Scientia Iranica E (2020) 27(3), 1634–1654

Sharif University of Technology
Scientia Iranica
Transactions E: Industrial Engineering
http://scientiairanica.sharif.edu

Mathematical modelling of a decentralized multi-echelon supply chain network considering service level under uncertainty

R. Nourifar[a], I. Mahdavi[a,*], N. Mahdavi-Amiri[b], and M.M. Paydar[c]

a. *Department of Industrial Engineering, Mazandaran University of Science and Technology, Babol, P.O. Box 4716685635, Iran.*
b. *Department of Mathematical Sciences, Sharif University of Technology, Tehran, Iran.*
c. *Department of Industrial Engineering, Babol Noshirvani University of Technology, Babol, Iran.*

Received 6 April 2018; received in revised form 1 August 2018; accepted 3 December 2018

KEYWORDS
Decentralized decision making;
Supply chain network;
Uncertainty;
Tri-level programming;
Service level.

Abstract. In this study, a planning problem for multi-time, multi-product, multi-echelon Supply Chains (SCs) is studied and the implications of formulating a tri-level model to integrate Procurement, Production, and Distribution (PPD) while maintaining the existing hierarchy in the decision process are discussed. In our model, there were three different decision makers controlling the PPD processes in the absence of cooperation because of different optimization strategies. First, a hierarchical Tri-Level Programming (TLP) model was developed to deal with decentralized SC problems. Then, an algorithm was formulated to solve the proposed model. A numerical sample was investigated to scrutinize applicability of the optimization model and the proposed algorithm. In order to evaluate the application of the model and the proposed algorithm, 10 sets of small and large problems were randomly generated and tested. The experimental results showed that our proposed fuzzy-stochastic Simulation-based Hierarchical Interactive Particle Swarm Optimization (Sim-HIPSO) performed well in finding good approximate solutions within reasonable computational times.

1. Introduction

Supply Chain (SC) management plays a vital role in entrepreneurial activities and is an integral part of most businesses. Today's fierce competition among markets, introduction of products with new characteristics in short life cycles, the rise in the expectations of customers, and rapid growth of technology have caused business enterprises to seriously invest in their SCs in order to retain and promote their own positions in the marketplace. An SC is a network of organizations working together to convert and move products from the raw material stage to the final customer. These organizations are interconnected through physical, informational, and financial flows. Having an efficient and lean Supply Chain Network (SCN) with effective interconnections between various levels of the chain can lead to a significant increase in profitability, reduction in costs, and improvement of customer satisfaction. Pursuing these objectives without taking into account certain strategic, tactical, and operational decisions could have a negative impact on the performance of an SC.

*. *Corresponding author. Tel.: +98 1132191205*
E-mail address: irajarash@rediffmail.com (I. Mahdavi)

doi: 10.24200/sci.2018.50733.1842

Decision making in an SCN is characterized in two ways: centralized or decentralized. In a centralized SC, decisions are made by a single decision maker at a central location for the entire SC system. The typical objective of a centralized SC is to optimize the total cost/profit of the system. In a decentralized SC, each entity decides on its own effective strategy without considering the impact on other entities of the SC system. This way, centralized decisions lead to global optimization, whereas decentralized decisions lead to local optimization. Almost all SCNs, due to the existence of various enterprises with different objectives which often conflict with each other in various levels of the chain, are not centrally handled by a single decision maker [1]. Hence, in today's competitive markets, the study of SC management problems appears to be beneficial in nearly every business [2]. In this regard, the role of decentralized SC management is even more significant in industries such as cement, glass, etc. As pointed out by Ma et al. [3], poor Procurement-Production-Distribution Planning (PPDP) in these industries can lead to significant losses, especially in large-scale SCs. To the best of our knowledge, in most research works dealing with decentralized SCNs, procurement policy on lot sizing decisions for the raw material is often examined separately from the production planning problem and logistics management because of the complexity of the interplays among these areas. Therefore, the current research aims at simultaneously determining the Procurement, Production, and Distribution (PPD) policies with decentralized decision-making structure. The main features of some recent works in the study of decentralized SCNs are summarized in Table 1.

Table 1. Review of the decentralized Supply Chain Network (SCN) problem.

Row	**Reference**	**# of levels**	**Chain levels**				**Uncertain parameters**				**Types of problem**			**Types of modelling**		**Multi-period**	**Multi-product**	**Solution approach**
			Supplier	**Producer**	**Distributer**	**Customer**	**Supply**	**Process**	**Demand**	**Other**	**LA**	**PD**	**PPD**	**Bi-level**	**Tri-level**			
1	Vallejo et al. [37]	3	*	*	*	*					*			*				Scatter search and Stackelberg equilibrium
2	Ma et al. [3]	3		*	*	*			*	*		*		*			*	Genetic algorithm
3	Calvete et al. [2]	3		*	*	*						*		*				Evolutionary algorithm
4	Calvete et al. [16]	3		*	*	*						*		*				Ant colony algorithm
5	Wang and Lee [12]	3	*	*		*			*		*			*				Ant colony algorithm
6	Guo et al. [10]	2		*	*							*		*				Bi-Level Evolutionary Optimization (BLEO)
7	Xu et al. [18]	4	*	*	*	*						*			*			KKT conditions
8	Jia et al. [40]	3		*	*	*						*		*				Hybrid genetic algorithm
10	Saranwang and Likasiri [26,27]	3		*	*	*					*							Heuristic algorithm
11	Yue and You [28]	2	*	*							*			*			*	Decomposition algorithm
12	Fard and Hajiaghaee [29]	2		*	*						*				*			Metaheuristic method
8	**Present work**	4	*	*	*	*	*	*	*	*			*		*	*	*	A simulation approach to handling uncertain parameters and a tri-level PSO algorithm for solving the problem

*: LA: Location-Allocation; PD: Production-Distribution; PPD: Procurement-Production-Distribution.

An important issue in optimizing decentralized SCNs is uncertainty. Although deterministic SC models are useful, they may not be realistic, because they use the average values of the system parameters while various sources of uncertainty can be identified in these systems. On the other hand, incorporating inherent uncertainties helps companies to provide better levels of service for their customers. In an SC, service level is usually defined as the probability of being able to service the demand of customers without facing any backorder or lost sale. Thus, increasing the inventory level of Distribution Centers (DCs) improves service level, but also increases the risk of unsold products. However, probabilistic features of these costs and the complications associated with considering them in the mathematical model may be the reasons for their absence in the operational costs considered for decentralized network design problems. Here, we attempt to incorporate these costs in the proposed model.

The principal focus of our work is on proposing an appropriate model for the PPD system in a multi-echelon SCN with a decentralized decision structure and multiple uncertainty issues including demand side, process side, and supply side, simultaneously. Service level is considered and effective computational tools are presented for determining solutions to the proposed model. To overcome the described problem, the main contributions of our work as compared to the available approaches can be summarized as follows:

1. Development of a mathematical model for the decentralized PPDP problem with a Tri-Level Programming (TLP) structure;
2. Handling multiple uncertainties including demand-side, process-side, and supply-side to more closely reflect reality;
3. Considering uncertainties with fuzzy and stochastic parameters, simultaneously (available works with uncertain parameters deal with fuzzy random parameters or stochastic problems);
4. Incorporating the responsiveness level and its corresponding cost in the problem;
5. Presenting a novel hybrid PSO approach combined with a simulation-based method for solving large problems.

The remainder of our work is organized as follows. In the next section, literature review is presented to discuss different approaches to SCN design followed by an outline of our contributions. In Section 3, first, some concepts of Multi-Level Programming (MLP) and fuzzy set theoretical applications are given. Then, a description of the PPDP problem along with notation, assumptions, and mathematical formulation of a decentralized SC model is given in Section 4. In Section 5, a hybrid algorithm is proposed for solving the tri-level PPDP model. Numerical experimentations and some results obtained by the proposed algorithm are presented and discussed in Section 6. Finally, conclusions and directions for future research are provided in Section 7.

2. Literature review

An SCN is commonly defined as a set of organizations directly or indirectly involved in satisfying the orders of customers. These organizations include customers, wholesalers/distributors, manufacturers, and suppliers. The network is comprised of all functions involved in fulfilling the requirements and needs of customers. These functions consist in PPD. Most SC models are merely concerned with the competitive market in the sales process (demand side), while the chains often face competitive markets in the procurement process. They are provided with the raw materials needed by the competitive market through multiple suppliers and this upstream competition affects the costs to the chain. Thus, in order to improve the income of the chain, this upstream competition (supply side) is very important [1]. In the literature, in most research works dealing with SCNs, procurement policy on lot sizing decisions for raw materials is often examined separately from the production planning problem and inventory and logistics management because of the complexity of the interplays among these areas [4–9]. However, PPDP in a perfect SCN that includes several suppliers, producers, distributors, and customers is one of the most important optimization problems in the design of SCNs and has rarely been addressed in the literature.

A decentralized approach to SCN is subject to various economic factors at different levels of the chain and effectiveness of the decisions is quite important. Decision makers at every level control a set of variables and the objective function of the level, which may be in contrast with the optimal solution for the whole system. Every entity in the SC could also make its own decisions based on its goals and objectives, potentially causing distributed and decentralized decision making in the SC. The important and fundamental challenge is appropriately modeling the relationships among the entities of the SC. The optimization problem needs to be formulated as a multi-level (bi-level or tri-level) mathematical programming problem when two or more decision makers in a hierarchical structure are being considered [10].

Study of decentralized SCN design has attracted much attention in the past decade [11–14,15]. Most researchers so far have focused on modeling Production-Distribution (PD) problems for a decentralized SCN design. For instance, Calvete et al. [16] considered a PD problem in a three-level SCN. The authors presented a bi-level model for the problem wherein the first level decided on the design of the routes serving the clients and the second level was involved with the decision of production planning. They utilized an ant-colony-based approach to solving the problem at the first level using ants to create the routes and a direct optimization method to solve the second-level problem. Camacho-Vallejo et al. [17] studied the problem of planning PD in an SC. The problem was formulated as a bi-level mathematical problem in which the upper level was to decide on the amount of products sent from the DCs to the retailers with the aim of minimizing transportation costs of acquiring the products coming from the plants. On the other hand, the lower level controlled the amount of produced products with the aim of minimizing operational costs of the plant. To solve the problem, they proposed a heuristic algorithm based on Scatter Search incorporating the Stackelberg game. It is noteworthy that their solution approach was a development of the obtained objective function value and the time required by the method of Calvete et al. [16]. Xu et al. [18] presented a tri-level model for the SC problem consisting of supplier, manufacturer, and retailer based on the Conditional Value-at-Risk (CVaR) measure in risk management. In this model, the supplier and the manufacturer, at the top and the middle levels, maximized their own profits while, at the bottom level, the retailer maximized their CVaR of the expected profit. To solve the problem, the proposed model was transformed into a Bi-Level Programming (BLP) one. Calvete et al. [2] investigated a decentralized SCN composed of manufacturing plants, warehouses, and customers. The authors adopted a bi-level model for the PD planning problem. In the bi-level model, the distribution company controlled the opening of depots and how to dispatch products from depots to customers, and the manufacturer company controlled the manufacturing process. A metaheuristic approach based on an evolutionary optimization algorithm was presented to solve the problem. An integrated PD problem considering a parallel-machine production environment and a product batch-based delivery was investigated by Guo et al. [10]. A mixed-integer nonlinear BLP problem was formulated in this model and solved by a bi-level evolutionary optimization procedure based on a memetic algorithm and an evolutionary strategy.

Another important challenge in designing an SCN is how to incorporate different issues of disruption and uncertainty in different SC entities from suppliers to customers due to the complex interactions among them. In the optimization model, ignoring uncertainty results in infeasible or non-optimal solutions in real cases. Therefore, the need to appropriately consider uncertainty and variations in the SC is inevitable for the design and modelling of SCNs from the strategic point of view.

In order to cope with uncertainty, different approaches have been adopted in the literature. The first one is the distribution-based approach in which normal distribution with specified mean and Standard Deviation (SD) is widely invoked for modeling the uncertain parameters. The second one is fuzzy programming in which the forecasting parameters are considered to be fuzzy numbers with accompanied membership functions. The third approach is stochastic programming in which uncertainty is directly based on scenarios. In these scenarios, several discrete scenarios with associated probability levels are used to describe expected occurrences of particular outcomes. Finally, the last approach is the chance-constraint-based one in which each uncertain parameter is treated as a random variable with a given probability distribution.

Some studies have incorporated uncertainty in SC models [19–24,25]. However, inclusion of uncertainties in decentralized SCNs has rarely been considered in the literature. Wang and Lee [12] addressed a location-allocation problem for two-level and three-level SCs considering risky demands. A bi-level stochastic programming model was developed to maximize the profit. The problem was solved by an improved algorithm based on ant colony optimization. Ma et al. [3] developed a BLP model for the integrated PD planning problem in SCNs considering contradiction and coordination under uncertainty. In this model, the core firm at the upper level of the hierarchy controlled the opening of plants and warehouses for serving the customers to minimize the total cost. At the lower level, the production sub-problem was aimed at minimizing manufacturing and transportation costs and the distribution sub-problem at minimizing the inventory and distribution costs to make decisions based on core decisions of the firm. In their model, customer demands and transportation costs were considered to be fuzzy variables. They utilized an improved Genetic Algorithm (GA) with a fuzzy logic controller in order to control the GA parameters during the genetic search process. Saranwong and Likasiri [26,27] modeled a distribution network problem using a BLP approach and applied it to a case study in Thailand. The authors proposed five heuristic algorithms and an exact approach using the CPLEX software package. Yue and You [28] presented an SC design and operation problem using a BLP framework. They developed a reformulation and decomposition algorithm for decoding the problem. Fard and Hajaghaei-Keshteli [29] studied the location-allocation problem in an SC considering the forward and reverse networks, simultaneously. They formulated the problem as a TLP problem and proposed five metaheuristic algorithms to solve it.

In accordance with the above considerations, here, we propose a multi-period, multi-product PPDP model in a decentralized SC with multiple uncertainty parameters considering service level and responsiveness. A forward SCN encompasses companies involved with various stages of the chain consisting of suppliers, manufacturers, DCs, and consumer zones. There are uncertainties involved in all the three levels of PPDP of the SC network. The uncertainties corresponding to imprecise parameters on the demand side, process side, and supply side are handled with stochastic and fuzzy numbers. In order to elevate the responsiveness level of the chain, two major factors consisting of shortage and delay costs are considered in the model.

3. Preliminaries

3.1. Multi-level optimization

Multi-level linear programming models have been used to address a variety of real-world problems in which a hierarchical and competitive structure of decision making prevails.

In hierarchical decision making, the first level is known as the leader and the second level is called the follower [30]. Control variables divide decisions between different levels of decision-making and the individual objective functions are optimized [31]. Bi-Level Linear Programming (BLLP) is applied to multi-level decision problems when there are only two levels of decision. A general BLLP problem can be stated as follows [32]:

$$\begin{aligned}&\min_{x\in X}\ F_1(x,y)\\&\qquad \text{s.t.}\quad G_1(x,y)\le 0\\&\qquad\qquad \min_{y\in Y}\ F_2(x,y)\\&\qquad\qquad\qquad \text{s.t.}\quad G_2(x,y)\le 0\\&\qquad\qquad\qquad\qquad x,y\ge 0,\end{aligned}\tag{1}$$

where $x\in R^p$ and $y\in R^q$ are the decision variables of the upper level and the lower level, respectively. $F_i: R^p\times R^q\rightarrow R$, $i=1,2$, are the objective functions and $G_i: R^p\times R^q\rightarrow R$, $i=1,2$, are the constraints of the upper level and lower level, respectively.

By extending BLP, MLP, e.g. TLP, is introduced in which the second level is itself a bi-level program. To describe a tri-level decision problem, a basic Linear Tri-Level Programming (LTLP) model can be stated as follows:

$$\begin{aligned}&\min_{x\in X} F_1(x,y,z)\\&\qquad \text{s.t.}\quad G_1(x,y,z)\le 0\\&\qquad \min_{y\in Y} F_2(x,y,z)\\&\qquad\qquad \text{s.t.}\quad G_2(x,y,z)\le 0\\&\qquad\qquad \min_{z\in Z} F_3(x,y,z)\\&\qquad\qquad\qquad \text{s.t.}\quad G_3(x,y,z)\le 0\\&\qquad\qquad\qquad\qquad x,y,z\ge 0,\end{aligned}\tag{2}$$

where $x\in R^p$, $y\in R^q$, and $z\in R^r$ are the decision variables; $F_i: R^p\times R^q\times R^r\rightarrow R$, $i=1,2,3$, are the objective functions; and $G_i: R^p\times R^q\times R^r\rightarrow R$, $i=1,2,3$, are the constraints of the top level, middle level, and bottom level, respectively.

Considering its conditions, the tri-level problem is non-convex, due to the involvement of a constraint zone specified by another optimization problem, and strongly NP-hard even when all the functions are linear. Several heuristic methods have been proposed for solving MLP problems [30,33,34].

3.2. Fuzzy set theory

Here, we state some necessary results of the fuzzy set theory. We refer to [35,36] for details.

Definition 1. A fuzzy set $\tilde{a}$ of a universe X is a set of ordered pairs $(x,\mu_{\tilde{a}}(x))$, $\mu_{\tilde{a}}(x)$, called the membership functions, which is a real number in the interval [0, 1] and associated with each element x in X. The classical membership degrees are represented by 1 (a member) and 0 (not a member).

Definition 2. A fuzzy variable $\tilde{x}$ is said to be convex if $\mu_{\tilde{a}}(\lambda x_1+(1-\lambda)x_2)\ge\min\{\mu_{\tilde{a}}(x_1),\mu_{\tilde{a}}(x_2)\}$, $\forall\, x_1,x_2\in X,\lambda\in[0,1]$.

Definition 3. A fuzzy number $\tilde{a}$ is a convex subset of the real line satisfying the following conditions:

(a) $\mu_{\tilde{a}}(x)$ is piecewise continuous;

(b) $\mu_{\tilde{a}}(x)$ is normalized, that is, there exists $m\in\Re$ with $\mu_{\tilde{a}}(m)=1$, where m is called the mean value of $\tilde{a}$.

Definition 4. $\alpha-cut$, $\tilde{A}_\alpha$, and strong $\alpha-cut$, $\tilde{A}_{\alpha+}$, of the fuzzy set $\tilde{A}$ in the universe of discourse X are respectively defined by:

$$\tilde{A}_\alpha=\{x|\mu_{\tilde{A}}(x)\ge\alpha,\ x\in X\},\quad \text{where}\quad \alpha\in[0,1],\tag{3}$$

$$\tilde{A}_{\alpha+}=\{x|\mu_{\tilde{A}}(x)>\alpha,\ x\in X\},\quad \text{where}\quad \alpha\in[0,1].\tag{4}$$

The lower and upper points of any $\alpha - cut$, $\tilde{A}_\alpha$, are represented by $\inf \tilde{A}_\alpha$ and $\sup \tilde{A}_\alpha$, respectively, and we assume that both are finite. For convenience, we denote $\inf \tilde{A}_\alpha$ by A_α^- and $\sup \tilde{A}_\alpha$ by A_α^+ (see Figure 1).

Definition 5. Let $\tilde{a}$ be a fuzzy number with membership function $\mu_{\tilde{a}}(x)$. Then, for $\xi > r$, the possibility, necessity, and credibility measures are defined as follows:

Possibility measure:

$$Poss\,(\xi \geq r) = sup(\mu(x)). \tag{5}$$

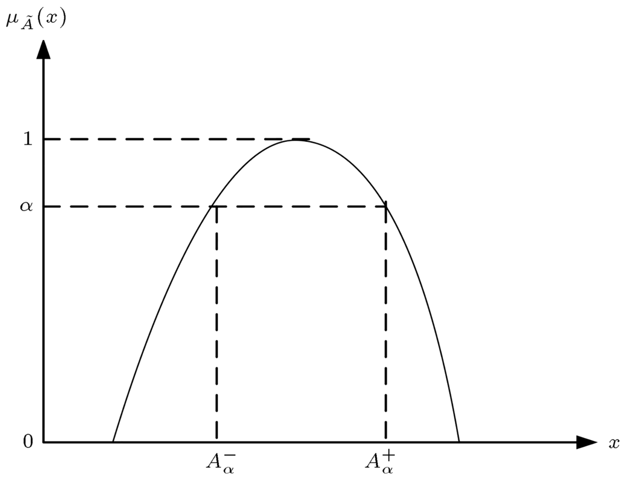


Figure 1. An example of an $\alpha - cut$.

Necessity measure:

$$Nec\,(\xi \geq r) = 1 - Pos\,(\xi < r) = 1 - sup(\mu(x)). \tag{6}$$

Credibility measure:

$$Cr\,(\xi \geq r) = \frac{1}{2}\left(Pos\,(\xi \geq r) + Nec\,(\xi \geq r)\right). \tag{7}$$

4. Problem description

Here, we consider the development of a model for a decentralized SCN in the market. The chain is composed of several customer groups, distributers, manufacturers, and suppliers (see Figure 2).

Customers are at the first level; at the second level, there are storage centers which transport final products to the end customers and at the third level, there are manufacturers (producers) which direct the end products to storage centers. The unsatisfied demand of each customer in each time period is assumed to be backordered; however, the entire unsatisfied demands of customers must eventually be satisfied. Finally, at the lowest level, there are suppliers providing raw materials to the factories for production. The main assumptions for this problem are:

1. The decision making approach within the hierarchical or the tri-level structure is decentralized;
2. The network corresponds to a multi-product, multi-echelon, and multi-time SC problem involving several suppliers, manufacturers, potential DCs, and end customers;

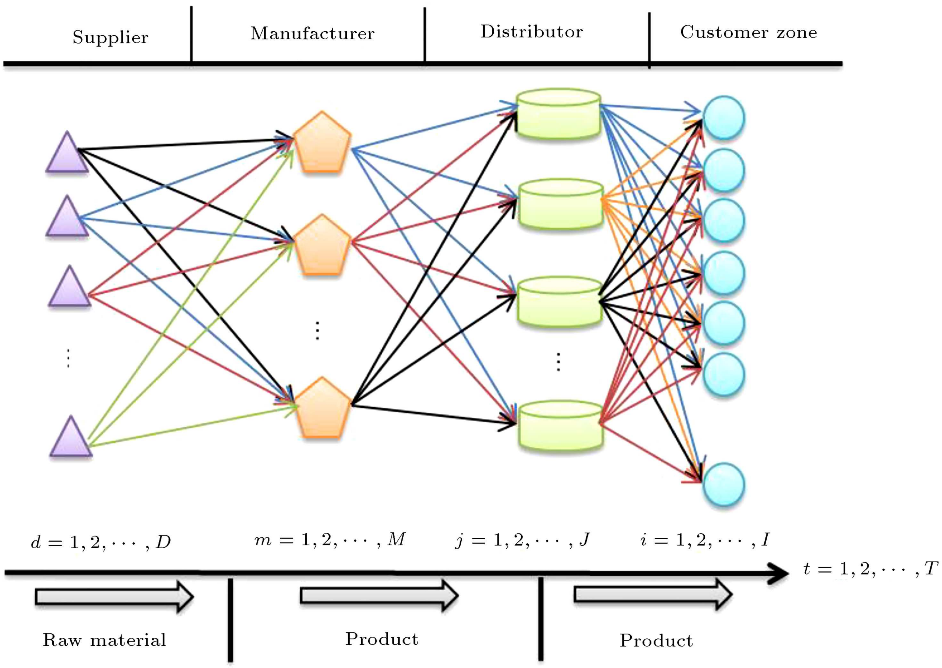


Figure 2. Architecture of a four-echelon Supply Chain Network (SCN).

3. The precise demand and material supply at the DC and supplier levels are not known; thus, demand and supply parameters are considered to be stochastic with known distribution functions;
4. There are several products in the chain and each customer zone can be supplied by any DC;
5. The numbers and locations of suppliers, manufacturers, and customers are known.

To formulate the mathematical model, the following notation is used.

Indices:

D Number of suppliers ($d = 1, \cdots, D$)

M Number of manufacturing plants ($m = 1, \cdots, M$)

J Number of potential warehouse DCs ($j = 1, \cdots, J$)

I Number of customers ($i = 1, \cdots, I$)

K Number of end products ($k = 1, \cdots, K$)

L Number of raw materials ($l = 1, \cdots, L$)

T Number of time periods ($t = 1, \cdots, T$)

Parameters:

pc^k_{mt} Cost of production per unit of end product k by plant m in time period t

pcr^l_{dt} Cost of manufacturing per unit of raw material l by supplier d in time period t

hp^k_{mt} Cost of holding per unit of end product k in plant m in period t

hd^k_{jt} Holding cost per unit of end product k in DC j in period t

hs^l_{dt} Holding cost per unit of raw material l in supplier d in period t

tp^k_{mjt} Transportation cost per unit of end product k from plant m to DC j in period t

td^k_{jit} Transportation cost per unit of end product k from DC j to customer i in period t

ts^l_{dmt} Transportation cost per unit of raw material l from suppliers d to plant m in period t

scp^k_{mt} Set-up cost per unit of end product k by plant m in period t

scs^l_{dt} Set-up cost of production per unit of raw material l by supplier d in period t

pp^k_{jmt} Price per unit of product k supplied by plant m to DC j in period t

$\widetilde{pr}^l_{dmt}$ Price per unit of raw material l supplied by supplier d to plant m in time period t, which is a fuzzy number

bc^k_{it} Cost per unit of backorder of product k to supply the demand of customer i in time period t

dc^k_{it} Delay cost of shipping per unit of product k for the demand of customer i in time period t

$\widehat{S}^l_{dt}$ Capacity of production of supplier d for raw material l in time period t, which is a stochastic number

$\widehat{A}^k_{mt}$ Capacity of production of plant m for end product k in time period t, which is a stochastic number

$\widehat{D}^k_{it}$ Demand value of customer i for end product k in time period t, which is a stochastic number

f_j Fixed cost for establishing DC j

$\widehat{pt}^k_{mt}$ Processing time required by production plant m to produce one unit of product k in time period t, which is a stochastic number

$\widehat{st}^k_{mt}$ Set-up time of producing end product k by manufacturing plant m in time period t, which is a stochastic number

$\widehat{tt}_{mt}$ Total time for production plant m to produce products in time period t, which is a stochastic number

$\widetilde{lt}_{jit}$ Delay time of shipping for demand of customer i from DC j in time period t, which is a fuzzy number

WP_m Total capacity for manufacturing plant m to store products (m^3)

WR_m Total capacity for manufacturing plant m to store raw materials (m^3)

W_j Total capacity for DC j to store products (m^3)

R^k_m Total transportation capacity of manufacturing plant m to deliver product k in period t

v_k Volume of a unit of end product k (m^3)

β_{lk} Usage rate of raw material l to manufacture end product k

M A positive large number

Decision variables:

G^k_{mt} Binary variable, which is 1 if product k is produced by plant m in period t, and 0 otherwise

Y_j	Binary variable, which is 1 if DC j is open and 0 otherwise
X_d	Binary variable, which is 1 if supplier d produces raw material and 0 otherwise
ℓ^k_{jit}	Binary variable, which is 1 if DC j ships product k to customer i and 0 otherwise
QP^k_{mt}	Quantity of product k produced by plant m in period t
QR^l_{dt}	Quantity of raw material l produced by supplier d in period t
U^k_{jmt}	Quantity of end product k transported by manufacturing plant m to warehouse j in period t
N^k_{ijt}	Quantity of end product k dispatched by DC j to customer i in period t
P^l_{dmt}	Quantity of raw material l shipped from supplier d to plant m in period t
B^k_{it}	Backorder quantity of product k for demand of customer i in period t
IP^k_{mt}	Inventory of end product k in plant m at the end of period t
IR^l_{mt}	Inventory of raw material l in plant m at the end of period t
Id^k_{jt}	Inventory of product k in DC j at the end of period t
IS^l_{dt}	Inventory of raw material l in supplier d at the end of period t.

4.1. Decentralized Procurement-Production-Distribution Planning (PPDP) model

For modeling the PPDP problem with a decentralized structure, we will introduce a TLP model considering the optimal decisions of the supplier, the manufacturer, and the distributer in a four-echelon SC. The top level (distribution problem) as the leader in the hierarchical structure controls which DCs should be applied and specifies the amounts of customer orders provided by the DCs aiming to minimize the total fixed costs and operating costs of the opened DCs associated with inventory and transportation. The unmet demand of each customer for any product in each period will be considered as backordered, although the demands of all customers must be met eventually.

Sufficient supplies of product must be available at the DCs in order to meet the demands of the customers. After decisions of the leader (top level), the middle level (production problem), as the follower, aims for cost reduction by minimizing production cost, transportation cost, and inventory cost from the plant to the DCs. Ultimately, at the bottom level, procurement problem is to decide which suppliers should provide raw materials to the manufacturers. Feedback regarding the production, transportation, and inventory decisions by the production problem and procurement problem are sent to the leader and the leader may modify the decisions for cost minimization. Until an optimal solution is found, this process continues. Therefore, through the interactivity of all involved SC members, an optimal decision is found. TLP can be utilized to demonstrate the interaction between the leader and the follower as shown by the tri-level structure of the complex real-life decentralized PPDP model in Figure 3. The tri-level optimization PPDP problem is formulated as follows:

$$\begin{aligned}\min\ F_{dis} = {} & \sum_{j=1}^{J} f_j.Y_j + \sum_{t=1}^{T}\sum_{m=1}^{M}\sum_{k=1}^{K}\sum_{j=1}^{J} \widetilde{pp}^k_{jmt}.U^k_{jmt} \\ & + \sum_{j=1}^{J}\sum_{t=1}^{T}\sum_{k=1}^{K} hd^k_{jt}.Id^k_{jt} \\ & + \sum_{t=1}^{T}\sum_{i=1}^{I}\sum_{k=1}^{K}\sum_{j=1}^{J} td^k_{ijt}.N^k_{ijt} \\ & + \sum_{i=1}^{I}\sum_{t=1}^{T}\sum_{k=1}^{K} bc^k_{it}.B^k_{it} \\ & + \sum_{i=1}^{I}\sum_{t=1}^{T}\sum_{k=1}^{K}\sum_{j=1}^{J} \max\Big(0, \widetilde{lt}_{jit}\Big).dc^k_{jit}.\ell^k_{jit}, \end{aligned} \quad (8)$$

s.t.:

$$\sum_{k=1}^{K} v_k.Id^k_{jt} \le W_j.Y_j, \qquad \forall\ j,t, \quad (9)$$

$$\sum_{j=1}^{J} U^k_{jmt} \le R^k_m.G^k_{mt}, \qquad \forall\ k,t,m, \quad (10)$$

$$Id^k_{j,t} = Id^k_{j,t-1} + \sum_{m=1}^{M} U^k_{jmt} - \sum_{i=1}^{I} N^k_{ijt}, \qquad \forall\ j,k,t, \quad (11)$$

$$B^k_{j,t} = B^k_{j,t-1} + \widehat{D}^k_{it} - \sum_{j=1}^{J} N^k_{ijt}, \qquad \forall\ i,k,t, \quad (12)$$

$$\begin{aligned}\min\ F_{man} = {} & \sum_{d=1}^{D}\sum_{m=1}^{M}\sum_{t=1}^{T}\sum_{l=1}^{L} \widetilde{pr}^l_{dmt}.P^l_{dmt} \\ & + \sum_{m=1}^{M}\sum_{t=1}^{T}\sum_{k=1}^{K} scp^k_{mt}.G^k_{mt}\end{aligned}$$

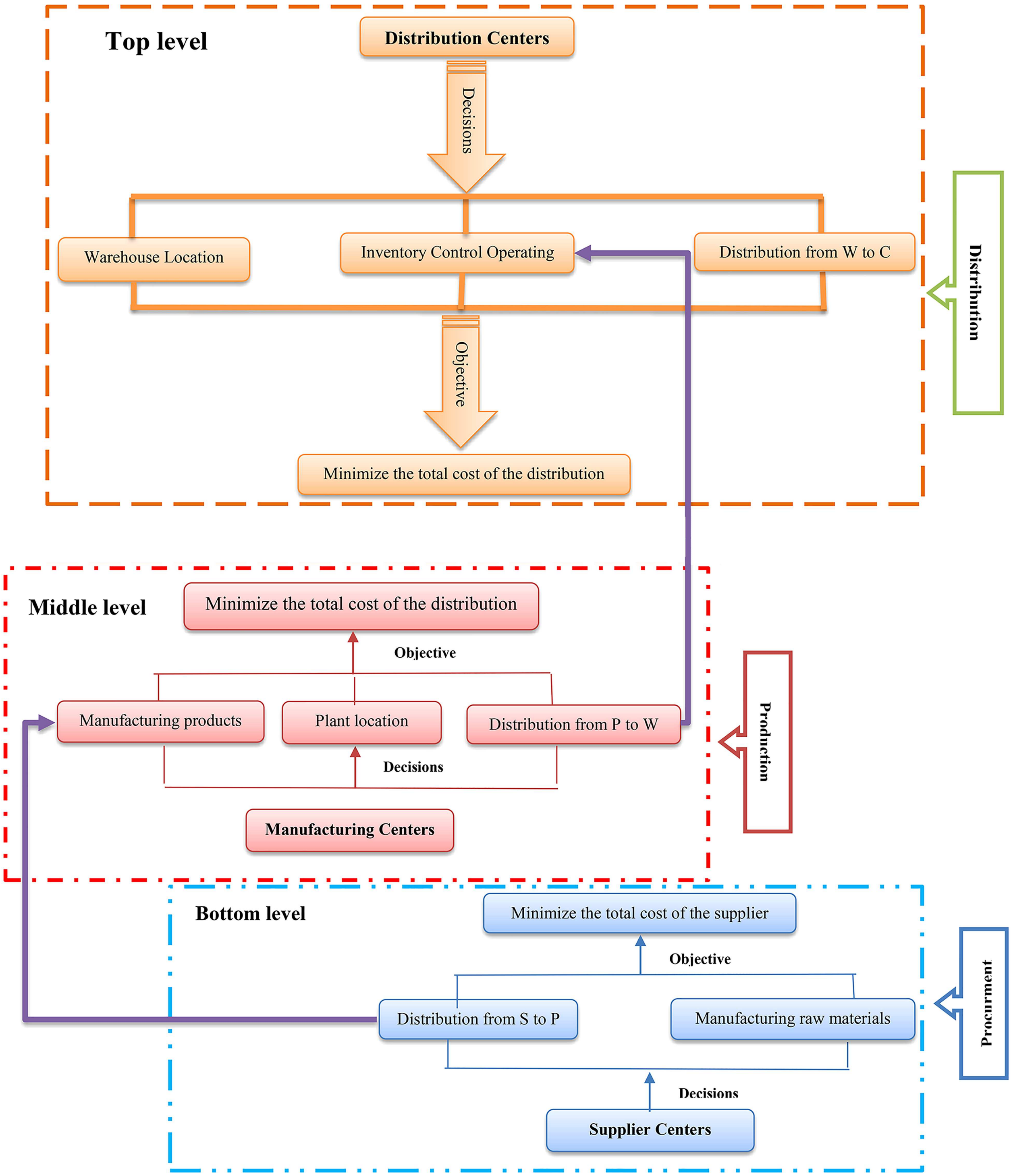


Figure 3. Decentralized Procurement-Production-Distribution Planning (PPDP) model based on a tri-level structure.

$$+\sum_{j=1}^{J}\sum_{m=1}^{M}\sum_{t=1}^{T}\sum_{k=1}^{K} pc^k_{mt}.QP^k_{mt}+\sum_{t=1}^{T}\sum_{m=1}^{M}\sum_{k=1}^{K}$$

$$\sum_{j=1}^{J} tp^k_{jmt}.U^k_{jmt}+\sum_{m=1}^{M}\sum_{t=1}^{T}\sum_{k=1}^{K} hp^k_{mt}.IP^k_{mt}, \quad (13)$$

s.t.:

$$\sum_{k=1}^{K}\sum_{m=1}^{M} v_k.U_{jmt}\leq W_j.Y_j, \qquad \forall\, j,t, \quad (14)$$

$$\sum_{k=1}^{K} \widehat{pt}^{k}_{mt}.QP^{k}_{mt}+\sum_{k=1}^{K} \widehat{st}^{k}_{m}.G^{k}_{mt} \leq \widehat{tt}_{mt}, \qquad \forall\, m,t, \quad (15)$$

$$\sum_{k=1}^{K} v_k.QP^{k}_{mt} \leq WP_m, \qquad \forall\, m,t, \quad (16)$$

$$\sum_{k=1}^{K} QP^{k}_{mt} \leq \widehat{A}^{k}_{mt}, \qquad \forall\, d,t, \quad (17)$$

$$\sum_{k=1}^{K} v_k.IP^{k}_{mt} \leq WP_m, \qquad \forall\, m,t, \quad (18)$$

$$\sum_{k=1}^{K} U^{k}_{jmt} \leq \sum_{k=1}^{K} N^{k}_{ijt}, \qquad \forall\, j,m,t, \quad (19)$$

$$IP^{k}_{m,t}=IP^{k}_{m,t-1}+QP^{k}_{mt}+\sum_{j=1}^{J} U^{k}_{jmt}, \qquad \forall\, m,k,t, \quad (20)$$

$$\min\ F_{sup}=\sum_{d=1}^{D}\sum_{l=1}^{L}\sum_{t=1}^{T} mcr^{l}_{dt}.QR^{l}_{dt} + scs^{l}_{dt}.X^{l}_{dt} + hs^{l}_{dt}.IS^{l}_{dt} + \sum_{d=1}^{D}\sum_{m=1}^{M}\sum_{t=1}^{T}\sum_{l=1}^{L} ts^{l}_{dmt}.P^{l}_{dmt}, \quad (21)$$

s.t.:

$$\sum_{l=1}^{L} QR^{l}_{dt} \leq \widehat{S}^{l}_{dt}, \qquad \forall\, d,t, \quad (22)$$

$$\sum_{m=1}^{M}\sum_{l=1}^{L} P^{l}_{dmt} \leq QR^{l}_{dt}, \qquad \forall\, d,t, \quad (23)$$

$$\sum_{l=1}^{L} QR^{l}_{dt} \leq M.X_{dt}, \qquad \forall\, d,t, \quad (24)$$

$$IS^{l}_{d,t-1}+QR^{l}_{d,t}=\sum_{m=1}^{M} P^{l}_{dmt}+IS^{l}_{d,t}, \qquad \forall\, d,l,t, \quad (25)$$

$$IR^{l}_{m,t-1}+\sum_{d=1}^{D} P_{dmt}-\sum_{k=1}^{K} QP^{k}_{mt}\cdot\beta_{lk}=IR^{l}_{mt}, \qquad \forall\, m,l,t, \quad (26)$$

$$X_{dt}, Y_{jt}, Z^{k}_{mt} \in \{0,1\}, \qquad \forall\, d,m,j,k,t \quad (27)$$

$$QP^{k}_{mt}, IP^{k}_{mt}, IR^{l}_{mt}, A^{k}_{mt}, S^{l}_{it}, N^{k}_{ijt}, U^{k}_{jmt}, I^{k}_{jt}, IS^{l}_{dt} \geq 0, \qquad \forall\, d,m,j,k,t. \quad (28)$$

It should be noted that parameters with a tilde (˜) and a hat (ˆ) at the top are respectively signified as fuzzy and stochastic parameters. The first objective function shown in Eq. (8) corresponds to the minimization of the sum of the costs to DCs and customers, including fixed costs of establishing DCs and purchasing products, holding costs of products, transportation costs from DCs to customers, inventory shortage costs to customers, and costs of delay in shipping orders to customers.

The control of potential warehouse capacities is considered in Constraint (9). The total quantity shipped from each plant is controlled by Constraint (10). Constraint (11) corresponds to the balance of end inventories for potential DCs. Similarly, balance equations for the shortages in the demands of customers are considered by Constraint (12).

The second objective function, as given by Eq. (13), is to minimize the costs to producers, including procurement (buying) of raw materials, set-up cost of production, manufacturing of the products, transportation costs from plants to DCs, and inventory holding costs of products. Constraint (14) shows that the volumes of the products transported to potential DCs correspond to their total storage capacities. Constraint (15) ensures that the sum of the required times to manufacture the products is not more than the total time for the production plant. According to Constraint (16), production volume is less than the total storage capacity of the plants. Constraint (17) determines that the volume of finished products produced by plants should be less than the total production capacity of the plants. The final product inventory based on the total storage capacity of the plants is controlled by Constraint (18). Based on Constraint (19), the total quantity shipped from any plant is not more than its capacity. The remaining inventory in each plant at the end of each period is controlled by Constraint (20). The third objective function in Eq. (21) is to minimize the total operational costs of suppliers, including production, set-up, and transportation costs from suppliers to the plants. Constraints (22) and (23) respectively specify that volumes of the raw material produced and shipped by suppliers are less than the total production capacity of the suppliers. Constraint (24) determines the supplier manufacturing raw materials in each time period. Constraints (25) and (26) respectively give the inventory balances of the raw materials for suppliers and plants. Types of the variables are defined by Eqs. (27) and (28).

4.2. Hybrid uncertainty of model

The parameters in the deterministic models of SC problems are known and deterministic. In these models, determining the features of activities extending from the suppliers to the customers precisely is very difficult, if not impossible. In general, deterministic assumptions for SC models are unrealistic. In reality, both tangible and intangible information is usually available to decision makers for decision criteria and constraints. Model parameters and environmental coefficients have hybrid uncertainties due to some inherent variability and/or unavailability of information over the planning horizon and they are frequently imprecise. With the SC containing several layers and several parts in each layer, uncertainty may occur in different parts of PPD because of the dynamic and turbulent nature of the SC. Table 2 defines different parameters that are considered uncertain according to the identified three uncertainty types in an SC tactical planning problem.

Price and delay time uncertainties are different from demand uncertainty in SC, as the price of material or product and delay time cannot be predicted more accurately than the demand uncertainty due to usual fluctuations in large ranges. Thereupon, they are inherently fuzzy data rather than crisp values. Hence, for modeling uncertainty in the price of a material, a possibility theory is more suited than a probability theory.

Most business enterprises for forecasting demands focus on collecting a large data set [37]. Therefore, demand uncertainties are often modeled by probabilistic distributions. Also, processing time, production capacity, and supplier capacity are uncertain. Therefore, in order to engage these hybrid uncertainties of ill-known parameters in the mathematical model, we develop an integrated tri-level model to accommodate both fuzzy and stochastic data, which we call a Fuzzy-Stochastic Tri-Level Programming (FSTLP) model. For this purpose, fuzzy set theory is used to handle fuzzy data and utility theory is employed to treat stochastic data.

5. Solution methodology

The intrinsic complexity of a multi-level optimization model due to the mutual relations among various layers makes the problem one of the most challenging and most difficult problems to solve in real cases. There are many approaches to tackling complicated Multi-level Optimization Problems (MOPs) [34]. They can be categorized into four main approaches:

1. Nested sequential approach;
2. Single-level transformation approach;
3. Multi-objective approach;
4. Co-evolutionary approach.

In the first approach, the optimization problem at the lower level is solved in a sequential and nested manner assessing the solutions produced at the upper levels of MOP. The second approach reformulates MOP into a single-level structure by a penalty function, a marginal function, and Karush-Kuhn-Tucker (KKT) conditions. The single-level problem can be solved by the available methods. In the third approach, after MOP is converted into a multi-objective optimization problem, a multi-objective metaheuristic approach is utilized to solve it. In the fourth approach, which is a well-known methodology to solve MOPs, at each level, an evolutionary-based optimization approach (e.g., GA, particle swarm, simulated annealing, etc.) is directly employed to find an (approximate) optimal solution. Each level tries to maintain and improve its own solutions separately. In general, the levels can evolve in parallel and cooperate by exchanging information.

A multi-period PPD planning problem in a decentralized SC formulated as an MOP is naturally complex. Only small problems can be solved to optimality in a reasonable computing time and medium and large problems are difficult to solve optimally. This fact has led us to develop a heuristic algorithm for finding approximate optimal solutions. Among the methods described above, a co-evolutionary approach, due to its capability and efficiency in providing satisfactory solutions in reasonable times, is employed to develop a novel algorithm entitled Hierarchical Interactive Parti-

Table 2. Uncertain parameters of the model.

Source of uncertainty in Supply Chains (SCs)	Fuzzy and stochastic coefficient	Notation
Demand	Product demand	$\widehat{D}_{it}^{k}$
	Delay time	$\widetilde{lt}_{jit}$
	Price of product	$\widetilde{pp}_{it}^{k}$
Process	Processing time	$\widehat{pt}_{it}^{k}$, $\widehat{st}_{it}^{k}$, $\widehat{tt}_{it}^{k}$
	Production capacity	$\widehat{A}_{it}^{k}$
Supply	Price of raw material	$\widetilde{pr}_{it}^{k}$
	supplier capacity	$\widehat{S}_{dt}^{l}$

cle Swarm Optimization (HIPSO). In the following, the technical steps needed to solve the model are described.

5.1. Review of Particle Swarm Optimization (PSO)

Particle Swarm Optimization (PSO) is a biologically inspired computational search developed in 1995 by Kennedy and Eberhart [38]. PSO simulates the behavior of bird flocking or fish schooling. A number of basic variations have been developed to improve the speed of convergence and quality of solution achieved by PSO.

PSO begins with an initial population of solutions, called initial population, generated randomly. Each single solution is a bird in the search space, called particle, which represents a feasible solution to the problem. A set of particles that are produced in every step of the algorithm is called a swarm. Every particle has a fitness value, which is obtained by the fitness function to be optimized, and a velocity to reach a new position. The particles fly through the problem space by following the current optimal particles. Then, PSO searches for better solutions by updating the generation. Each particle moving in the search space to reach new positions (solutions) is composed of the following three components:

Cognitive component: The best solution that a particle acquires alone (X^i_{pb}) with coefficient c_1;

Social component: The best solution recognized by the entire group (X_{gb}) with coefficient c_2;

Inertia component: The impact velocity of a particle in the step before the current speed is determined by weight w.

After finding the two best values (X^i_{lb}, X_{gb}), the particle updates its velocity and position as follows:

$$V^i_{k+1}=c_1.r_1(X^i_{pb}-X^i_{gb})+c_2.r_2(X_{gb}-X^i_k)+w.V^i_k, \tag{29}$$

$$X^i_{k+1}=V^i_{k+1}+X^i_k, \tag{30}$$

where V^i_k is the velocity vector of particle i in iteration k, w is the inertia weight coefficient, c_1 and c_2 respectively represent the relative influence of the social and cognitive components, X^i_k is position of particle i in iteration k, X^i_{pb} is personal best position and X_{gb} is general best position, and r_1 and r_2 are random numbers uniformly distributed between 0 and 1.

5.2. Hierarchical Interactive Particle Swarm Optimization (HIPSO)

Here, we develop a PSO-based method (denoted as HIPSO) to solve the FSTLP problem. We use a tri-level decision procedure. Our model consists of three hierarchical levels of decision making termed as the leader in the top-level DC, as the follower at the mid-level (manufacturer), and as the sub-follower at the bottom level (supplier). The successive decision making takes place from the top level to the mid-level and then, to the bottom level by decision entities individually with the aim of optimizing the respective objectives. In particular, the leader is of the top priority for decision making. However, its decisions are implicitly influenced by the actions of the followers. Then, the mid-level follower optimizes its own objective function and reacts to the decisions made by the leader with regard to the implicit reactions of the follower of the bottom level. Ultimately, the decision process is repeatedly executed until the Stackelberg equilibrium is met in the three-level vertical structure. In the following, details of the procedure for solving the problem in hand are given:

1. *Generating initial population.* The initial population of particles of size N, representing candidate solutions, is randomly generated. Each particle is shown as an M-dimensional real valued vector, where M is the number of optimized parameters. Initially, we produce the random numbers for the decision variables at the top level. Then, the mid-level and bottom-level problems are solved to obtain the corresponding solutions by a branch and bound approach;
2. *Evaluating fitness value.* The fitness values of the generated particles are obtained by using the top-level objective function. The fitness value of each particle is compared with its *local best* solution. If current value is better than *local best*, then *lbest* is reset as the current value. The global best solution, *gbest*, of the group at every iteration is also identified as the best value obtained so far and its index is saved in the variable g;
3. *Updating particles.* In every iteration k, each particle i tends to move from current position toward a new position in the problem space with velocity V_i^{K+1}. The position and velocity vectors of each particle are updated according to Eqs. (29) and (30);
4. *Termination criterion.* HIPSO algorithm is terminated after a criterion is met, typically after reaching the maximum number of iterations or when sufficiently good fitness values are in hand;
5. *Optimization process of the HIPSO algorithm.* Based on the above discussions, a flowchart of the proposed HIPSO algorithm is given in Figure 4.

5.3. Simulation-based tri-level optimization

According to the formulation of the problem, each of the imprecise parameters in the constraints, namely demand, supply, processing time, or capacity, is represented by a stochastic random number and among

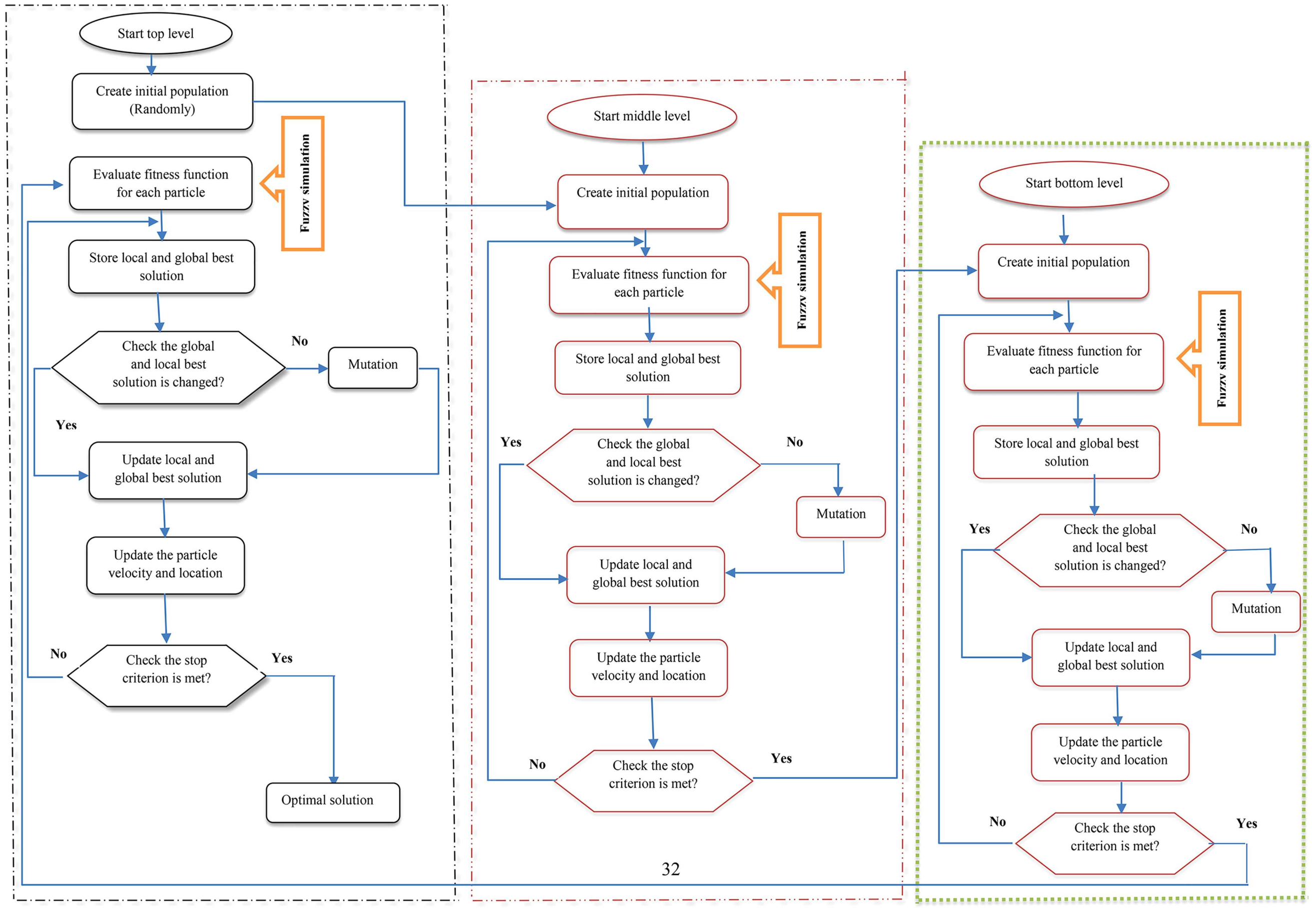


Figure 4. Flowchart of the proposed simulation based Hierarchical Interactive Particle Swarm Optimization (HIPSO) algorithm.

the objective functions, costs and price are represented by fuzzy random numbers. Thus, the proposed model is a fuzzy and stochastic programming one. Chance-Constrained Programming Technique (CCPT), presented by Liu [39], is a stochastic programming approach. In this approach, the main idea is to optimize the critical value of a stochastic or fuzzy constraint at a predetermined confidence level. Here, we use this idea to tackle uncertainty in uncertain functions. Our chance-constraint programming model is defined as follows:

$$\begin{cases} \min\limits_{x} \quad \bar{f} \\ \text{s.t.:} \\ \qquad Cr\left\{C(x,\xi) \leq f\right\} \geq \beta_1 \\ \qquad Pr\left\{g_k(x,\partial) \leq 0\right\} \geq \beta_i, \\ \qquad\quad i = 2,3,\cdots,n, \quad k = 1,2,\cdots,P, \end{cases} \tag{31}$$

where $\beta_i \in [0,1]$, $i = 1,\cdots,n$, represents the acceptable confidence levels, x denotes the decision variables of the model, ξ is a fuzzy random variable, ∂ denotes the stochastic variable, $Cr\{\cdot\}$ is the credibility measure of a fuzzy event, and $Pr\{\cdot\}$ denotes the probability of a stochastic event.

In general, directly solving a mixed fuzzy and stochastic programming tri-level problem is difficult. Monte Carlo simulation can be applied as an effective tool to these problems, to which analytical solutions do not exist or are too complex to obtain. We adopt a Monte Carlo simulation approach to search for an optimal solution within the range of random numbers in order to handle the uncertain parameters of the proposed model. In the following, we present two simulation-based methods using fuzzy random and stochastic programming, called fuzzy random simulation and stochastic simulation. The fuzzy random simulation and stochastic simulation programs for the CCPT model are shown in detail in Algorithms 1 and 2, respectively.

Next, we explain the steps of our algorithm in more details in order to clarify the proposed algorithm. Steps of the simulation-based HIPSO are defined as follows:

Step 1. {Set parameters}

Step 1. Sample ω according to the probability measure Pr from the sample space Ω.

Step 2. Generate $\omega_1, \omega_2, ..., \omega_n$ randomly from the ε-level set of fuzzy vectors ξ, where ε is a sufficiently small number so that $\mu(\omega_j) \geq \varepsilon$.

Step 3. Generate $\beta \in [0,1]$ uniformly.

Step 4. Compute upper bound $\overline{f_{\min}} = \max_{1 \leq j \leq N} \{C(x, \omega_j)\}$ and lower bound $\underline{f_{\min}} = \min_{1 \leq j \leq N} \{C(x, \omega_j)\}$.

Step 5. Compute $f_{\min} = \frac{(\overline{f_{\min}} + \underline{f_{\min}})}{2}$.

Step 6. Compute $Cr(C(x, \xi) \leq f)$.

Step 7. If $Cr(C(x, \xi) \leq f) \geq \beta$, then $f_{\min} = \overline{f_{\min}}$; else, $f_{\min} = \underline{f_{\min}}$.

Step 8. Repeat steps two to seven until $(\overline{f_{\min}} - \underline{f_{\min}}) > \phi$ (with ϕ being a sufficiently small number).

Step 9. Return $f_{\min}$.

Algorithm 1: Fuzzy random simulation for the objective functions.

Step 1. Set $p = 0$ and $p' = 0$.

Step 2. Generate $\beta \in [0,1]$ uniformly.

Step 3. Generate sample $\omega = (\omega_1, \omega_2, ..., \omega_n)^T$ uniformly according to the probability measure Pr from the sample space Ω.

Step 4. If $g_i(x, \partial) \leq 0, \quad i=1,2,...,n,$ then $p = p + 1$; else, $p' = p' + 1$.

Step 5. Let Q be the number of times; we have $g_i(x, \partial) \leq 0, \quad i=1,2,...,n,$ and let $\beta = \frac{Q}{N}$.

Step 6. If $\frac{Q}{N} \leq \beta$, then the constraint $g_i(x, \partial) \leq 0, \quad i=1,2,...,n,$ is feasible; else, it is not feasible.

Algorithm 2: Stochastic simulation for the constraints.

a) Set population size N,

b) Set the maximum velocity $V_{\max}$,

c) Set the inertial weight coefficient w,

d) Set the learning factors C_1 and C_2,

e) Set the random numbers between 0 and 1, $rand()$,

f) Set mutation rate, m_rate.

Step 2. {Initialization}

a) Create the initial population of particles (solutions) and velocities randomly with initial position X_{id} and speed V_{id};

b) Create the local best solution as lbest.

Step 3. Compute the individual fitness value of each particle and store the particle with the best fitness value.

Step 4. Update local position of the best solution and the global best position.

Step 5. Set $i = 1$.

Step 5.1. Fix the top-level variables, solve the mid-level problem, and obtain the solution;

Step 5.2. Fix the top-level and mid-level variables, solve the bottom-level problem, and obtain the solution;

Step 5.3. If the obtained solution is feasible, then update the particle; else, go to the next step;

Step 5.4. Update the local best solution. If $i < N$, then set $i = i + 1$ and go to Step 5.1; else, go to Step 6.

Step 6. Update the global best solution.

Step 7. If the local and global best solutions change, then go to Step 9; else, go to Step 8.

Step 8. Apply the mutation operator of GA and update the particle as follows:

$$X_{k+1}^{new} = X_k^i + rand3 \times N[0,1].$$

Step 9. Update all the particles X_{k+1}^{new} using Eqs. (29) and (30).

Step 10. Compute fitness value for every particle.

Step 11. Update the global best solution.

Step 12. Check the termination criterion. If the number of generations reaches the maximum satisfying value, then stop; else, go to Step 2.

6. Computational results

The presented model is a novel tri-level linear programming one and there is no available test problem for its numerical assessment. Therefore, we considered generating two different sets of test problems for evaluating the performance of the proposed algorithm. First, 10 sets of small problems were generated to assess capability of the algorithm in obtaining the global optimum. The problem was solved by LINGO 14 optimization software package for the global optimal values. Second, 10 sets of large problems were randomly generated in order to consider a widespread range of problems. We implemented our algorithms in MATLAB 7.12 (R2014a) software environment. All computations were run on a Pentium IV 2.8 GHz processor personal computer with 4 GB memory. The ranges of parameter were generated using uniform distributions as shown in Table 3. The population size was 100. The maximum number of iterations was set to 150. Other parameters including C_1, C_2, w, $V_{\max}$, $NPop$, and $NGen$, were respectively set to 2.1, 2.15, 10, and 0.9. These values were set after experimentation with a simple tuning procedure. In the following, we provide the details of the generated test problems and parameter tunings for the algorithm.

6.1. Description of test problems

6.1.1. Small test problems

We apply the proposed algorithm to solving a set of tri-level problems having three units of every agent, that is, there will be three suppliers, three manufacturers, three DCs, and three customer groups for three products and three raw materials in three periods; there are 658 constraints and 1216 decision variables including 33 binary variables. To solve small problems, the parameters involved in the proposed algorithm are chosen as in Table 3. With these parameters, the proposed method is executed 30 independent times on each of the 10 test problems and the average results are presented.

6.1.2. Large test problems

Considering that the resulting model is a tri-level mixed-integer linear programming one, direct optimization algorithm can only solve small instances in a reasonable computing time. All the 10 kinds of small instances can be solved by the LINGO software package in a short time, but direct optimization methods are not efficient in terms of the computing time needed for large problems. Hence, in order to further assess the proposed HIPSO algorithm in terms of performance and capability, 10 different large instances are generated as shown in Table 4 and the average results are presented. We utilize the proposed algorithms to solve the instances using a maximum of 100 iterations.

6.2. Comparison

In this section, we are to show experimental results and compare the proposed algorithm with other algorithms.

Since the proposed problem is NP-hard, a Simulation-based Hierarchical Interactive Particle Swarm Optimization (Sim-HIPSO) algorithm is proposed. We use Optimality Difference Percentage (ODP) as a performance measure for comparing the solutions obtained by the proposed algorithm and the optimal solutions to the tri-level problem:

$$ODP = \left(\left(Z_{alg} - Z_{op}\right)/Z_{op}\right) \times 100, \tag{32}$$

where Z_{alg} is the solution value obtained by the proposed algorithm and Z_{op} is the optimal solution to the tri-level problem found by the LINGO software package. Furthermore, a Relative Difference Percentage (RDP) is computed to compare the solutions obtained by Sim-HIPSO and General PSO (GPSO) methods:

$$RDP = \left(\left(Z_{alg} - Z_G\right)/Z_G\right) \times 100, \tag{33}$$

where Z_G is the solution found by GPSO and Z_{alg} is the solution obtained by the proposed algorithm.

Given the results of the proposed algorithm for small test problems, the proposed algorithm is able to find near optimal solutions in a reasonable computational time. However, for small problems, LINGO computes optimal solutions in a shorter time. The differences between the global optima obtained by LINGO and those obtained by our proposed algorithm are small. Based on the results obtained for small problems, there is an average of only 12% between the global optima of the methods, which indicates acceptable performance of the proposed algorithm. The comparative results obtained for 30 runs of the simulation based HIPSO, the GPSO, and LINGO, including the sample maximum deviation, mean deviation, and SD for the fitness values and ODP with small problems are shown in Table 5. The fitness values of LINGO, GPSO, and simulation based HIPSO in this table reveal that there are very small deviations among the results obtained by the two algorithms and those obtained using LINGO. This proves effectiveness of the proposed algorithm. For large test problems, Table 6 presents the results obtained by 30 runs (i.e., sample size) of the simulation based HIPSO and GPSO, showing the fitness values, the numbers of iterations for convergence, the computation times, and RDP (%). It can be seen that the simulation based HIPSO is able to search for the global optima better than

Table 3. Range of parameters in the numerical study.

Parameter	Range: Small instance	Range: Large instance	Unit	Description
pc_{mt}^{k}	Uniform∼[3,8]	Uniform∼[45,80]	Dollar ($) per unit	Production cost
pcr_{dt}^{l}	Uniform∼[0.2,2]	Uniform∼[15,30]		
$\widetilde{bc}_{it}^{k}$	Uniform∼[10,50]	Uniform∼[10,50]	Dollar ($) per unit	Backorder cost
f_j	Uniform∼[400,900]	Uniform∼[2000,10000]	Dollar ($)	Fixed cost
scp_{mt}^{k}	Uniform∼[10,40]	Uniform∼[200,600]	Dollar ($)	Setup cost
scs_{dt}^{l}	Uniform∼[8,20]	Uniform∼[70,200]		
hp_{mt}^{k}	Uniform∼[5,15]	Uniform∼[4,10]	Dollar ($) per unit	Inventory cost
hd_{jt}^{k}	Uniform∼[1,4]	Uniform∼[10,20]		
hs_{dt}^{l}	Uniform∼[4,10]	Uniform∼[5,15]		
tp_{mjt}^{k}	Uniform∼[10,25]	Uniform∼[90,130]	Dollar ($) per Ton	Transportation cost
td_{jit}^{k}	Uniform∼[2,10]	Uniform∼[50,85]		
ts_{dmt}^{l}	Uniform∼[5,12]	Uniform∼[65,100]		
pp_{jmt}^{k}	Uniform∼[35,100]	Uniform∼[260,450]	Dollar ($) per unit	Price
$\widetilde{pr}_{dmt}^{l}$	Uniform∼[10,20]	Uniform∼[15,30]		
$\widehat{D}_{it}^{k}$	Uniform∼[25,80]	Uniform∼[100,500]	Ton	Demand
$\widehat{pt}_{mt}^{k}$	Uniform∼[0.1,0.95]	Uniform∼[10,15]	Second	Processing time
$\widehat{st}_{mt}^{k}$	Uniform∼[10,30]	Uniform∼[25,40]		
$\widehat{tt}_{mt}$	Uniform∼[150,500]	Uniform∼[3500,4000]		
$\widehat{S}_{dt}^{l}$	Uniform∼[150,300]	Uniform∼[400,1000]	Ton	Capacity
$\widehat{A}_{mt}^{k}$	Uniform∼[100,590]	Uniform∼[5000,10000]	Ton	
WP_m	Uniform∼[80,350]	Uniform∼[1500,2500]	Ton	
WR_m	Uniform∼[50,200]	Uniform∼[100,400]	Ton	
W_j	Uniform∼[400,600]	Uniform∼[2500,3500]	Ton	
R_m^k	Uniform∼[150,250]	Uniform∼[10,20]	Ton	
β_{lk}	Uniform∼[2,4]	Uniform∼[3,7]	—	Bill of material
Lt_{ji}	Uniform∼[1,8]	Uniform∼[1,15]	Day	Delay time
M	Large number	Large number		—

GPSO, but it is to some extent slower. As the interactive evolutionary mechanism in the simulation based HIPSO assists the algorithm to determine better solutions for the tri-level model, more time is needed for convergence.

In order to evaluate and compare the performances of the simulation based HIPSO algorithm and GPSO algorithm, a graphical approach has been used. Figure 5 displays the convergence process of the solution. It is obvious that the proposed algorithm

Table 4. Description of large test problems.

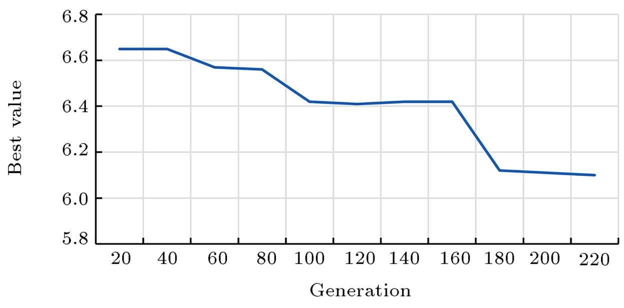
Problem	Indices							Number of variables		Number of constraints
	d	m	j	i	k	l	t	Total	Binary	
L1	3	4	5	3	3	2	3	1651	35	947
L2	4	5	4	5	3	2	4	2689	56	1465
L3	4	5	6	8	4	3	5	6586	110	3623
L4	5	6	6	9	5	4	6	12086	191	6350
L5	5	6	7	10	5	5	7	18301	193	9654
L6	5	8	8	12	4	5	7	19491	209	9662
L7	6	7	8	15	5	4	8	26319	259	13791
L8	7	8	8	12	6	6	8	24323	399	16775
L9	6	9	9	15	5	5	9	36317	420	19526
L10	8	6	15	20	8	5	10	90331	311	46683

Table 5. Obtained results by LINGO, General Particle Swarm Optimization (GPSO) and simulation based Hierarchical Interactive Particle Swarm Optimization (HIPSO) for small test problems.

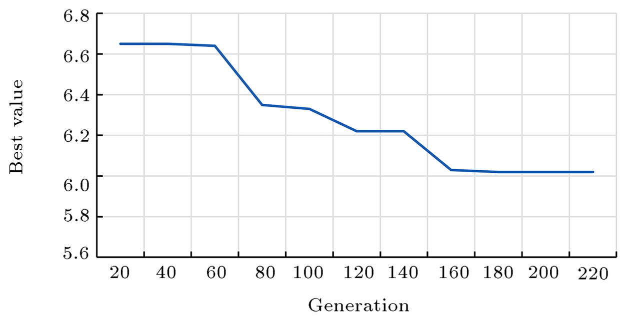
Problem	LINGO	GPSO			ODP(%)	Simulation based HIPSO			ODP(%)
		Max	Mean	SD		Max	Mean	SD	
S1	21287	29276	22499	5.7	17.7	23765	23499	10.1	10.4
S2	20142	23821	21374	6.1	13.5	22054	22544	6.43	11.9
S3	49020	56679	51643	5.4	14.2	55823	55643	12.08	13.5
S4	18094	28876	19288	6.6	24.4	21277	20288	9.65	12.1
S5	17376	28839	19384	11.6	18.8	20412	19684	8.43	13.6
S6	15994	21265	16703	4.4	28.7	19263	18703	6.5	16.9
S7	22777	26004	23990	5.3	10.5	25798	24990	4.76	9.7
S8	18009	22694	20231	12.3	21.7	21432	21091	11.04	17.1
S9	23697	31153	24872	5.0	22.2	27167	26872	10.73	13.4
S10	34623	47003	37486	8.3	20.2	39582	38486	8.61	11.2

Table 6. Obtained results by General Particle Swarm Optimization (GPSO) and simulation based Hierarchical Interactive Particle Swarm Optimization (HIPSO) for large test problems.

	Fitness value (mean)		RDP(%)	Number of iterations to converge		CPU time (seconds)	
Problem	GPSO	Sim-HIPSO		GPSO	Sim-HIPSO	GPSO	Sim-HIPSO
L1	65054	61299	5.77	19	10	62.59	26.54
L2	79429	72918	8.20	31	18	96.57	67.06
L3	85819	80611	6.07	48	29	118.07	75.35
L4	99633	95466	4.18	61	38	132.43	96.49
L5	1157173	1056480	8.70	83	42	138.43	98.03
L6	1581107	1475062	6.71	143	84	202.01	112.33
L7	1768096	1552178	12.21	177	97	246.23	137.76
L8	5617377	5016614	10.69	256	123	301.67	185.22
L9	6774706	6127052	9.56	415	187	416.44	254.76
L10	8758024	8197365	6.40	438	249	476.07	261.93

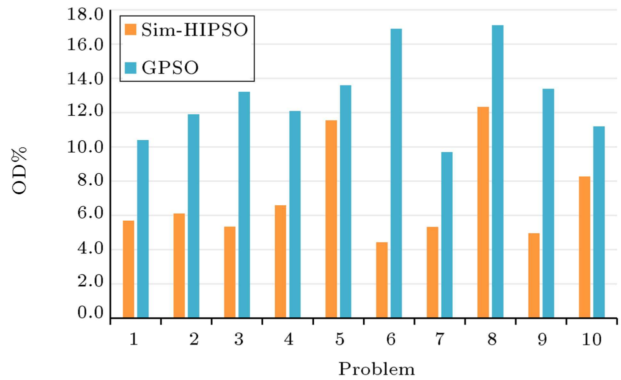


(a) GPSO

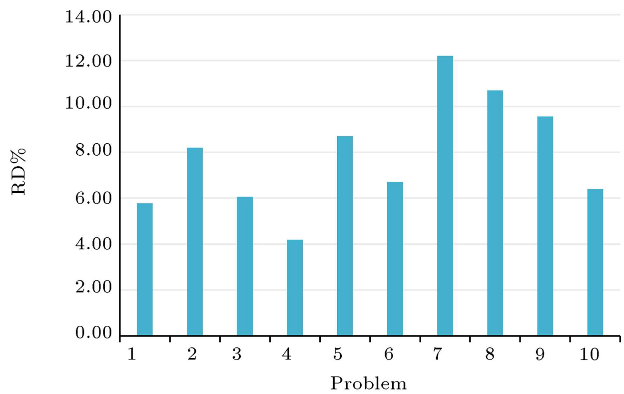


(b) Simulated based HIPSO

Figure 5. Comparative convergence results between the simulated Hierarchical Interactive Particle Swarm Optimization (HIPSO) and General Particle Swarm Optimization (GPSO) algorithms.



(a) Small test problems



(b) Large test problems

Figure 6. Comparative Optimality Difference Percentage (ODP)% and Relative Difference Percentage (RDP)% results for different problems.

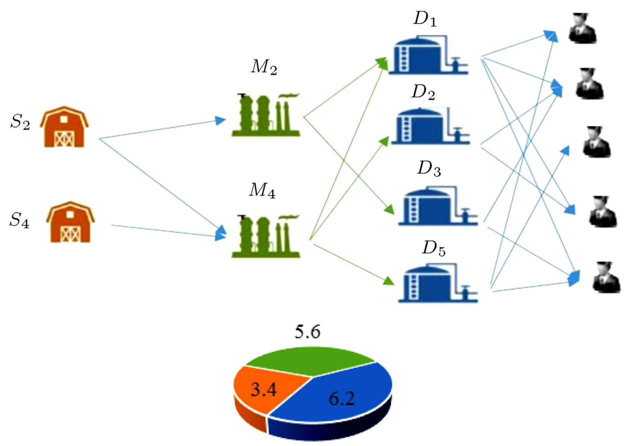


Figure 7. Results for the small-scale case study considered in this example.

converges to the optimal solution quickly and the fitness values of the solutions obtained after around 200 iterations are approximately equal to those of the optimal solutions. According to these graphs, it can be concluded that simulation based HIPSO algorithm has better performance than GPSO algorithm in terms of the obtained fitness values.

The ODP and RDP values obtained for both small and large problems are provided in Figure 6. For small problems, Figure 6(a) illustrates that the ODP values of LINGO are the best, followed by the values of Sim-HIPSO in most cases. In Figure 6(b), in addition, we observe that Sim-HIPSO performs better than other algorithms for large problems in most cases.

We give the details of the solution found for a small problem. This problem involves four suppliers, three production sites, six candidate distribution sites, and five customers. Each entity in the decentralized SC can gain a suitable profit. The results on the location and allocation of facilities are shown in Figure 7. In this case, distributers 1, 2, 3, and 5 are selected to be established, making the profit of 14.25 units, and manufacturers 2 and 4 are chosen to build two products with 235 and 182 units per period, making a profit of 10.34 units. The profit of the raw material supplier is 6.94 units. A total SC profit of 20.66 units is gained. Note that the distributer accounts for approximately 69% of the total profit. In fact, the distributer gains a considerably higher share of profit. This is likely due to the competitive advantage of the distributer for being able to act first. Thus, this entity can make the best decision based on all of the available data. Downstream entities should wait for the distributer to set the economic landscape before they make decisions. The findings show that the proposed model is able to efficiently solve small and large problems comprising inherent uncertain parameters.

7. Concluding remarks and suggestions for future studies

We studied the decentralized Supply Chain Network (SCN) as a multi-period multi-product Procurement-Production-Distribution Planning (PPDP) problem considering service level in presence of multiple uncertainties in Procurement, Production, and Distribution (PPD) layers. The proposed model incorporated strategic as well as tactical and operational decisions such as production planning, logistic, and inventory decisions. The problem was formulated as a Fuzzy-Stochastic Tri-Level Programming (FSTLP) model. Since the model was an uncertain mixed-integer tri-level problem, a hybrid Particle Swarm Optimization (PSO) algorithm along with a fuzzy simulation technique and stochastic simulation was designed to solve it. Effectiveness of the developed fuzzy stochastic optimization model and usefulness of the proposed solution approach were investigated by solving both small and large problems. The model was solved to optimality for small problems. In order to evaluate the application of the model and the proposed algorithm, 10 sets of small and large problems were randomly generated. The experimental results revealed that our proposed fuzzy-stochastic simulation based Hierarchical Interactive Particle Swarm Optimization (HIPSO) algorithm performed better in finding the best solutions within reasonable computational times and was effective for all the test problems. Our study, by presenting a comprehensive fuzzy and stochastic programming model, makes new contributions to the decentralized Supply Chain (SC) network literature. The provided model and solution methodology offer useful bases for the design of complex decentralized SC networks.

For future studies, incorporation of discounts on the orders of customer or inflation in system costs can be considered. Also, a more complete explanation of the features of the network can be offered by considering distance parameters between the layers or trip times for transportation of the products to the customer nodes. Incorporating man-made or natural disruptions such as terrorist attacks, floods, earthquakes, and economic crises and application of the robust optimization to the decentralized SCN design problem under uncertainty are also interesting to investigate. The proposed model may be considered by practical case studies, such as cement, brick, and iron industries.

Acknowledgments

The first two authors thank Mazandaran University of Science and Technology, the third author thanks Sharif University of Technology, and the fourth author thanks Babol University of Technology for supporting this work.

References

1. Farahani, R.Z., Rezapour, S., Drezner, T., and Fallah, S. "Competitive supply chain network design: An overview of classifications, models, solution techniques and applications", *Omega*, **45**, pp. 92–118 (2014).

2. Calvete, H.I., Galé, C., and Iranzo, J.A. "Planning of a decentralized distribution network using bilevel optimization", *Omega*, **49**, pp. 30–41 (2014).

3. Ma, Y., Yan, F., Kang, K., and Wei, X. "A novel integrated production-distribution planning model with conflict and coordination in a supply chain network", *Knowledge-Based Systems*, **105**, pp. 119–133 (2016).

4. Park, B.J., Choi, H.R., and Kang, M.H. "Integration of production and distribution planning using a genetic algorithm in supply chain management", *Analysis and Design of Intelligent Systems Using Soft Computing Techniques*, **41**, pp. 416–426 (2007).

5. Kazemi, A., Zarandi, M.F., and Husseini, S.M. "A multi-agent system to solve the production-distribution planning problem for a supply chain: a genetic algorithm approach", *The International Journal of Advanced Manufacturing Technology*, **44**, pp. 180–193 (2009).

6. Jolai, F., Razmi, J., and Rostami, N. "A fuzzy goal programming and meta heuristic algorithms for solving integrated production: distribution planning problem", *Central European Journal of Operations Research*, **19**, pp. 547–569 (2011).

7. Liu S., Sawik, T., and Papageorgiou, L.G. "Corrigendum to multiobjective optimisation of production, distribution and capacity planning of global supply chains in the process industry", *The International Journal of Management Science*, **44**, pp. 149–170 (2014).

8. Sarrafha, K., Rahmati, S.H.A., Niaki, S.T.A., and Zaretalab, A. "A bi-objective integrated procurement, production, and distribution problem of a multi-echelon supply chain network design: A new tuned MOEA", *Computers & Operations Research*, **54**, pp. 35–51 (2015).

9. Chang, Y.H. "Adopting co-evolution and constraint-satisfaction concept on genetic algorithms to solve supply chain network design problems", *Expert Systems with Applications*, **37**, pp. 6919–6930 (2010).

10. Guo, Z., Zhang, D., Leung, S., and Shi, L. "A bi-level evolutionary optimization approach for integrated production and transportation scheduling", *Applied Soft Computing*, **42**, pp. 215–228 (2016).

11. Marinakis, Y. "An improved particle swarm optimization algorithm for the capacitated location routing problem and for the location routing problem with stochastic demands", *Applied Soft Computing*, **37**, pp.

680–701 (2015).

12. Wang, K.J. and Lee, C.-H. "A revised ant algorithm for solving location-allocation problem with risky demand in a multi-echelon supply chain network", *Applied Soft Computing*, **32**, pp. 311–321 (2015).

13. Chu, Y., You, F., Wassick, J.M., and Agarwal, A. "Integrated planning and scheduling under production uncertainties: Bi-level model formulation and hybrid solution method", *Computers & Chemical Engineering*, **72**, pp. 255–272 (2015).

14. Wang, K.J., Makond, B., and Liu, S.Y. "Location and allocation decisions in a two-echelon supply chain with stochastic demand - A genetic-algorithm based solution", *Expert Systems with Applications*, **38**, pp. 6125–6131 (2011).

15. Amirtaheri, O., Zandieh, M., and Dorri, B. "A bi-level programming model for decentralized manufacturer-distributer supply chain considering cooperative advertising", *Scientia Iranica*, **25**(2), pp. 891–910 (2018).

16. Calvete, H.I., Galé, C., and Oliveros, M.-J. "Bilevel model for production-distribution planning solved by using ant colony optimization", *Computers and Operations Research*, **38**, pp. 320–327 (2011).

17. Camacho-Vallejo, J.-F., Muñoz-Sánchez, R., and González-Velarde, J.L. "A heuristic algorithm for a supply chain's production-distribution planning", *Computers and Operations Research*, **61**, pp. 110–121 (2015).

18. Xu, X., Meng, Z., and Shen, R. "A tri-level programming model based on conditional value-at-risk for three-stage supply chain management", *Computers and Industrial Engineering*, **66**, pp. 470–475 (2013).

19. Mirzapour Al-E-Hashem, S., Malekly, H., and Aryanezhad, M. "A multi-objective robust optimization model for multi-product multi-site aggregate production planning in a supply chain under uncertainty", *International Journal of Production Economics*, **134**, pp. 28–42 (2011).

20. Azaron, A., Brown, K., Tarim, S., and Modarres, M. "A multi-objective stochastic programming approach for supply chain design considering risk", *International Journal of Production Economics*, **116**, pp. 129–138 (2008).

21. Yongheng, J., Rodriguez, M.A., Harjunkoski, I., and Grossmann, I.E. "Optimal supply chain design and management over a multi-period horizon under demand uncertainty, Part II: A Lagrangean decomposition algorithm", *Computers & Chemical Engineering*, **62**, pp. 211–224 (2014).

22. Song, D.P., Dong, J.X., and Xu, J. "Integrated inventory management and supplier base reduction in a supply chain with multiple uncertainties", *European Journal of Operational Research*, **232**, pp. 522–536 (2014).

23. Pasandideh, S.H.R., Niaki, S.T.A., and Asadi, K. "Bi-objective optimization of a multi-product multi-period three-echelon supply chain problem under uncertain environments: NSGA-II and NRGA", *Information Sciences*, **292**, pp. 57–74 (2015).

24. da Silva, A.F. and Marins, F.A.S. "A fuzzy goal programming model for solving aggregate production-planning problems under uncertainty: A case study in a Brazilian sugar mill", *Energy Economics*, **45**, pp. 196–204 (2014).

25. Zhang, G., Shang, J., and Li, W. "Collaborative production planning of supply chain under price and demand uncertainty", *European Journal of Operational Research*, **215**, pp. 590–603 (2011).

26. Saranwong, S. and Likasiri, C. "Bi-level programming model for solving distribution center problem: a case study in Northern Thailand's sugarcane management", *Computers & Industrial Engineering*, **103**, pp. 26–39 (2017).

27. Saranwong, S. and Likasiri, C. "Product distribution via a bi-level programming approach: algorithms and a case study in municipal waste system", *Expert Systems with Application*, **44**, pp. 78–91 (2016).

28. Yue, D. and You, F. "Stackelberg-game-based modeling and optimization for supply chain design and operations: A mixed integer bilevel programming framework", *Computers and Chemical Engineering*, **102**, pp. 81–95 (2017).

29. Fard, A.M.F. and Hajaghaei-Keshteli, M. "A tri-level location-allocation model for forward/reverse supply chain", *Applied Soft Computing*, **62**, pp. 328–346 (2018).

30. Dempe, S., *Foundations of Bilevel Programming*, Springer Science & Business Media (2002).

31. Bard, J., *Practical Bilevel Optimization: Applications and Algorithms*, Kluwer Academic Press, Dordrecht, Netherlands (1998).

32. Bard, J.F. and Falk, J.E. "An explicit solution to the multi-level programming problem", *Computers & Operations Research*, **9**, pp. 77–100 (1982).

33. Bard, J.F. and Moore, J.T. "A branch and bound algorithm for the bilevel programming problem", *SIAM Journal on Scientific and Statistical Computing*, **11**, pp. 281–292 (1990).

34. Talbi, E.G., *Metaheuristics for Bi-level Optimization*, Springer (2013).

35. Zadeh, L.A. "Fuzzy sets", *Information and Control*, **8**, pp. 338–353 (1965).

36. Dubois, D. and Prade, H. "Systems of linear fuzzy constraints", *Fuzzy Sets and Systems*, **3**, pp. 37–48 (1980).

37. Camacho-Vallejo, J.-F., Cordero-Franco, Á.E., and González-Ramírez, R.G. "Solving the bilevel facility

location problem under preferences by a Stackelberg-evolutionary algorithm", *Mathematical Problems in Engineering*, **201**, pp. 30–44 (2014).

38. Kennedy, J. and Eberhart, R.C., *Swarm Intelligence*, San Francisco, USA: Morgan Kaufmann Publishers (2001).

39. Liu, B., *Uncertain Programming*, John Wiley & Sons, Inc. (1999).

40. Jia, L., Wang, Y., and Fan, L. "Multi-objective bilevel optimization for production-distribution planning problems using hybrid genetic algorithm [J]". *Integrated Computer-Aided Engineering*, **21**(1), pp. 77–90 (2014).

Biographies

Raheleh Nourifar is a PhD candidate in Industrial Engineering at Mazandaran University of Science and Technology, Babol, Iran. He was graduated with BSc and MSc in Industrial Engineering from Mazandaran University of Science and Technology, Babol, Iran, in 2006 and 2008, respectively. His research interests are supply chain management, multi-criteria decision making techniques, and network programming.

Iraj Mahdavi is a Full Professor of Industrial Engineering at Mazandaran University of Science and Technology, Babol, Iran. He received his PhD in Production Engineering in India. He is also on the editorial board of five journals and a scientific committee member of international conferences. He was awarded as the best researcher in the field of engineering in Iran and is among the best professors in the country. He has published over 350 research papers. His research interests include production planning, supply chain management, fuzzy networks, digital management, and intelligent operations management. He has also edited and authored two books on electronic supply network design and multi-objective optimization, respectively.

Nezam Mahdavi-Amiri is a Full Professor in Mathematical Sciences at Sharif University of Technology, Tehran, Iran. He received his PhD degree from Johns Hopkins University in Mathematical Sciences in 1981. He has been on the editorial board of several mathematical and computational journals in Iran including Bulletin of the Iranian Mathematical Society (as the style-language editor), Iranian Journal of Operations Research (Editor-in-Chief), and CSI Journal of Computer Science and Engineering. He was also the Editor-in-Chief of Mathematical Thought and Culture (in Persian) and a journal of the Mathematical Society of Iran. In addition, Dr. Mahdavi-Amiri was in the Executive Council of the Iranian Mathematical Society as well as the Computer Society of Iran. He is currently vice president of the Operations Research Society of Iran and its representative to IFORS. His research interests include optimization, numerical analysis and scientific computing, matrix computations, and fuzzy modeling and computing.

Mohammad Mahdi Paydar is an Assistant Professor of Industrial Engineering at Babol Noshirvani University of Technology. He received his PhD in Industrial Engineering from Iran University of Science and Technology. His research interests are cellular manufacturing systems, supply chain design, and modelling of manufacturing applications. He has published articles in journals such as Computers and Industrial Engineering, Computers and Operations Research, Expert Systems with Applications, Computers and Chemical Engineering, International Journal of Advanced Manufacturing Technology, Journal of Manufacturing Systems, International Journal of Production Research, and International Journal of Operational Research and 25 papers in international conferences.