

Research Article

Robust Optimization of Fourth Party Logistics Network Design under Disruptions

Jia Li,¹ Yanqiu Liu,¹ Ying Zhang,² and Zhongjun Hu¹

¹School of Management, Shenyang University of Technology, Shenyang 110870, China

²School of Science, Shenyang University of Technology, Shenyang 110870, China

Correspondence should be addressed to Jia Li; simsforever@sina.com

Received 25 September 2014; Revised 12 January 2015; Accepted 13 January 2015

Academic Editor: Pavel Rehak

Copyright © 2015 Jia Li et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The Fourth Party Logistics (4PL) network faces disruptions of various sorts under the dynamic and complex environment. In order to explore the robustness of the network, the 4PL network design with consideration of random disruptions is studied. The purpose of the research is to construct a 4PL network that can provide satisfactory service to customers at a lower cost when disruptions strike. Based on the definition of β -robustness, a robust optimization model of 4PL network design under disruptions is established. Based on the NP-hard characteristic of the problem, the artificial fish swarm algorithm (AFSA) and the genetic algorithm (GA) are developed. The effectiveness of the algorithms is tested and compared by simulation examples. By comparing the optimal solutions of the 4PL network for different robustness level, it is indicated that the robust optimization model can evade the market risks effectively and save the cost in the maximum limit when it is applied to 4PL network design.

1. Introduction

With the development of outsourcing and the rising expectations on service level of customers, unique and comprehensive supply chain solutions can no longer be achieved by a single third party logistics (3PL) provider. Thus, the fourth party logistics (4PL) need to be developed eagerly for the modern logistics challenges [1]. Depending on its strong capability of supply chain integration, 4PL greatly improves the efficiency and quality of logistics operations and reduces logistics costs [2].

Many researchers have studied the 4PL and it has received increasing attention. A tabu search algorithm for model of integration of 4PL was developed in [3]. A conceptual model based on 4PL implementation was presented in [4]. Cui et al. [5] described the 4PL routing problem with fuzzy duration time and established a chance-constrained programming model. Many scholars research on 4PL from different angles, but how to construct a 4PL network that not only performs efficiently under normal but also performs its intended function relatively well in the presence of disruptions is the essential core issue in 4PL mode. The robust optimization can be considered as an approach for dealing with the failures of

components or subsystems. Three of the most recent robust optimization approaches are introduced by Mulvey et al. [6], Ben-Tal and Nemirovski [7–9], and Bertsimas and Sim [10, 11]. Many robustness criteria have been proposed in the literature, and most have been applied to facility location or network design problems [12].

The 4PL network is a complex engineered system due to its size, span, and the network flow. Many potential threats can lead disruptions, such as industrial accidents, natural disasters, and terrorist strikes [13]. Bunschuh et al. [14] add supplier sourcing constraints to a supply chain network to mitigate disruptions. Various disruptions may make network cannot work normally and pose a threat to the security of 4PL network. Also, they could cause serious operational consequences, such as higher transportation costs, inventory shortages, order delays, and loss of market shares. So, it is very significant to explore the robustness of the network and improve its anti-interference ability for 4PL.

To our best knowledge, there is no research work about robust optimization of 4PL network design under uncertain conditions. However, the related problem about logistics and supply chain network optimization considering security and robustness has been studied by researchers. A review on

robust supply chain network design is given by Klibi et al. [15]. A robust model for an integrated logistics network design which takes facility disruptions into consideration was proposed by Hatefi and Jolai [16]. Peng et al. [17] presented a strategic supply chain management problem to design reliable logistics networks considering robustness level. Baghalian et al. [18] developed a mathematical model for supply chain network design comprising distribution centers, several capacitated production facilities, and retailers in markets under uncertainty. Chen [19] presented an efficient algorithm to solve the problem of the optimal capacity planning in robust supply chain management. Hu et al. [20] researched the robust scheduling optimization problem of resource constrained emergency rescue. Pishvaei et al. [21] and Hasani et al. [22] considered the robust optimization approach developed by Ben-Tal and Nemirovski [7–9] to design robust closed-loop supply chain networks.

The existing studies just can cope with the robust optimization of the 3PL network problems, while this paper studies robust optimization of 4PL network design under disruptions. According to the mentioned literature, we are the first to use the β -robustness measure to design the 4PL network to ensure high-level performance during disruptions. We construct 4PL networks which can provide satisfactory service to customers at a lower cost when disruptions strike. We formulate the robust optimization model of 4PL network design which incorporates the β -robustness measure in the constraints. For the NP-hard characteristic of the problem, we will use artificial fish swarm algorithm (AFSA) and genetic algorithm (GA) to optimize and simulate the model. The effectiveness of the algorithms is tested and compared by simulation examples. The results indicate that the AFSA outperforms the classic GA. Meanwhile, they validate that the robust optimization model can evade the market risks effectively and provide optimized strategy support for the 4PL providers. Section 2 describes the formulation of the 4PL network design under disruptions. Section 3 discusses the process of AFSA to solve this problem. The experimental results and comparisons are analyzed in Section 4. Finally, Section 5 provides the conclusions and prospects to this work.

2. Model Formulation

2.1. Description of Problem. We consider a multisource to multideestination 4PL network consisting of 3PL providers, suppliers, intermediate facilities, and demanders.

Figure 1 is a directed graph $G(V, E)$ for the potential 4PL network with nine nodes. The arcs denote 3PL providers. Since there may exist several 3PL providers for the transportation between any two nodes, there may have been multiple arcs between two nodes. 4PL network design decisions include determining 3PL providers and the locations of facilities. The objective is to construct a 4PL network under disruptions, which minimize the sum of the fixed cost and the expected transportation cost subjective to the robustness level constraint.

For the convenience of model establishment, we make the following assumptions:

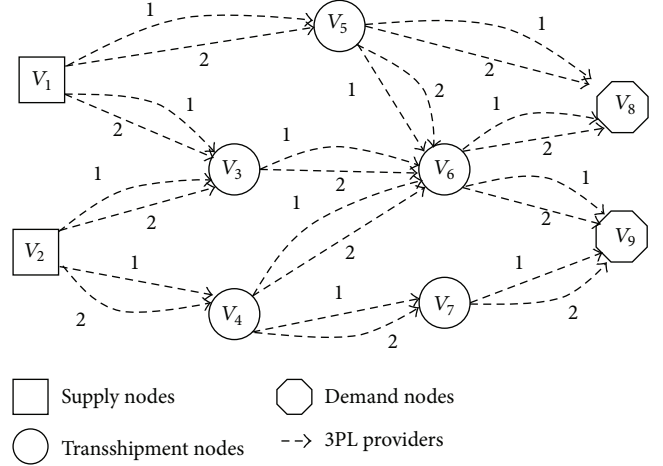


FIGURE 1: Potential 4PL network.

- (1) The nodes are all secure; they can not be disrupted.
- (2) Each 3PL provider may be disrupted randomly and independent to each other.

The variables and parameters that will be used in the model are defined as follows.

Sets

V : set of nodes including suppliers, transit facilities, and demanders, $V = O \cup M \cup D$.

O : sets of suppliers.

M : set of potential transit facilities.

D : set of demanders.

E : set of potential arcs.

K_{ij} : set of potential 3PL providers between nodes i and j .

S : set of scenarios.

Parameters

a_{ijk} : the k th 3PL provider between nodes i and j , $\forall i, j \in V, \forall k \in K_{ij}$.

F_{ijk} : the fixed cost of a_{ijk} , $\forall i, j \in V, \forall k \in K_{ij}$.

C_{ijk} : the unit transportation cost of a_{ijk} , $\forall i, j \in V, \forall k \in K_{ij}$.

L_{ijk} : the transportation capacity of a_{ijk} , $\forall i, j \in V, \forall k \in K_{ij}$.

P_{ijk} : the failure rate of a_{ijk} , $\forall i, j \in V, \forall k \in K_{ij}$.

W_d : the quantity of demand of demander d , $d \in D$.

U_o : the maximum supply quantity of supplier o , $o \in O$.

F'_m : the fixed cost of transit facility m , $m \in M$.

C'_m : the unit processing cost at facility m , $m \in M$.

L'_m : the processing capacity of facility m , $m \in M$.

λ : the unit penalty cost for not satisfying demand due to disruption.

Decision Variables

$$x_{ijk} = \begin{cases} 1 & \text{if } a_{ijk} \text{ is selected} \\ 0 & \text{otherwise.} \end{cases}$$

$$y_u = \begin{cases} 1 & \text{if facility } u \text{ is selected} \\ 0 & \text{otherwise.} \end{cases}$$

X : the vector including all decision variables x_{ijk} .

Y : the vector including all decision variables y_u .

f_{ijk}^s : the flow amount of a_{ijk} when scenario s occurs, $\forall i, j \in V, \forall k \in K_{ij}, \forall s \in S$.

Assistant Variables

$$R_{ijk}^s = \begin{cases} 1 & \text{if } a_{ijk} \text{ disrupts under scenario } s \\ 0 & \text{otherwise.} \end{cases}$$

2.2. Robustness of the 4PL Network. The robustness of the 4PL network is the ability of maintaining operation function at a lower cost when the system faces to the external and internal management in emergencies and other uncertainties of disturbance.

Definition of β -Robustness. For a given set S of scenarios, let β_s be a deterministic (i.e., single-scenario) optimization problem for scenario s . Let C_s^* be the optimal objective value for β_s . Let (X, Y) be a solution to β_s for all $s \in S$ and let $C_s(X, Y)$ be the objective value of problem β_s under solution (X, Y) . Then (X, Y) is called β -robust if, for all $s \in S$,

$$\frac{C_s(X, Y) - C_s^*}{C_s^*} \leq \beta \quad (1)$$

or equivalently,

$$C_s(X, Y) \leq (1 + \beta) C_s^* \quad (2)$$

where $\beta \geq 0$ is a given constant which denotes the robustness level. $(C_s(X, Y) - C_s^*)/C_s^*$ is the regret value for scenario s . The β -robust measure sets upper bounds on the maximum allowable regret value for each scenario.

The β -robust measure, in which the relative regret in each scenario is required to be no more than a constant β , was first applied in a facility layout problem [23] and later in an uncapacitated network design problem [24]. Snyder and Daskin [25] adopt the term “ β -robust” to describe this measure. They formulate the stochastic β -robust k -median problem and the β -robust uncapacitated fixed charge location problem.

2.3. Robust Optimization Model of 4PL Network Design under Disruptions. The robust model for 4PL network design under disruptions can be described as follows:

$$\min \sum_{i \in V} \sum_{j \in V} \sum_{k \in K_{ij}} x_{ijk} F_{ijk} + \sum_{m \in M} y_m F'_m + \sum_{s \in S} q_s C_s(X, Y) \quad (3)$$

subject to

$$C_s(X, Y) \leq (1 + \beta) C_s^*, \quad \forall s \in S, \quad (4)$$

$$x_{mjk} \leq y_m \quad \forall m \in M, \forall j \in V, \forall k \in K_{mj}, \quad (5)$$

$$x_{jmk} \leq y_m \quad \forall m \in M, \forall j \in V, \forall k \in K_{jm}, \quad (6)$$

$$\sum_{j \in V} \sum_{k \in K_{mj}} x_{mjk} \leq y_m \quad \forall m \in M, \quad (7)$$

$$\sum_{j \in V} \sum_{k \in K_{jm}} x_{jmk} \leq y_m \quad \forall m \in M, \quad (8)$$

$$x_{ijk}, y_m \in \{0, 1\} \quad \forall i, j \in V, \forall m \in M, \forall k \in K_{ij}. \quad (9)$$

The objective function (3) minimizes the sum of the fixed cost and the expected operation cost. Constraint (4) requires the cost in each scenario cannot be more than $100(1 + \beta)\%$ of the optimal operation cost C_s^* . Constraints (5) and (6) ensure that no 3PL provider is selected when the connected facility is not selected. Constraints (7) and (8) ensure that when a facility is selected, a 3PL provider will be selected at least, which transports products into or out of the facility. Constraint (9) denotes the binary variables.

The optimal operation costs C_s^* for each of the scenarios s can be calculated as follows:

$$C_s^* = \min \sum_{i \in V} \sum_{j \in V} \sum_{k \in K_{ij}} f_{ijk}^s C_{ijk} + \sum_{i \in V} \sum_{m \in M} \sum_{k \in K_{im}} f_{imk}^s C'_{imk} + \lambda \left(\sum_{d \in D} W_d - \sum_{d \in D} \sum_{i \in V} \sum_{k \in K_{id}} f_{idk}^s \right) \quad (10)$$

subject to

$$\sum_{j \in V} \sum_{k \in K_{oj}} f_{ojk}^s \leq U_o \quad \forall o \in O, s \in S \quad (11)$$

$$f_{ijk}^s \leq (1 - R_{ijk}^s) x_{ijk} L_{ijk} \quad \forall i, j \in V, \forall k \in K_{ij} \quad (12)$$

$$\sum_{j \in V} \sum_{k \in K_{mj}} f_{mjk}^s \leq y_m L'_m \quad \forall m \in M \quad (13)$$

$$\sum_{j \in V} \sum_{k \in K_{mj}} f_{mjk}^s - \sum_{j \in V} \sum_{k \in K_{jm}} f_{jmk}^s = 0 \quad \forall m \in M, s \in S \quad (14)$$

$$f_{ijk}^s \geq 0 \quad \forall i, j \in V, \forall k \in K_{ij}. \quad (15)$$

The objective function (10) is to minimize the operation cost in scenario s . It is the sum of the costs of transportation and penalty costs for not satisfying the demand. Constraint (11) requires the flow out to be less than or equal to maximum supply quantity of suppliers. Constraint (12) represents capacity of the selected 3PL provider must be not less than the transportation capacity required by customer. Constraint (13) ensures that the total flow through a facility does not exceed its processing capacity. Equation (14) means to keep a balance of the network flow. Constraint (15) is the nonnegative and integer condition constraints.

3. Intelligent Optimization Algorithms

The problem of 4PL network design under disruptions is NP-hard. The AFSA is developed to solve the problem. It is a stochastic search algorithm which simulates the food-seeking behaviors of the fish swarm [26]. It is a swarm intelligence algorithm and has been used successfully in various optimization problems [27]. W. Tian and Y. Tian [28] used the AFSA to solve task scheduling problems in autonomic multirobot group. Liu et al. [29] presented an AFSA to solve the multicast routing problem, which was abstracted as a Steiner tree problem in graphs. Yang [30] presented an AFSA by employing some improvements for the Multiple Sequence Alignment. The AFSA is capable of solving the problems by inspiration from the en masse movement of fishes. Fishes show different behaviors including seeking for food, following other fishes, protecting the group against threats, and stochastic search. This algorithm has many advantages including high convergence speed, flexibility, and high accuracy. The AFSA shows more intelligent behavior and obtains more optimized results compared with other algorithms. The classic GA is also used to compare the quality of the solutions obtained by AFSA.

3.1. Relevant Definitions. Let $Q = (Q_1, Q_2, \dots, Q_{NP})$ be the state of the artificial fish individual, and let NP be the population size of the artificial fish swarm.

$Q_i = \{q_i^1, q_i^2, \dots, q_i^r, \dots, q_i^n\}$ ($i = 1, 2, \dots, NP$) ($r = 1, 2, \dots, n$) are the decision variables, which denote the 4PL network structure (X, Y) . Fitness function $\text{Fitness}(Q)$ presents the food concentration of the artificial fish. $\text{distance}(Q_i, Q_j) = \|Q_i - Q_j\|$ is the distance between artificial fishes Q_i and Q_j . Visual is the artificial fish sight range. step is the step length. δ is a congestion factor. try_number is the maximum test number.

Definition 1. The distance between $Q_i = \{q_i^1, q_i^2, \dots, q_i^r, \dots, q_i^n\}$ and $Q_j = \{q_j^1, q_j^2, \dots, q_j^r, \dots, q_j^n\}$ is presented as follows:

$$\text{distance}(Q_i, Q_j) = \sum_{i=1}^{NP} \sum_{j=1}^{NP} \sum_{r=1}^n \text{sign}(|q_i^r - q_j^r|), \quad (16)$$

where

$$\text{sign}(x) = \begin{cases} 0, & x = 0 \\ 1, & x > 0 \\ -1, & x < 0 \end{cases} \quad (17)$$

is the number of elements which belong to neither Q_i nor Q_j .

Definition 2. Consider

$$\text{center}(Q_1, Q_2, \dots, Q_{NP}) = \text{most}_{i=1, \dots, NP} (q_i^1, q_i^2, \dots, q_i^n) \quad (18)$$

is the center of variables Q_1, Q_2, \dots, Q_{NP} .

3.2. Encoding. In order to use AFSA, we design a method of encoding the solution. As shown in Figure 2, the roads

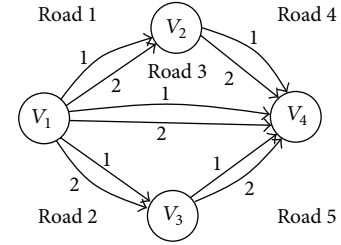


FIGURE 2: Encoding.

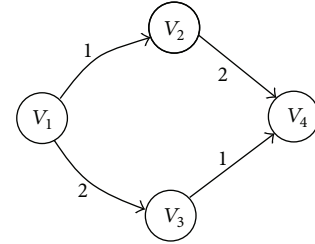


FIGURE 3: Decoding.

between two corresponding nodes of the 4PL network can be coded by Roads 1 to 5. The arcs (3PL providers) between facilities can be coded by integer number. The element encoded as 1 indicates that no 3PL provider is selected between two corresponding nodes. The element encoded as 2 means that the first 3PL provider is selected. The element encoded as 3 means that the second 3PL provider is selected. And the element encoded as 4 means that all the 3PL providers between two corresponding nodes are selected. For example, string $[2 \ 3 \ 1 \ 3 \ 2]$ represents the 4PL network structure in Figure 3.

3.3. Calculation of Food Concentration. The fitness function $\text{Fitness}(Q)$ denotes the food concentration of the artificial fish.

In this paper, the robustness level constraint (4) is used as a penalty added to the objective function. The fitness function is represented by

$$\begin{aligned} \text{Fitness}(Q_i) = & \min \sum_{i \in V} \sum_{j \in V} \sum_{k \in K_{ij}} x_{ijk} F_{ijk} \\ & + \sum_{m \in M} y_m F'_m + \sum_{s \in S} q_s C_s(X, Y) \\ & + \omega \cdot \max \left\{ 0, \sum_{s \in S} [C_s(X, Y) - (1 + \beta) C_s^*] \right\}. \end{aligned} \quad (19)$$

ω is the penalty factor. A smaller fitness value $\text{Fitness}(Q_i)$ indicates better fitness.

3.4. Artificial Fish Behavior

3.4.1. Prey Behavior. Let the current state of artificial fish be Q_i , and select a state Q_j which is selected randomly in its visual distance ($\text{distance}(Q_i, Q_j) < \text{Visual}$). If $\text{Fitness}(Q_j) < \text{Fitness}(Q_i)$, it goes forward a step in this direction; that is, let

$Q_i = Q_i + \text{rand}() \cdot \text{step} \cdot (Q_j - Q_i) / \|Q_i - Q_j\|$, where $\text{rand}()$ produces random numbers between 0 and 1. Otherwise, select a state Q_j randomly; that is, $Q_i = Q_i + \text{rand}() \cdot \text{Visual}$. Repeat the procedure try_number times. If it cannot satisfy the forward condition, it moves a step randomly.

3.4.2. Swarm Behavior. Let Q_{center} be the center position. n_f is the number of its companions in the current neighborhood ($\text{distance}(Q_i, Q_j) < \text{Visual}$), and n is total fish number. If $\text{Fitness}(Q_{\text{center}}) < \text{Fitness}(Q_i)$ and $n_f/n < \delta$, it means that the center position has more food and is not very crowded. Then it goes forward a step to the center position; that is, let $Q_i = Q_i + \text{rand}() \cdot \text{step} \cdot (Q_{\text{center}} - Q_i) / \|Q_{\text{center}} - Q_i\|$. Otherwise, execute the prey behavior.

3.4.3. Follow Behavior. Let Q_i explore the companion Q_{best} in the neighborhood ($\text{distance}(Q_i, Q_j) < \text{Visual}$), which has the best $\text{Fitness}(Q_{\text{best}})$. If $\text{Fitness}(Q_{\text{best}}) < \text{Fitness}(Q_i)$ and $n_f/n < \delta$, it goes forward a step to the companion Q_{best} ; that is, let $Q_i = Q_i + \text{rand}() \cdot \text{step} \cdot (Q_{\text{best}} - Q_i) / \|Q_{\text{best}} - Q_i\|$. Otherwise, execute the prey behavior.

3.4.4. Random Move Behavior. Choose a state Q_{new} randomly in the visual distance ($\text{distance}(Q_i, Q_j) < \text{Visual}$); then move towards this state; that is, let $Q_i = Q_i + \text{rand}() \cdot \text{step} \cdot (Q_{\text{new}} - Q_i) / \|Q_{\text{new}} - Q_i\|$.

3.4.5. Bulletin Board. The bulletin board keeps a record of the optimal artificial fish state and fitness value.

3.5. The Procedure of AFSA

Step 1. Generate the initial artificial fish population randomly. Set population size of the artificial fish NP , the maximum number of iterations NG , the maximum test number try_number , the artificial fish sight range Visual , the step length step , and the congestion factor δ .

Step 2. Calculate the food concentration, that is, fitness function $\text{Fitness}(Q)$.

Step 3. Simulate fish preying, following, swarming, and random moving behavior respectively. Update the bulletin board if a smaller fitness value arises.

Step 4. Check whether iteration times are over NG . If it is true, stop the procedure and output the optimum solution; else turn to Step 2.

3.6. Calculation of the Expected Operation Cost. The number of the disruption scenarios is very large. The expected operation cost $\sum_{s \in S} q_s C_s(X, Y)$ can be approximated using Monte Carlo simulation.

The following are steps of emulation computation.

Step 1. Generate N samples of scenarios, according to the disruption probability of the 3PL providers P_{ijk} .

TABLE 1: The data of nodes for the 9-node problem.

V_i	U_o	W_d	F'_m	C'_m	L'_m
1	157	\	\	\	\
2	142	\	\	\	\
3	\	\	636	34	120
4	\	\	763	44	142
5	\	\	651	37	119
6	\	\	645	24	166
7	\	\	761	48	179
8	\	62	\	\	\
9	\	44	\	\	\

TABLE 2: The data of arcs for the 9-node problem.

V_i	V_j	a_{ijk}	F_{ijk}	C_{ijk}	L_{ijk}
1	3	1	743	34	123
		2	728	40	189
	5	1	722	43	102
		2	961	42	162
2	3	1	968	26	118
		2	947	26	192
	4	1	907	24	106
		2	601	16	101
3	6	1	873	11	120
		2	733	47	145
4	6	1	919	37	120
		2	690	37	102
	7	1	855	30	183
		2	595	22	143
5	6	1	849	16	154
		2	927	44	138
	8	1	950	30	159
		2	909	36	182
6	8	1	864	31	134
		2	784	44	131
	9	1	773	38	137
7	9	2	811	38	144
		1	990	17	188
		2	938	20	127

Step 2. Calculate the value of $C_s(X, Y)$ for each sample scenario s .

Step 3. Estimate the expected operation cost as follows: $\sum_{s \in S} q_s C_s(X, Y) = (1/N) \sum_{s=1}^N C_s(X, Y)$.

4. Experimental Results and Comparisons

In this section, an instance is developed to evaluate the performance of the AFSA and the GA. The algorithms are coded in Matlab 7.0 and run on an Intel Core 2 Quad 2.66 GHz computer.

Assume that a 4PL provider plans to transport a consignment of goods from the supply nodes to the demand nodes.

TABLE 3: Comparison of two algorithms with different regret value β constraint.

Algorithm	Regret value β constraint	Objective value (\$) best	Objective value (\$) worst	Objective value (\$) ave.	Ave. CPU time (second)
AFSA	0.4	26160	28530	26856	576
	0.2	32970	33190	33070	
	0.1	33150	34880	33278	
GA	0.4	27180	29980	28742	868
	0.2	33010	33260	33192	
	0.1	34860	35980	35276	

Decision makers need to design the 4PL network which can provide satisfactory service to customers at a lower cost when disruptions strike. The topological structure of the potential 4PL network is shown in Figure 1. The failure rate of each 3PL provider is drawn uniformly from $[0, 1]$. The unit penalty cost for not satisfying demand due to disruption λ is set to be 500. By running a lot of experiments firstly, the parameters in AFSA are set below:

$$\begin{aligned} \omega = 2, \quad NP = 50, \quad NG = 200, \quad \text{Visual} = 20, \\ \text{step} = 1, \quad \delta = 0.3, \quad \text{try_number} = 20. \end{aligned} \quad (20)$$

The costs of the facilities and the 3PL providers are recorded by dollar. The capacity is recorded by ton, and the CPU time is recorded by second. The node and arc data are given in Tables 1 and 2.

The AFSA and GA are run 200 times. The experimental results and comparisons are shown in Table 3. It can be seen that the AFSA outperforms the classic GA under different regret value β . The AFSA has better stabilization and it can obtain better solutions.

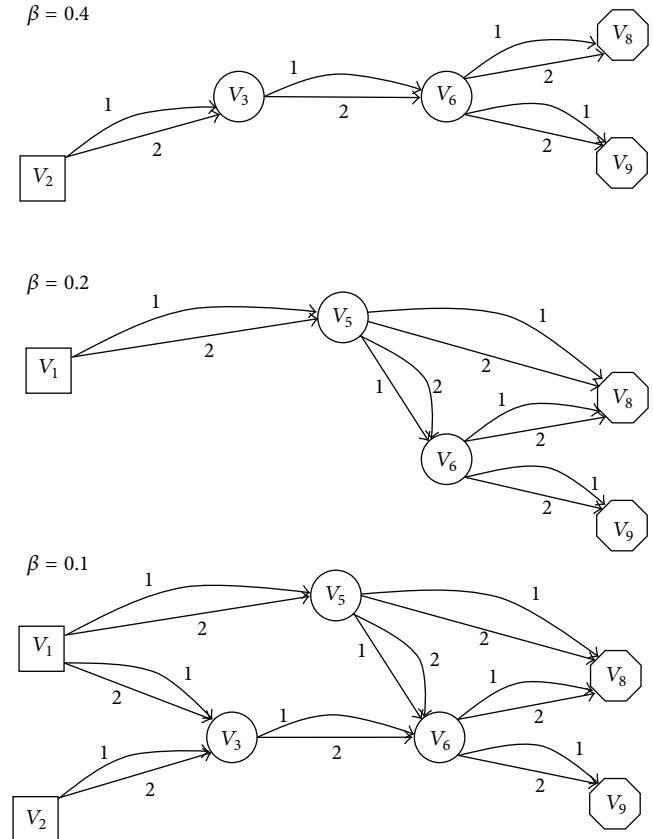
In order to analyze the influence of regret value β constraint to the 4PL network construction, we perform numerical experiments with different regret value β .

Figure 4 shows the results of the different 4PL network structures. $V_2, V_3, V_6, V_8,$ and V_9 are selected with $\beta = 0.4$. The minimum total cost is \$26,160. $V_1, V_5, V_6, V_8,$ and V_9 are selected with $\beta = 0.2$. The minimum total cost is \$32,970. $V_1, V_2, V_3, V_5, V_6, V_8,$ and V_9 are selected with $\beta = 0.1$. The minimum total cost is \$33,150. It is indicated that the 4PL network becomes more complex when the higher robustness level is desired. Besides, the total cost increases when the regret value β decreases.

Hence, the 4PL provider is allowed to adjust the robustness level and has options to design 4PL network.

5. Conclusions and Prospects

This paper studies the robust optimization of 4PL network design under disruptions. In order to make the 4PL network operate safely and efficiently, the concept of β -robustness is introduced into the network design problem. A robust optimization model with β -robust constraints is established to construct a 4PL network at a lower cost among disruption scenarios. The objective function of the mathematical model

FIGURE 4: 4PL network structures under different regret value β .

is minimizing the sum of the fixed cost and the expected operation cost. Based on the NP-hard characteristic of the problem, the AFSA and GA are presented and the performance of the algorithms is tested by simulation examples. According to the simulation of Matlab, the AFSA performs better in effectiveness and stability than the classic GA. After solving the model, we obtain different 4PL network structures which satisfy different robustness requirements with the lowest cost. We also demonstrate that the robust optimization model can evade the market risks effectively and save the cost in the maximum limit when it is applied to 4PL network design.

For the future research, more realistic problem such as a multiperiod and multiproduct integrated 4PL network design with disruptions may be taken into consideration. The scenario approach can also be adopted to model other sources of

uncertainties, such as customer demands and transportation costs. It is also possible to consider network design together with other supply chain problems, such as inventory management, capacity expansion, and vehicle routing.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (no. 70431003), the Science and Technology Support Program of Liaoning Province (no. 2013216015), and the Science and Technology Support Program of Shenyang (no. F13-051-2-00 and no. F14-231-1-24).

References

- [1] J. Li, Y. Q. Liu, and Z. J. Hu, "Routing optimization of fourth party logistics with reliability constraints based on Messy GA," *Journal of Industrial Engineering and Management*, vol. 7, no. 5, pp. 1097–1111, 2014.
- [2] J. M. Yao, "Decision optimization analysis on supply chain resource integration in fourth party logistics," *Journal of Manufacturing Systems*, vol. 29, no. 4, pp. 121–129, 2010.
- [3] Y. Wang, H. Zhao, and Y. Li, "Tabu search algorithm for optimization model of integration of job of 4th party logistics," *Journal of Systems Engineering*, vol. 21, no. 2, pp. 143–149, 2006.
- [4] A. Win, "The value a 4PL provider can contribute to an organisation," *International Journal of Physical Distribution and Logistics Management*, vol. 38, no. 9, pp. 674–684, 2008.
- [5] Y. Cui, M. Huang, S. Yang, L. H. Lee, and X. Wang, "Fourth party logistics routing problem model with fuzzy duration time and cost discount," *Knowledge-Based Systems*, vol. 50, pp. 14–24, 2013.
- [6] J. M. Mulvey, R. J. Vanderbei, and S. A. Zenios, "Robust optimization of large-scale systems," *Operations Research*, vol. 43, no. 2, pp. 264–281, 1995.
- [7] A. Ben-Tal and A. Nemirovski, "Robust convex optimization," *Mathematics of Operations Research*, vol. 23, no. 4, pp. 769–805, 1998.
- [8] A. Ben-Tal and A. Nemirovski, "Robust solutions of uncertain linear programs," *Operations Research Letters*, vol. 25, no. 1, pp. 1–13, 1999.
- [9] A. Ben-Tal and A. Nemirovski, "Robust solutions of linear programming problems contaminated with uncertain data," *Mathematical Programming*, vol. 88, no. 3, pp. 411–424, 2000.
- [10] D. Bertsimas and M. Sim, "Robust discrete optimization and network flows," *Mathematical Programming Series B*, vol. 98, no. 1–3, pp. 49–71, 2003.
- [11] D. Bertsimas and M. Sim, "The price of robustness," *Operations Research*, vol. 52, no. 1, pp. 35–53, 2004.
- [12] L. V. Snyder, "Facility location under uncertainty: a review," *IIE Transactions*, vol. 38, no. 7, pp. 547–564, 2006.
- [13] L. Huang, W. Wang, and M. G. Wang, "Simulation research of space-time evolution of emergency logistics network reliability based on complex network theory," *Discrete Dynamics in Nature and Society*, vol. 2013, Article ID 303187, 7 pages, 2013.
- [14] M. Bunschuh, D. Klabjan, and D. Thurston, "Modeling robust and reliable supply chains," *Optimization Online*. In press.
- [15] W. Klibi, A. Martel, and A. Guitouni, "The design of robust value-creating supply chain networks: a critical review," *European Journal of Operational Research*, vol. 203, no. 2, pp. 283–293, 2010.
- [16] S. M. Hatefi and F. Jolai, "Robust and reliable forward-reverse logistics network design under demand uncertainty and facility disruptions," *Applied Mathematical Modelling*, vol. 38, no. 9–10, pp. 2630–2647, 2014.
- [17] P. Peng, L. V. Snyder, A. Lim, and Z. Liu, "Reliable logistics networks design with facility disruptions," *Transportation Research Part B: Methodological*, vol. 45, no. 8, pp. 1190–1211, 2011.
- [18] A. Baghalian, S. Rezapour, and R. Z. Farahani, "Robust supply chain network design with service level against disruptions and demand uncertainties: a real-life case," *European Journal of Operational Research*, vol. 227, no. 1, pp. 199–215, 2013.
- [19] S.-G. Chen, "Optimal logistic planning in robust supply chain management," in *Proceedings of the 40th International Conference on Computers and Industrial Engineering (CIE40 '10)*, pp. 1–6, July 2010.
- [20] X. B. Hu, Z. W. He, and Y. Xu, "Robust scheduling optimization of emergency rescue based on resource constraints," *Operations Research and Management Science*, vol. 22, no. 2, pp. 72–79, 2013.
- [21] M. S. Pishvaei, M. Rabbani, and S. A. Torabi, "A robust optimization approach to closed-loop supply chain network design under uncertainty," *Applied Mathematical Modelling*, vol. 35, no. 2, pp. 637–649, 2011.
- [22] A. Hasani, S. H. Zegordi, and E. Nikbaksh, "Robust closed-loop supply chain network design for perishable goods in agile manufacturing under uncertainty," *International Journal of Production Research*, vol. 50, no. 16, pp. 4649–4669, 2012.
- [23] P. Kouvelis, A. A. Kurawarwala, and G. J. Gutiérrez, "Algorithms for robust single and multiple period layout planning for manufacturing systems," *European Journal of Operational Research*, vol. 63, no. 2, pp. 287–303, 1992.
- [24] G. J. Gutiérrez, P. Kouvelis, and A. A. Kurawarwala, "A robustness approach to uncapacitated network design problems," *European Journal of Operational Research*, vol. 94, no. 2, pp. 362–376, 1996.
- [25] L. V. Snyder and M. S. Daskin, "Stochastic p-robust location problems," *IIE Transactions*, vol. 38, no. 11, pp. 971–985, 2006.
- [26] W. Shen, X. Guo, C. Wu, and D. Wu, "Forecasting stock indices using radial basis function neural networks optimized by artificial fish swarm algorithm," *Knowledge-Based Systems*, vol. 24, no. 3, pp. 378–385, 2011.
- [27] M. Neshat, G. Sepidnam, M. Sargolzaei, and A. N. Toosi, "Artificial fish swarm algorithm: a survey of the state-of-the-art, hybridization, combinatorial and indicative applications," *Artificial Intelligence Review*, vol. 42, no. 4, pp. 965–997, 2014.
- [28] W. Tian and Y. Tian, "An improved artificial fish swarm algorithm for resource leveling," in *Proceedings of the International Conference on Management and Service Science (MASS '09)*, pp. 1–4, IEEE, Wuhan, China, September 2009.
- [29] Q. Liu, T. Odaka, J. Kuroiwa, H. Shirai, and H. Ogura, "An artificial fish swarm algorithm for the multicast routing problem," *IEICE Transactions on Communications*, vol. 97, no. 5, pp. 996–1011, 2014.
- [30] W.-H. Yang, "An improved artificial fish swarm algorithm and its application in multiple sequence alignment," *Journal of Computational and Theoretical Nanoscience*, vol. 11, no. 3, pp. 888–892, 2014.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

