τ_d^R). Then, we estimate our genes by comparing with 0.5. We designate one gene value as Y if this gene value is greater than 0.5, else we evaluate the gene value as N. Then, we assign 1 to all decision variables, which estimate is Y, and assign 0 to all decision variables, which estimate is N. Finally, we obtain $\tau_1^F = \tau_2^F = \tau_3^R = 0$ and $\tau_3^F = \tau_1^R = \tau_2^R = 1$. Moreover, DCs, which both τ_j^F and τ_j^R equal one, are lacking; thus, we obtain $\tau_1^C = \tau_2^C = \tau_3^C = 0$.

	τ_j^F			τ_j^R		
Index of DCs	1	2	3	1	2	3
Genes	0.2	0.4	0.8	0.7	0.6	0.3
Estimate	N	N	Y	Y	Y	N
Assignment	0	0	1	1	1	0

Figure 2: The encoding and decoding scheme

The solutions are evaluated in accordance with their objective function value, that is, the minimization of total costs. The corresponding solutions are improved when the objective function value is smaller.

4.2. Initialization

Initially, we require an initial CLSC network design as a seed for representing a point within the search space. This initial solution is either the current or the best-known solution. We generate a set of random genes, as discussed in Section 4.1, and consider these genes as the initial solution. Then, a set of neighboring solutions is generated from the current one to search for an enhanced solution.

4.3. Moves

A neighborhood structure for moving from the current to other solutions must be established after generating an initial solution. To this end, we use a stochastic descent method in which numerous neighboring solutions are stochastically generated in accordance with the multi-swap neighborhood process. In this process, we generate a set of random numbers within the range [0, 1] and use the multi-swap neighborhood structure if this number is greater than 0.5, as demonstrated in Figure 3. The number of swaps is set to be high at the beginning of the search process but decreases with the increase in the iterations. Figure 3 exhibits the three swap neighborhood structures as an example. The optimum non-tabu and admissible solution is selected from the neighboring solutions and set as the current solution after generating the neighboring solutions.

		τ_j^F			τ_j^R	
index of DCs	1	2	3	1	2	3
Before	0.7	0.7	0.2	0.7	0.4	0.3
After	0.2	0.4	0.7	0.3	0.7	0.7

Figure 3: The multi-swap neighborhood procedure (example of $1 \leftrightarrow 3, 2 \leftrightarrow 5, 4 \leftrightarrow 6$)

4.4. Tabu list

A tabu list of size m, as an essential element of the tabu search algorithm, remains the last m moves of the previous search and prevents their repetition. Specifically, the tabu list is a dynamic memory, which stores the attributes of new solutions and blocks them to be revisited until a number of iterations have been executed. The tabu attributes are released on a first-in−first-out basis after each iteration; the most recent move is added into the tabu list. The tabu list is checked, and a move is permitted only if it is not in the tabu list when a multi-swap neighborhood procedure occurs.

4.5. Aspiration criterion

We sometimes override the tabu status of a move when it is satisfied to ensure an increased flexibility in our search process. If an improved solution (move) is generated by the attributes that are already in the tabu list and should not be allowed by previous rules, then the solution will be accepted in accordance with the aspiration criterion.

4.6. Diversification

We adopt a diversification method to widely explore and find an improved new solution. All newly generated solutions should be compared with the optimum existing solution. The best-known solution will be updated if a generated solution is better than the best-known solution. If the best-known solution has not been improved for a certain number of iterations, then our search process will be directed to another area of the solution space by randomly generating a new solution (Zhen et al., 2016). Hence, a diversification constantly leads to an increase in the number of swaps, which can generate added diverse solutions.

4.7. Stopping criterion

The stopping criterion determines the end time of our algorithm. In this study, we use two stopping criteria, that is, the maximum number of iterations searched by using the tabu search, and the variance of the solutions smaller than a given threshold (Zhen, 2016). The tabu search algorithm stops if one of the above mentioned stopping criterion is satisfied.

5. Computational experiments

This section presents the computational experiments conducted to evaluate the proposed model and verify the manner which adding valid inequalities accelerates the computation.

5.1. Experimental setting

We summarize our parameter values, where most of which are similar to Zhang et al. (2015). d_{sr} and r_{sr} (demand and returned products at retailer r under scenario s) follow a uniform distribution ($\sim U$ (10, 20)). P_s is determined by the number of scenarios, $P_s = \frac{1}{\text{the number of scenarios}}$. In addition, Table 1 and Table 2 list the parameter values of the proposed model and tabu search algorithm, respectively.

Table 1: Parameter values of the model

Parameter	Value	Parameter	Value	Parameter	Value
b_{rd}	$\sim U(0.5,2.9)$	h	10	k	100
v	1	N	365	m	10%
Z_α	1.96	α	97.5%	c_d^F, c_d^R	10
l_d	1	t	0.001	g_d^F, g_d^R	10
f	1	i	0.1	S_d^F, S_d^R	10
o_d^F, o_d^R	10	t_d^F, t_d^R	10	a_d^F, a_d^R	5

Table 2: Parameter values of tabu search algorithm

Parameter	Value
$max\ interation$	30
$tabu\ list\ size$	15
$N\ size$	10

The mathematical model is implemented by CPLEX 12.5.1 (Visual Studio 2015, C#) on a PC (Intel Core i7, 2.6 G Hz; Memory, 8 G).

5.2. Performances of the model

In our study, the number of scenarios has an important effect because our DC location problem is a stochastic programming model. The determination of the most appropriate number of scenarios by testing various numbers of scenarios is considered a critical step. Five sets of scenarios under the scale of three DCs and ten retailers, that is, 10, 20, 50, 80, and 100, are provided. Each set has ten randomly generated cases. Table 3 summarizes the computational results, including the minimum (Min), maximum (Max), gap (Gap Max-Min), average (Avg.), and standard deviation values (S. D.), and average CPU running time (Avg. CPU time). It is noted that the gap and standard deviation values decrease with the increase in the number of scenarios. The standard deviation values do not decrease rapidly when the number of scenarios exceeds 80, although the CPU running time increases sharply with the number of scenarios. Hence, in the following computation experiments, we set the number of scenarios to 80.

Table 3: Testing the proposed model under different numbers of scenarios

Number of Scenarios	Min	Max	Gap Max-Min	Avg.	S.D.	Avg.CPU Time(s)
10	578	602	24	594.8	20.79006	0.9

20	577	678	101	590.3	31.33680	1.5
50	728	778	50	772.2	15.54778	785.5
80	777	778	1	777.1	0.333333	1386.4
100	777	778	1	777.1	0.252982	3519.3

Table 4 displays the results solved through three methods: using CPLEX directly (abbreviated to CPLEX), CPLEX with added valid inequalities (abbreviated to CPLEX+cuts), and the tabu search algorithm (abbreviated to Tabu). The results of the first two methods contain objective (OBJ) gap and CPU running time. In particular, the tabu method results consist of OBJ values and CPU running time.

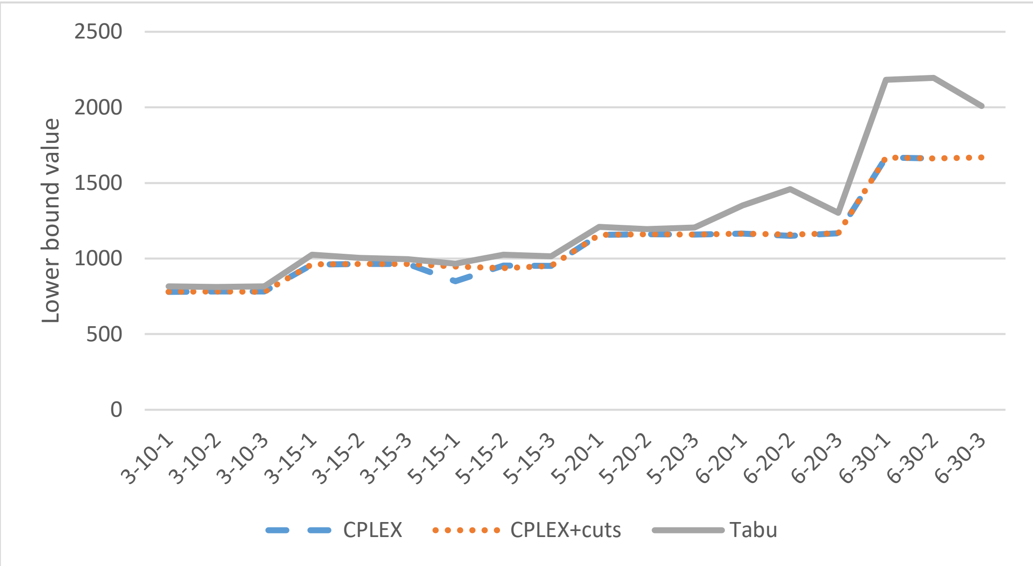
Table 4: Results solved through three methods

Case ID	CPLEX		CPLEX+cuts		Tabu	
	OBJ Gap	Time(s)	OBJ Gap	Time(s)	OBJ	Time(s)
3-10-1	0.01	92	0.01	33	817	927
3-10-2	0.01	255	0.01	125	812	892
3-10-3	0.01	119	0.01	37	817	984
3-15-1	0.01	212	0.01	138	1025	1317
3-15-2	0.01	175	0.01	60	1005	1089
3-15-3	0.01	264	0.01	115	995	1161
5-15-1	0.1833	3600limit	0.0931	3600limit	967	2981
5-15-2	0.3692	3600limit	0.1042	3600limit	1025	3006
5-15-3	0.3519	3600limit	0.0888	3600limit	1015	2976
5-20-1	0.3005	3600limit	0.0775	3600limit	1210	3209
5-20-2	0.2999	3600limit	0.0774	3600limit	1195	3379
5-20-3	0.2046	3600limit	0.0771	3600limit	1205	3380
6-20-1	0.4008	3600limit	0.2451	3600limit	1351	3600limit
6-20-2	0.4042	3600limit	0.2962	3600limit	1459	3600limit
6-20-3	0.3996	3600limit	0.2664	3600limit	1302	3600limit
6-30-1	0.7346	3600limit	0.3552	3600limit	2183	3600limit
6-30-2	0.7280	3600limit	0.4237	3600limit	2195	3600limit
6-30-3	–	3600limit	0.4431	3600limit	2008	3600limit

Note: In 'Case ID', the first two values denote the number of DCs and retailers, respectively. The en dash means we did not find any solution within the time limits.

We find that the CPU running time increases with the instance scale. We note that the OBJ gap values are much larger in the CPLEX method than in the CPLEX with valid inequalities. The time of CPLEX with valid inequalities is half that of the time of CPLEX method, thereby indicating that valid inequalities, including polymatroid inequalities and extended cover cuts, can accelerate the model solving process and improve the performance of the solution quality.

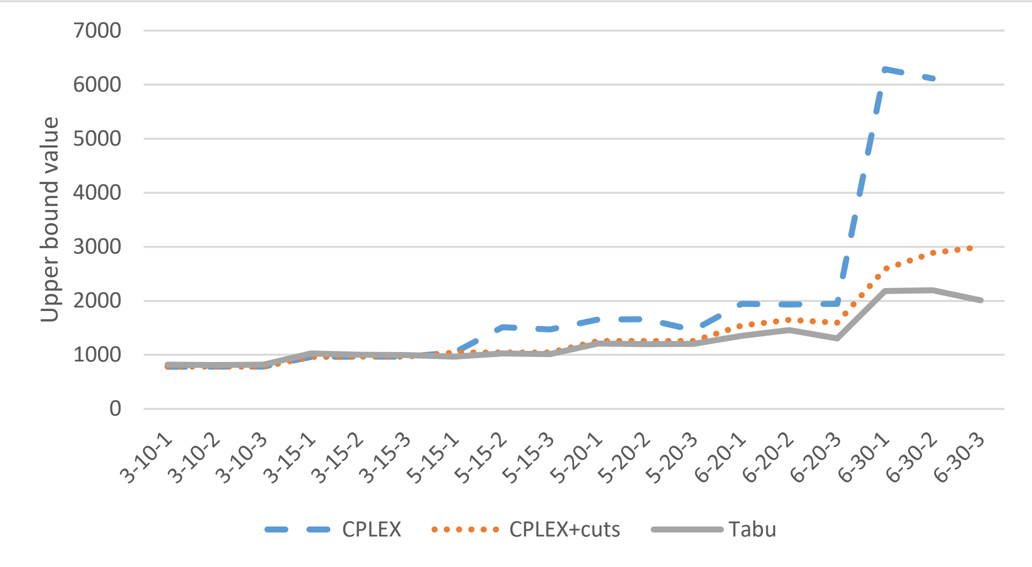
However, finding the optimal solution is difficult when the CPLEX directly solves large-scale problem instances under 3600 seconds limit, such as 5-15-1. Meanwhile, the tabu method can find an improved solution when solving large-scale problem instances. We summarize the lower and upper bounds of the abovementioned methods in Figures 4 and 5, respectively to compare the results solved by CPLEX directly, CPLEX with valid inequalities, and the tabu search algorithm. The value of the upper and lower bounds of the tabu search algorithm is the same OBJ value as displayed in Table 4. Furthermore, we did not find any solution for the last one sets of problem instances (6-30-3) when using the CPLEX directly within the time limits. Hence, Figures 4 and 5 do not exhibit these values of the CPLEX method.



Note: Case id 3-10-1 means the first computational example of 3 DCs and 10 retailers.

Figure 4: Solution lower bound comparison between results of CPLEX, CPLEX with valid inequalities and tabu search algorithm

In Figure 4, we find that the solution lower bound is lower in the CPLEX method than in the CPLEX with valid inequalities, thereby denoting that valid inequalities can accelerate the model solving process and provide a fast means with which to find an improved solution. In addition, the tabu search algorithm outperforms the CPLEX in terms of large-scale problem instances.



Note: Case id 3-10-1 means the first computational example of 3 DCs and 10 retailers.

Figure 5: Solution upper bound comparison between results of CPLEX, CPLEX with valid inequalities and tabu search algorithm

In Figure 5, we find that the solution upper bound is higher in the CPLEX method than in the CPLEX with valid inequalities, thus demonstrating that valid inequalities can improve the performance of the solution quality. The tabu search algorithm can also find acceptable solutions when solving large-scale problem instances within a reasonable amount of time.

5.3. Sensitivity analysis

This study considers many realistic factors simultaneously, which means the model will have many different set of input parameters. Hence, we use the sensitivity analysis to study how sensitive the model result would be with different input parameters. We start by varying the inventory and transportation weights as shown in Table 5. The subsequent analysis studies the impact of the marginal value of time of returned products as shown in Table 6−7.

Studying the impact of the inventory and transportation weights, we take an example of 2 DCs and 5 retailers, other input settings are similar with the Table 1 except the inventory and transportation weights. As shown in Table 5, we find that total cost, and CPU time increase when inventory weight (i) or transportation weight (t) increase.

Table 5: Sensitivity analysis of the inventory and transportation weights

i	t	Objective value	CPU time (s)

0.1	0.001	642	5
0.1	0.002	643	5
0.1	0.003	643	5
0.1	0.004	644	5
0.1	0.005	644	6
0.2	0.002	666	28
0.5	0.005	718	470
1	0.005	783	1938
2	0.005	881	2844
5	0.005	1097	3565

Next is the impact of returned products' marginal value of time. Time sensitivity of the returned products' price has an influence on products' marginal value of time. We take an example of 5 DCs and 15 retailers. Table 6 lists the parameters of this experiment. Table 7 demonstrates the effect of value of time (m) given daily transportation cost per unit $k = 500$. The marginal value of time (m) is set to $1\%, 10\%, 30\%, 50\%, 70\%, \text{and } 90\%$. $\mathrm{DC}^{\mathrm{F}}, \mathrm{DC}^{\mathrm{R}}, \text{and } \mathrm{DC}^{\mathrm{C}}$ record the number of open forward DCs, reverse DCs, and joint DCs, respectively. As shown in Table 7, we find that fewer reverse DCs are needed for highly time-sensitive (higher m) returned products, which is because the storage time of time-sensitive returned products has been reduced to get more salvage value from them. This means highly time-sensitive returned products need more shipments of smaller quantity. At the same time, the storage space needed decreases. Besides, total cost, and CPU time increase when marginal value of time m increases. In most cases, joint DCs are preferred because of the cost saving.

Table 6: Parameter values in the experiment of the impact of marginal value of time

Parameter	Value
v	10
i	0.1
t	0.005
k	500

Table 7: Sensitivity analysis of returned products' marginal time value

$m(\%)$	Objective value	CPU time (s)	DC^{F}	DC^{R}	DC^{C}
1	780	29	1	2	1
10	1059	3474	1	1	1
30	1394	876	1	1	1
50	1628	901	1	1	1
70	1819	1351	1	0	1
90	1983	1320	1	0	1

6. Conclusion

This study investigates the CLSC network design and optimization problem and is aimed at minimizing total costs, including fixed and expansion cost of each DC, transportation cost, working inventory cost, safety stock inventory cost, and time value of returned products. We also consider the trade-off between reprocess efficiency and responsive costs when making location decisions. Owing to the complexity of our problem, we build a two-stage stochastic non-linear model and transform the previous model into a conic quadratic MIP model. This model can be solved efficiently in certain cases by using the CPLEX directly. Certain valid inequalities, such as polymatroid inequalities and extended cover cuts, are added to improve the efficiency of the branch-and-cut algorithm and quality of the solutions. We also utilize the tabu search algorithm to solve large-scale problem instances efficiently.

We have made three main contributions with respect to the related literature despite numerous studies on the CLSC network design.

(1) This study integrates several interconnected decisions: capacitated distribution centers, uncertainty demands of new & returned products, risk pooling to buffer random demands, savings from collocating a joint distribution center, value loss related to inventory and transportation time, relationships between distribution center capacity and cost, facility location and capacity determination, services provided by the DCs selection, and product recovery and remanufacturing. Few studies have considered the abovementioned realistic factors simultaneously. A novel model for the CLSC network design is proposed and then converted to a conic quadratic MIP model.

(2) We find that valid inequalities, such as polymatroid inequalities and extended cover cuts,

are computationally beneficial for the proposed model based on extensive quantitative computational experiments.

(3) After conducting quantitative computational experiments, we conclude several interesting managerial insights that can be of great significance in practical application. For example, from the DC type perspective, joint DCs are preferred because of the cost saving. Besides, marginal value of time of returned products has an influence on the location and inventory decisions, fewer reverse DCs are needed for highly time-sensitive returned products. We also show the effects of inventory and transportation cost on total cost.

However, this study also has limitations. In our future study, we can consider additional realistic factors. For example, this model can be naturally extended to incorporate multiple periods and multiple products. We can also incorporate related decision issues, such as ordering, backlogging, and forecasting problems. Moreover, we can consider three pillars of sustainability as objective functions, that is, economic through Net Present Value, environmental through a Life Cycle Analysis methodology (i.e., ReCiPe), and social through a developed GDP-based metrics.

Acknowledgment

This study is sponsored by the National Natural Science Foundation of China (Nos. 71671107, 71422007, 71701178), Shanghai Social Science Research Program (2014BGL006), and Massey University Research Fund.

References

Atamtürk A., Berenguer G. and Shen Z. J. (2012) A conic integer programming approach to stochastic joint location-inventory problems. *Operations Research* 60(2): 366−381.

Atamtürk A. and Narayanan V. (2008) Polymatroids and mean-risk minimization in discrete optimization. *Operations Research Letters* 36(5): 618−622.

Blackburn J. D., Guide V. D. R., Souza G. C. and Wassenhove L.N.V. (2004) Reverse supply chains for commercial returns. *California Management Review* 46(2): 6−22.

Drezner T., Drezner Z. and Kalczynski P. (2015) A leader–follower model for discrete competitive facility location. *Computers & Operations Research* 64(C): 51−59.

Easwaran G. and Üster H. (2009) Tabu search and benders decomposition approaches for a capacitated closed-loop supply chain network design problem. *Transportation Science* 43(3): 301-320.

Edmonds J. (1971) Matroids and the greedy algorithm. *Mathematical Programming* 1(1): 127−136.

Fischetti M., Ljubić I. and Sinnl M. (2016) Benders decomposition without separability: A computational study for capacitated facility location problems. *European Journal of Operational Research* 253(3): 557−569.

Fleischmann M., Beullens P., Bloemhof-Ruwaard J. M. and Wassenhove L. N. V. (2001) The impact of product recovery on logistics network design. *Production & Operations Management* 10(2): 156−173.

Fleischmann M., Bloemhof-Ruwaard J. M., Dekker R., Laan E. V. D., Nunen J. A. E. E. V. and Wassenhove L.N.V. (1997) Quantitative models for reverse logistics: A review. *European Journal of Operational Research* (103): 1−17.

Geyer R., Wassenhove L. N. V. and Atasu A. (2007) The economics of remanufacturing under limited component durability and finite product life cycles. *Management Science* 53(1): 88−100.

Glover F. (1989) Tabu search—Part I. *ORSA Journal on Computing* 1(1): 89−98.

Govindan K., Soleimani H. and Kannan D. (2015) Reverse logistics and closed-loop supply chain: A comprehensive review to explore the future. *European Journal of Operational Research* 240(3): 603−626.

Guide V. D. R. J. and Van-Wassenhove L. N. (2009) The evolution of closed-loop supply chain research. *Operations Research* 57(1): 10−18.

Haddadsisakht A. and Ryan S. M. (2018) Closed-loop supply chain network design with multiple transportation modes under stochastic demand and uncertain carbon tax. *International Journal of Production Economics* 195: 118−131.

Jabbarzadeh A., Haughton M. and Khosrojerdi A. (2018) Closed-loop supply chain network design under disruption risks: A robust approach with real world application. *Computers & Industrial Engineering* 116: 178−191.

Krarup J. and Pruzan P. M. (1983) The simple plant location problem: Survey and synthesis. *European Journal of Operational Research* 12(1): 36−81.

Li Y., Xu F. and Zhao X. (2017) Governance mechanisms of dual-channel reverse supply chains with informal collection channel. *Journal of Cleaner Production* 155: 125−140.

Listeş O. (2007) A generic stochastic model for supply-and-return network design. *Computers & Operations*

Research 34(2): 417−442.

Ljubić I. and Moreno E. (2018) Outer approximation and submodular cuts for maximum capture facility location problems with random utilities. *European Journal of Operational Research* 266(1): 45−56.

Melo M. T., Nickel S. and Saldanha-Da-Gamad F. (2009) Facility location and supply chain management – A review. *European Journal of Operational Research* 196(2): 401−412.

Mota B., Gomes M. I., Carvalho A. and Barbosa-Povoa A. P. (2018) Sustainable supply chains: an integrated modelling approach under uncertainty. *Omega* 77: 32−57.

Nagy G. and Salhi S. (2007) Location-routing: Issues, models and methods. *European Journal of Operational Research* 177(2): 649−672.

Noham R. and Tzur M. (2018) Designing humanitarian supply chains by incorporating actual post-disaster decisions. *European Journal of Operational Research* 265: 1064-1077.

Ozsen L. Coullard C. R. and Daskin M. S. (2010) Capacitated warehouse location model with risk pooling. *Naval Research Logistics* 55(4): 295−312.

Punyim P., Karoonsoontawong A., Unnikrishnan A. and Xie C. (2018) Tabu search heuristic for joint location-inventory problem with stochastic inventory capacity and practicality constraints. *Networks & Spatial Economics* 18(2): 1-34.

Rezaei S. and Kheirkhah A. (2017) A comprehensive approach in designing a sustainable closed-loop supply chain network using cross-docking operations. *Computational & Mathematical Organization Theory* (4): 1−48.

Savaskan R. C., Bhattacharya S. and Wassenhove L. N. V. (2004) Closed-loop supply chain models with product remanufacturing. *Management Science* 50(2): 239−252.

Schrijver, A. (2003) Combinatorial optimization: polyhedra and efficiency. *Springer*, Berlin, Germany.

Senthil S., Murugananthan K. and Ramesh A. (2018) Analysis and prioritisation of risks in a reverse logistics network using hybrid multi-criteria decision making methods. *Journal of Cleaner Production* 179: 716−730.

Shen Z. J. M., Coullard C. and Daskin M. S. (2003) A joint location-inventory model. *Transportation Science* 37(1): 40−55.

Srivastava S. K. (2007) Green supply-chain management: A state-of-the-art literature review. *International Journal of Management Reviews* 9(1): 53−80.

Taleizadeh A. A., Sane-Zerang E. and Choi T. M. (2018) The effect of marketing effort on dual-channel closed-loop supply chain systems. *IEEE Transactions on Systems Man & Cybernetics Systems* 48(2): 265−276.

Üster H. and Hwang S. O. (2016) Closed-loop supply chain network design under demand and return uncertainty. *Transportation Science* 51(4): 1063−1085.

Vivaldini K., Rocha L. F., Martarelli N. J., Becker M. and Moreira A. P. (2016) Integrated tasks assignment and routing for the estimation of the optimal number of AGVs. *International Journal of Advanced Manufacturing Technology* 82(1−4): 719−736.

Wu C. H. and Kao Y. J. (2018) Cooperation regarding technology development in a closed-loop supply chain. *European Journal of Operational Research* 267: 523−539.

Xu X., Zeng S. and He Y. (2017) The influence of E-services on customer online purchasing behavior toward remanufactured products. *International Journal of Production Economics* 187: 113−125.

Zhang Z. H., Berenguer G. and Shen Z. J. M. (2015) A capacitated facility location model with bidirectional flows. *Transportation Science* 49(1): 114−129.

Zhen, L., Xu, Z., Wang, K., Ding, Y. (2016) Multi-period yard template planning in container terminals. *Transportation Research Part B* 93: 700-719.

Zhen, L. (2016) Modeling of yard congestion and optimization of yard template in container ports. *Transportation Research Part B* 90: 83-104.