

## Research Article

# A Hybrid Genetic-Simulated Annealing Algorithm for the Location-Inventory-Routing Problem Considering Returns under E-Supply Chain Environment

Yanhui Li,<sup>1</sup> Hao Guo,<sup>1</sup> Lin Wang,<sup>2</sup> and Jing Fu<sup>1,3</sup>

<sup>1</sup> School of Information Management, Central China Normal University, Wuhan 430079, China

<sup>2</sup> School of Management, Huazhong University of Science and Technology, Wuhan 430074, China

<sup>3</sup> Department of Industrial and System Engineering, State University of New York at Buffalo, Buffalo, NY 14228, USA

Correspondence should be addressed to Lin Wang; wanglin982@gmail.com

Received 4 October 2013; Accepted 18 November 2013

Academic Editors: T. Chen, Q. Cheng, and J. Yang

Copyright © 2013 Yanhui 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.

Facility location, inventory control, and vehicle routes scheduling are critical and highly related problems in the design of logistics system for e-business. Meanwhile, the return ratio in Internet sales was significantly higher than in the traditional business. Many of returned merchandise have no quality defects, which can reenter sales channels just after a simple repackaging process. Focusing on the existing problem in e-commerce logistics system, we formulate a location-inventory-routing problem model with no quality defects returns. To solve this NP-hard problem, an effective hybrid genetic simulated annealing algorithm (HGSAA) is proposed. Results of numerical examples show that HGSAA outperforms GA on computing time, optimal solution, and computing stability. The proposed model is very useful to help managers make the right decisions under e-supply chain environment.

## 1. Introduction

The increasing progress of information and prevalence of internet in the 21st century have forced the e-commerce to develop in world-wide rage. In 2012, B2C e-commerce sales grew 21.1% to top \$1 trillion for the first time in history of the whole world [1]. Comparing with traditional commerce, customers are liable to return goods under e-commerce environment. Note that many customer returns online accounts for 35% of original orders [2, 3]. Therefore, logistics systems as an important support system in e-commerce need to be adjusted and improved. To adapt to the reality of e-commerce market environment, reverse logistics network and highly integrated logistics process should be the necessities.

Facility location, inventory control, and vehicle routing decisions are critical problems in the design of logistics system. There is much previous work on these three areas. Furthermore, the related work on location and vehicle routing was extended into the field of computer communication and networks [4, 5]. In fact, there is a mutually dependent

relationship among these problems in logistics system. Comprehensive optimizing and logistics activities management should be based on this relationship [6]. According to this idea, besides location allocation problem and vehicle routing problem, two-two integration such as location-routing problem (LRP), inventory-routing problem (IRP), and location-inventory problem (LIP) and three integration problem (location-inventory-routing problem, LIRP) start to be researched.

Many papers about the LIP, LRP, and IRP are studied deeply and have made some abundant achievements. However, research on the integration of location-inventory-routing problem is limited. Some researchers strongly appeal to carry out research on LIRP [7, 8]. Liu and Lee [9] firstly proposed the LIRP; they built a model for single merchandise, multi-DPs LRP taking inventory control decisions into consideration and proposed a two-stage heuristic algorithm. In order to avoid being trapped in local optima, Liu and Lin [10] proposed a global optimum heuristic based on the algorithm in the above papers to solve the LIRP. Max Shen and Qi



decisions are as follows: (1) location decisions, the optimal number of MCs and their locations; (2) inventory decisions, the optimal order times and order size on a route; (3) routing arrangement, the vehicles deliver merchandises and collect returned merchandises in the order.

## 2.2. Assumptions

- (1) There is a single type of merchandise.
- (2) The total demand on each route is less than or equal to the vehicle capacity.
- (3) The vehicle type is homogeneous.
- (4) Each route is served by one vehicle.
- (5) Each route begins and ends at the same MC.
- (6) The capacity of MCs is infinite.
- (7) The forward distribution and reverse collection service could be met at the same time.
- (8) The daily demand and return of each DP are known.
- (9) The returned merchandises are without quality defect.
- (10) Returned merchandises are processed and repackaged at MCs.

**2.3. Model Formulation and Analysis.** The cost of  $MC_r$  consists of the following components.

- (i) The annual cost of the dispatching vehicle at  $MC_r$  is given by  $N_r e_r$ .
- (ii) The annual cost of placing an order at  $MC_r$  is given by  $N_r f_r$ .
- (iii) As for the returned merchandise without quality defects, it can be sold again as a normal goods being repackaged; therefore, transportation volume from plant to  $MC_r$  shall deduct the returns  $\sum_{i \in S} q_i$ . Thus, the annual transportation cost from plant to  $MC_r$  is given by  $\lambda \sum_{v \in V} \sum_{i \in S} b_r (d_i - q_i) Y_{ir}^v$ .
- (iv) As the same reason as the third point above, the annual inventory holding costs at  $MC_r$  should consider the returns, too. So, the annual inventory holding cost at  $MC_r$  is given by  $(\lambda \sum_{v \in V} \sum_{i \in S} h(d_i + q_i) Y_{ir}^v) / 2N_r$ . Since there is  $\sum_{i \in S} (d_i - q_i)$  in component (iii), and here has  $\sum_{i \in S} (d_i + q_i)$ , adding this two parts together, that is  $\sum_{i \in S} (d_i - q_i) + \sum_{i \in S} (d_i + q_i) = \sum_{i \in S} d_i$ . It means the goods flow is equal to demand.
- (v) The annual handling cost at  $MC_r$  is given by  $\lambda \sum_{v \in V} \sum_{i \in S} c_r d_i Y_{ir}^v$ .
- (vi) The annual repackaging cost of returned merchandises at  $MC_r$  is given by  $\lambda \sum_{v \in V} \sum_{i \in S} p q_i Y_{ir}^v$ .
- (vii) The annual total distribution costs from every MC to each DP is given by  $N_r \sum_{r \in R} \sum_{i \in S^+} \sum_{j \in S^+} \sum_{v \in V} l_{s_{ij}} X_{ijr}^v$ .
- (viii) The construction cost of  $MC_r$  is given by  $\sum_{i \in R} a_r Z_r$ .

The objective is to minimize the total cost of the system; we formulate the model as follows:

$$\begin{aligned} \min Z = & \sum_{r \in R} (e_r + f_r) N_r + \lambda \sum_{r \in R} \sum_{v \in V} \sum_{i \in S} b_r (d_i - q_i) Y_{ir}^v \\ & + \frac{\lambda \sum_{r \in R} \sum_{v \in V} \sum_{i \in S} h(d_i + q_i) Y_{ir}^v}{2N_r} \\ & + \lambda \sum_{r \in R} \sum_{v \in V} \sum_{i \in S} c_r d_i Y_{ir}^v + \lambda \sum_{r \in R} \sum_{v \in V} \sum_{i \in S} p q_i Y_{ir}^v \\ & + N_r \sum_{r \in R} \sum_{i \in S^+} \sum_{j \in S^+} \sum_{v \in V} l_{s_{ij}} X_{ijr}^v + \sum_{r \in R} a_r Z_r \end{aligned} \quad (1)$$

s.t.

$$Z_r \geq 1, \quad r \in R; \quad (2)$$

$$\sum_{v \in V} \sum_{r \in R} \sum_{i \in S^+} X_{ijr}^v = 1, \quad j \in S; \quad (3)$$

$$\sum_{v \in V} \sum_{r \in R} Y_{ir}^v = 1, \quad i \in S; \quad (4)$$

$$\sum_{j \in S^+} X_{kjr}^v - \sum_{i \in S^+} X_{ikr}^v = 0, \quad k \in S^+, \quad v \in V, \quad r \in R; \quad (5)$$

$$\sum_{i \in S} \sum_{r \in R} d_i Y_{ir}^v \leq g, \quad v \in V; \quad (6)$$

$$Y_{ir}^v - Z_r \leq 0, \quad r \in R, \quad i \in S, \quad v \in V; \quad (7)$$

$$\sum_{j \in S^+} X_{ijr}^v + \sum_{j \in S^+} X_{rjr}^v - Y_{ir}^v \leq 1, \quad i \in S, \quad v \in V, \quad r \in R; \quad (8)$$

$$Z_r = \{0, 1\}, \quad r \in R; \quad (9)$$

$$Y_{ir}^v = \{0, 1\}, \quad r \in R, \quad i \in S, \quad v \in V; \quad (10)$$

$$X_{ijr}^v = \{0, 1\}, \quad i \in S, \quad j \in S^+, \quad r \in R, \quad v \in V, \quad (11)$$

where the objective function (1) minimizes the system's total cost; (2) ensures at least one MC is established; (3) ensures each DP is served by the only one vehicle which belongs to a certain MC; (4) ensures that each route has only one vehicle; (5) ensures the continuity of delivery routes; (6) ensures vehicle cannot be overloaded; (7) ensures that only the selected MC can carry out distribution services; (8) ensures as long as a route passing through a DP, the corresponding MC would also be on this route; (9)–(11) ensure the integrality of decision variables.

## 3. Solution Approach

In this section, we first give the formula for solving optimal order times  $N_r$  and the optimal order size  $Q_r^v$ . Since calculating  $N_r$  and  $Q_r^v$  still relies on the decision variables  $X_{ijr}^v$ ,  $Y_{ir}^v$ , and  $Z_r$ , so we present a heuristic algorithm to get the optimized  $X_{ijr}^v$ ,  $Y_{ir}^v$ , and  $Z_r$ .

**3.1. Finding the Optimal Order Times.** In the models (1)–(11), the decision variable  $N_r$  only has appeared in the objective function. Also, the objective function is convex for  $N_r > 0$ . Consequently, we can obtain the optimal value of  $N_r$  by taking the derivative of the objective function with respect to  $N_r$  as

$$N_r = \sqrt{\frac{\sum_{v \in V} \sum_{i \in S} \lambda h(d_i + q_i) Y_{ri}^v}{2(e_r + f_r + \sum_{i \in S^+} \sum_{j \in S^+} \sum_{v \in V} l s_{ij} X_{ijr}^v)}}. \quad (12)$$

Then, the optimal order size can be given by

$$Q_r = \frac{\sum_{v \in V} \sum_{i \in S} \lambda d_i Y_{ri}^v}{N_r} = \left( \sum_{v \in V} \sum_{i \in S} \lambda d_i Y_{ri}^v \times \left( \sqrt{\frac{\sum_{v \in V} \sum_{i \in S} \lambda h(d_i + q_i) Y_{ri}^v}{2(e_r + f_r + \sum_{i \in S^+} \sum_{j \in S^+} \sum_{v \in V} l s_{ij} X_{ijr}^v)}} \right)^{-1} \right). \quad (13)$$

**3.2. Hybrid Genetic Simulated Annealing Algorithm (HGSAA).** The LIRP contains the VRP. As we know, the VRP is an NP-hard problem. This makes LIRP more complicated. It is generally believed that there is no complete, accurate, and not too slow analytic algorithm to solve NP-hard problems. Noting bioinspired computation is widely used for solving optimization problems, we designed a hybrid algorithm based on GA and simulated annealing (SA) to solve the proposed model.

Traditional GA has strong global search ability in solving such problems, but also has defects such as premature and weak local search ability. On the other hand, SA has strong local search ability and no premature problem. Therefore, the combination of GA and SA can overcome the defects of each of the two methods, bring into play their respective advantages, and improve the solving efficiency. This algorithm is named hybrid genetic simulated annealing algorithm (HGSAA).

### 3.2.1. Relevant Operations of GA

**(1) Encoding.** In a genetic algorithm, a population of candidate solutions (called individuals) to an optimization problem is evolved toward better solutions. Each individual with a set of properties, such as its chromosomes or genotype, can be mutated and altered. Traditionally, solutions are represented in binary as strings of 0s and 1s.

This study adopts the natural number coding method; using unrepeatable  $R + S$  natural numbers constitute a sequence, which represents an individual. While  $1, 2, \dots, R$  indicate candidate MCs and  $R + 1, \dots, R + S$  indicate DPs,

the encoding can describe a candidate solution of the above optimization problem. For example, in Figure 1, the code of the individual corresponding solutions is  $\{1, 5, 13, 6, 2, 7, 14, 21, 15, 8, 3, 4, 9, 16, 22, 17, 10, 11, 18, 20, 19, 12\}$ .

**(2) Fitness Function.** A fitness function is a particular type of objective function that is used to measure the quality of the represented solution. In this study, the fitness function is defined as

$$f_k = \frac{1}{Z}. \quad (14)$$

**(3) Selection.** During each successive generation, a proportion of the existing population is selected to breed a new generation. Individual solutions are selected through a fitness-based process, where fitter solutions (as measured by a fitness function) are typically more likely to be selected. Wheel selection operator [21] (also known as proportional selection operator) is used. Suppose the population size is  $N$ , the fitness value of the individual  $k$  is  $f_k$ , a number  $\xi \in [0, 1]$  is generated randomly. If  $(\sum_{k=1}^{i-1} f_k / \sum_{k=1}^N f_k) < \xi \leq (\sum_{k=1}^i f_k / \sum_{k=1}^N f_k)$  then individual  $i$  is selected to be replicated.

Because this method has great randomness in the selection of individuals, the simulated annealing algorithm with faster local convergence is added to the GA to increase the convergence speed in selection operation.

**(4) Crossover.** Crossover is a process of taking more than one parent individuals and producing a child individual from them. Crossover is used to vary the programming of a chromosome or chromosomes from one generation to the next. Partially matched crossover (PMX) [22] is used in this paper.

**Step 1.** Select two parent individuals randomly from the population;

**Step 2.** generate two random cut points to represent the mapped segments;

**Step 3.** exchange the segments of the two parent individuals to produce two new individuals;

**Step 4.** determine the mapping relations between two segments;

**Step 5.** legalize two new individuals with mapping relationship through repair strategy.

**(5) Mutation.** Mutation is used to maintain genetic diversity from one generation of a population of individuals to the next. The purpose of mutation in GAs is preserving and introducing diversity. Mutation should allow the algorithm to avoid local minima by preventing the population of individuals from becoming too similar to each other, thus slowing or even stopping evolution. A simple and efficient mutation operation, that is, swap mutation [23], is used. The details are as follows.



*Step 1.* Select one parent individual randomly from the population;

*Step 2.* generate two random numbers to represent the mutation points;

*Step 3.* swap the positions of these two mutation points to produce a new individual.

Compared with other mutations, studies show that convergence rate of this method has a greater advantage in population control. It can effectively prevent premature convergence of GA and avoid the occurrence of local optimal solution.

### 3.2.2. Relevant Operations of SA

(1) *The Annealing Process to Accept the New Individual.* In order to prevent the population into local optimization, the Metropolis acceptance criteria in SA are applied into the GA in this paper. We reserved the best parent individual in a population named old, and then selected the best offspring individual in another population named new; old and new go into the next generation population through competition. Let  $\Delta_f = f_{\text{new}} - f_{\text{old}}$ , if  $\Delta_f < 0$ , then the individual *new* is received, and hold it to the next generation; otherwise, the individual *new* is received with the probability  $p = \exp(-\Delta_f/t) > \text{random digit}$ , where  $t$  is annealing temperature.

(2) *Temperature Amended Criterion.* One of the key steps in the process of SA is to determine the update function of temperature; the function is used to continuously reduce the temperature value, when its temperature is reduced to approximately zero, the final solution is considered as the global optimal solution. The update function is  $t_{k+1} = \alpha t_k$ ,  $k \geq 0$ ,  $0 < \alpha < 1$ ; the nearer  $\alpha$  is to 1, the slower the temperature decreases.

3.2.3. *Termination.* Commonly, the algorithm terminates when either a maximum number of generations have been produced, or a satisfactory fitness level has been reached for the population. In this paper, the termination condition is that the fitness has reached a plateau such that successive  $M$  iterations no longer produce better results.

### 3.2.4. Algorithm Flow

*Step 1.* Set the initial parameters: coordinates of the DPs and the candidate MCs, demands and returns of the DPs, the maximum capacity of the vehicle  $g$ , the population size  $N$ , evolution terminate generation  $M$ , crossover probability  $p_c$ , mutation probability  $p_m$ , temperature of the cooling coefficient  $\alpha$ , the initial annealing temperature  $T_0$ , and so on.

*Step 2.* Calculate the fitness value of an individual. If the parent optimal solution and offspring optimal solution are equal during continuous  $M$  generations, the algorithm stops

and outputs the current optimal solution; otherwise, go to the next step.

*Step 3.* Perform individual selection, crossover, and mutation operations, generate new population, and calculate the fitness value.

*Step 4.* If  $f_i < f_j$  ( $j > i$ ), accept the new individual; otherwise, accept the new individual with the probability  $p = \exp(-\Delta_f/t)$ ;

*Step 5.* Update the annealing temperature and return to Step 2.

The pseudocodes of HGSA are shown in Pseudocode 1.

## 4. Computational Experiments and Results Analysis

4.1. *An Example.* An example is used to illustrate the proposed heuristic method. The data of Gaskell 67-29×5 come from the LRP database at University of Aveiro [24]. Gaskell 67 is the name of this instance; 29×5 means there are 29 DPs and 5 candidate MCs. The coordinate of all nodes and the demands of DPs are given by the database. To facilitate the calculation, the daily demands of DPs are set as 1/25 of corresponding demands in the LRP database. The other data are as follows: the inventory holding cost per unit of merchandise per year  $h = 5$ ; the vehicle capacity  $g = 500$ ; the delivering cost per unit distance  $l = 1$ ; the handling cost per unit product at MC<sub>*r*</sub>  $c_r = 4$ ; fixed cost of dispatching vehicles per time at MC<sub>*r*</sub>  $e_r = 18$ ; the conversion constant  $\lambda = 300$ ; repackaging cost of unit returned merchandise  $p = 3$ ;  $q_i$ ,  $b_r$ ,  $f_r$  are uniformly generated from  $U[1, 5]$ ,  $U[6, 10]$ ,  $U[16, 20]$ .

According to the experience of literatures [25, 26], the related parameters of the HGSA are set as follows: the population size  $N = 20$ ; crossover probability  $p_c = 0.8$ ; mutation probability  $p_m = 0.001$ ; evolution terminate generation  $M = 300$ ; initial temperature  $T_0 = 100$ ; temperature cooling coefficient  $\alpha = 0.9$ .

Based on Matlab 6.5 platform, we programmed the HGSA and then run it 30 times on a computer (CPU: Intel Core 2 Duo P8400 @ 2.26 GHz 2.27 GHz; RAM: 3 GB DDR; OS: Windows Vista). One of the minimum values of objective function is 1404100; the individual is encoded as {5, 3, 34, 23, 21, 33, 11, 32, 4, 12, 28, 19, 29, 14, 25, 13, 22, 27, 7, 15, 26, 8, 6, 18, 30, 2, 1, 10, 20, 24, 31, 9, 16, 17}, and  $N_r = \{56, 0, 46, 94, 0\}$ ,  $Q_r^v = \{333, 231, 444, 300, 409, 368\}$ . Figure 2 shows the trends of optimal objective function values along with the evolution generations. Table 1 shows the solution.

For comparison, GA is programmed by Matlab 6.5 as well, and the instance Gaskell 67 was run 30 times on the same computer. The optimal objective function values of these two algorithms are shown in Table 2, the CPU time for calculation is shown in Table 3.

Figure 3 shows the trends of optimal objective function value along with the evolution generations by GA.

The fluctuation curves of optimal objective function values in 30 times are shown in Figures 4 and 5, respectively.

```

Procedure: HGSAA for LIRP
Input: coordinates of nodes, demands and returns of DPs, MC parameters, vehicle capacity,
      HGSAA parameters
Output: the best solution (include routes, MCs locations, order times and order size)
Begin
Take pop
  for i = 1 to pop_size
    Calculate individual fitness value  $f_i$ 
  end
   $f_{best} = \max\{f_i\}$ 
  n = 1
  k = 1
while n ≤ 300
  select operator
    if random ≤  $p_c$  then
      crossover operator
    end
    if random ≤  $p_m$  then
      mutation operator
    end
  newpop
    for i = 1 to newpop_size
      Calculate individual fitness value  $f_i$ 
    end
     $f_{newbest} = \max\{f_i\}$ 
     $f_{worst} = \min\{f_i\}$ 
    if  $f_{newbest} > f_{best}$  then
      n = 1
       $f_{best} = f_{newbest}$ 
      pop(best,:) = newpop(newbest,:)
    elseif  $f_{newbest} == f_{best}$ 
      n = n + 1
    else
      n = 1
       $\Delta = \frac{1}{f_{newbest}} - \frac{1}{f_{best}}$ 
       $z = e^{(-\Delta/t)}$ 
      if z > random then
         $f_{best} = f_{newbest}$ 
        pop(best,:) = newpop(newbest,:)
      end
      T =  $\alpha T$ 
    end
    newpop(worst,:) = pop(best,:)
    pop = newpop
    k = k + 1
  end
output the best solution
end

```

PSEUDOCODE 1: Pseudocode of the proposed HGSAA.

TABLE 1: Solution of Gaskell 67-29×5.

MC	Routing number	Routing	Order times	Order quantity
1	1	$i_{10}-i_{20}-i_{24}-i_{31}-i_9$	56	333
	2	$i_{16}-i_{17}$	56	231
3	3	$i_{34}-i_{23}-i_{21}-i_{33}-i_{11}-i_{32}$	46	444
	4	$i_{12}-i_{28}-i_{19}-i_{29}-i_{14}-i_{25}-i_{13}-i_{22}-i_{27}$	94	300
4	5	$i_7-i_{15}$	94	409
	6	$i_{26}-i_8-i_6-i_{18}-i_{30}$	94	368

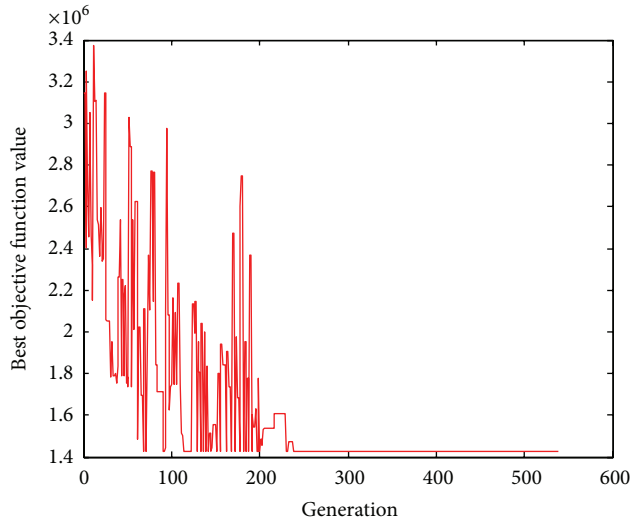


FIGURE 2: Trends of optimal objective function value by HGSA.

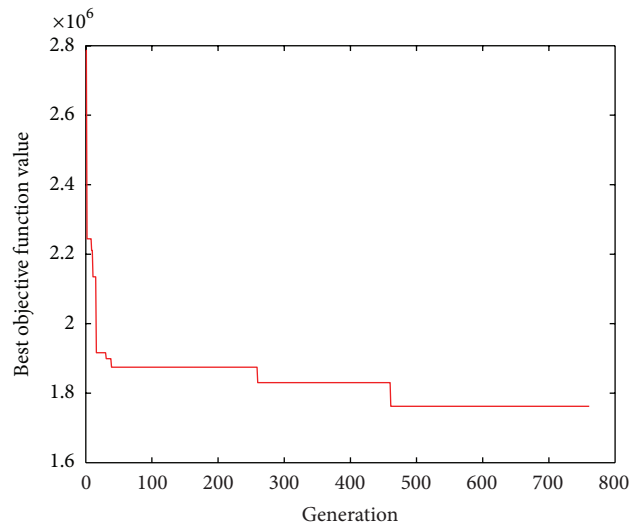


FIGURE 3: Trends of optimal objective function value by GA.

Figures 2 and 3 show that HGSA can converge to the optimal solution more quickly than GA. Moreover, HGSA has better stability than GA, which can easily be found from Tables 2 and 3 and Figures 4 and 5.

**4.2. Extended Experiments.** In this section, a series of experiment is given to show that HGSA is more efficient and stable than GA. Similarly as Section 4.1, all the experiments in this section come from LRP database of the University of Aveiro [24]. In order to ensure the demands of DPs are not more than the vehicle capacity, we need to enumerate some instances. In this study, the daily demands are set as 1/15 of corresponding demands of Gaskell 67-22 $\times$ 5.

Results of numerical example in Section 4.1 show that the related parameters of HGSA in Gaskell 67-22 $\times$ 5 are reasonable. Thus, we employ these parameters in the remainder of this section. Each instance was calculated 30 times by

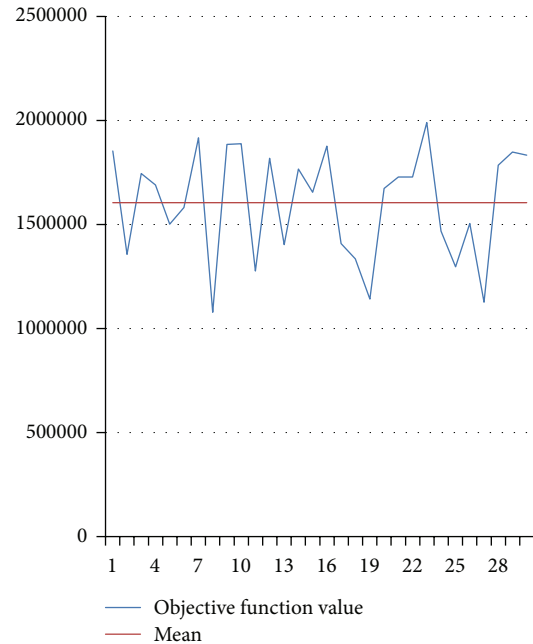


FIGURE 4: The fluctuation curve of optimal objective function value by HGSA.

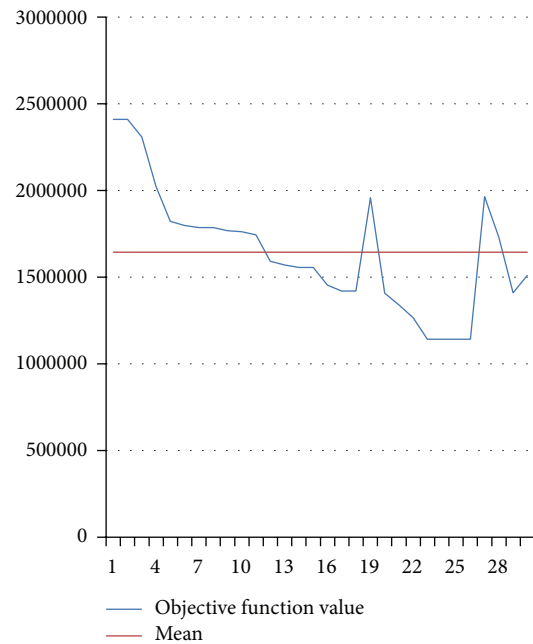


FIGURE 5: The fluctuation curve of optimal objective function value by GA.

HGSA and GA, respectively; the results are shown in Tables 4 and 5.

Table 4 shows that HGSA can obtain better objective function value than GA. Table 5 shows that HGSA takes less time to achieve the optimal solution than GA. Results of Tables 4 and 5 show that HGSA is more stable than GA.

TABLE 2: Statistical results of optimal objective function value of two algorithms.

	Maximum	Minimum	Mean	Standard deviation	Coefficient of variation
HGSAA	1989900.00	1078700.00	1605543.33	262462.64	0.16
GA	2410900.00	1142000.00	1644678.00	353993.10	0.22

TABLE 3: Statistics results of CPU time for calculation of two algorithms.

	Maximum	Minimum	Mean	Standard deviation	Coefficient of variation
HGSAA	5.93	3.35	4.68	0.80	0.17
GA	7.35	3.63	4.93	1.18	0.24

## 5. Conclusion and Future Research

Under the e-commerce environment, customers have a higher return rate. At the same time, the returned goods have generally no quality defect and with great integrity. Just after a simple repackaging process, the returned goods can reenter the sales channels, which put forward high requirements to the logistics system that support the operation of e-commerce. This study handles the above interesting problem and provides an effective heuristic. The main contributions are as follows.

- (1) In reality, the cost of processing returned merchandises is produced considering the customers are not satisfied with products and maybe return them. We firstly design a LIRP model to minimize the total cost produced by both forward and reverse logistics networks. It is very useful to help managers make the right decision under e-supply chain environment.
- (2) An integration LIRP model with returns is an NP-hard problem and very hard to be solved by analytical method. So, a heuristic algorithm named HGSAA is designed by integrating GA with SA.
- (3) Results of experimental data show that HGSAA outperforms GA on computing time, optimal solution, and computing stability. HGSAA is a good candidate to solve the proposed LIRP model effectively.

However, some extensions should be considered in further work. Considering the variety of the types of products and service vehicles, the multiple products and multiple vehicles type model should be established. In reality, decision makers are always in front of imprecise and vague operational conditions [27]. Uncertainties have been tackled in a lot of ways and fuzzy set theory has a long history for handle imprecise values [28]. Considering the fuzzy demand of customs or related fuzzy costs, more practical LIRP model should be developed. Moreover, differential evolution algorithms (DEs) have turned out to be one of the best evolutionary algorithms in a variety of fields [29]. In the future, we may use an improved DE to find better solutions for the LIRPs. The integration research and practice of the management of e-commerce logistics system can be constantly improved.

## Notations

### Sets

- R**: Set of candidate MC  
**S**: Set of DPs  
**S**<sup>+</sup>: Set of candidate MCs and DPs; that is,  
**S**<sup>+</sup> = **S** ∪ **R**  
**V**: Set of routes are available for the routing from the MCs to DPs.

### Constants

- $a_r$ : Fixed (annual) administrative and operational cost of MC<sub>r</sub>  
 $b_r$ : The unit of product transportation cost from plant to MC<sub>r</sub>  
 $c_r$ : The handling costs per unit product at MC<sub>r</sub>  
 $d_i$ : Mean (daily) demand at DP<sub>i</sub>  
 $e_r$ : Fixed cost of dispatching vehicles per time at MC<sub>r</sub>  
 $f_r$ : Fixed cost of placing an order at MC<sub>r</sub>  
 $g$ : The vehicle capacity  
 $h$ : The inventory holding cost per unit of merchandise per year  
 $l$ : The delivering cost per unit distance  
 $p$ : Repackaging cost of unit returned merchandise at MC  
 $q_i$ : The quantity of merchandises returned by DP<sub>i</sub> per day  
 $s_{ij}$ : The distance from depot  $i$  to depot  $j$   
 $\lambda$ : Working days per year.

### Decision Variables

- $N_r$ : Optimal order times at MC<sub>r</sub>  
 $Q_r^v$ : Optimal order size on routing  $v$  for MC<sub>r</sub>  
 $X_{ijr}^v$ : 1, if depot  $j$  is from depot  $i$  served by a MC<sub>r</sub> on routing  $v$ , and 0 otherwise  
 $Y_{ir}^v$ : 1, if DP<sub>i</sub> is assigned to MC<sub>r</sub> on routing  $v$ , and 0 otherwise  
 $Z_r$ : 1, if candidate MC<sub>r</sub> is selected as a MC location, and 0 otherwise.



TABLE 4: Optimal objective function values of two algorithms.

Instance name	Algorithm	Maximum	Minimum	Mean	Standard deviation	Coefficient of variation
Perl 183-12×2	HGSAA	740420	157810	502684	176218.7	0.350555
	GA	1322000	219800	660282	234416.0	0.355024
Gaskell 67-22×5	HGSAA	1718000	1071600	1432110	213118.8	0.148815
	GA	3146500	1506200	2170660	553894.4	0.255173
Gaskell 67-36×5	HGSAA	3635500	1877000	2879510	561628.4	0.195043
	GA	3836900	2219400	2886460	582830.7	0.201919
Perl 183-55×15	HGSAA	4120700	3606800	3985110	152566.1	0.038284
	GA	4307000	3755100	4090590	185978.4	0.045465
Christofides 69-75×10	HGSAA	5562400	4290900	4826600	394532.4	0.081741
	GA	6359300	4859800	5418878	525869.4	0.097044
Perl 83-85×7	HGSAA	6529200	5050500	5693300	531113.4	0.093287
	GA	7057200	5545800	6283210	428004.8	0.068119
Christofides 69-100×10	HGSAA	5978900	5074600	5592260	332403.7	0.05944
	GA	6177500	5211900	5792190	354180.4	0.061148

TABLE 5: CPU time (seconds) for calculation of two algorithms.

Instance name	Algorithm	Maximum	Minimum	Mean	Standard deviation	Coefficient of variation
Perl 183-12×2	HGSAA	0.8424	0.4992	0.65156	0.085708	0.131543
	GA	0.9672	0.5304	0.6696	0.102803	0.153529
Gaskell 67-22×5	HGSAA	1.716	1.092	1.37436	0.169531	0.123353
	GA	2.1216	1.17	1.55688	0.272075	0.174756
Gaskell 67-36×5	HGSAA	3.7596	2.2776	3.2058	0.501429	0.156413
	GA	4.8984	2.5584	3.81108	0.760696	0.199601
Perl 183-55×15	HGSAA	12.0121	8.5957	9.91249	0.999262	0.100808
	GA	12.6517	9.8125	10.91837	0.846487	0.077529
Christofides 69-75×10	HGSAA	13.3225	10.1713	11.74674	1.162717	0.098982
	GA	13.7437	10.2805	12.30382	1.370948	0.111425
Perl 83-85×7	HGSAA	21.6373	17.0509	18.94194	1.305246	0.068908
	GA	24.3518	17.8621	21.04736	2.279007	0.10828
Christofides 69-100×10	HGSAA	33.2906	30.1859	31.36279	0.93519	0.029818
	GA	33.5558	30.6386	32.21258	0.962075	0.029866

## Acknowledgment

This work was supported by the National Natural Science Foundation of China (nos. 70871050; 71171093; 71371080; 71131004; and 70801030).

## References

- [1] "Ecommerce sales topped \$1 trillion for the first time in 2012," <http://www.emarketer.com/Article/Ecommerce-Sales-Topped-1-Trillion-First-Time-2012/1009649>.
- [2] H. Meyer, "Many happy returns," *Journal of Business Strategy*, vol. 20, no. 4, pp. 27–31, 1999.
- [3] C. R. Gentry, "Reducing the cost of returns," *Chain Store Age*, vol. 75, no. 10, pp. 124–126, 1999.
- [4] J. Yang and Z. Fei, "Statistical filtering based broadcast protocol for vehicular networks," in *Proceedings of the 20th International Conference on Computer Communications and Networks (ICCCN '11)*, pp. 1–6, Hawaii, USA, July 2011.
- [5] J. Yang and Z. Fei, "HDAR: hole detection and adaptive geographic routing for ad hoc networks," in *Proceedings of the 19th International Conference on Computer Communications and Networks (ICCCN '10)*, pp. 1–6, Zurich, Switzerland, August 2010.
- [6] C. Watson-Gandy and P. Dohrn, "Depot location with van salesmen: a practical approach," *Omega*, vol. 1, no. 3, pp. 321–329, 1973.
- [7] G. Nagy and S. Salhi, "Location-routing: issues, models and methods," *European Journal of Operational Research*, vol. 177, no. 2, pp. 649–672, 2007.
- [8] N. H. Moin and S. Salhi, "Inventory routing problems: a logistical overview," *Journal of the Operational Research Society*, vol. 58, no. 9, pp. 1185–1194, 2007.
- [9] S. C. Liu and S. B. Lee, "A two-phase heuristic method for the multi-depot location routing problem taking inventory control decisions into consideration," *International Journal of Advanced Manufacturing Technology*, vol. 22, no. 11-12, pp. 941–950, 2003.
- [10] S. C. Liu and C. C. Lin, "A heuristic method for the combined location routing and inventory problem," *International Journal*

- of *Advanced Manufacturing Technology*, vol. 26, no. 4, pp. 372–381, 2005.
- [11] Z.-J. Max Shen and L. Qi, “Incorporating inventory and routing costs in strategic location models,” *European Journal of Operational Research*, vol. 179, no. 2, pp. 372–389, 2007.
- [12] A. Ahmadi Javid and N. Azad, “Incorporating location, routing and inventory decisions in supply chain network design,” *Transportation Research E*, vol. 46, no. 5, pp. 582–597, 2010.
- [13] A. Ahmadi-Javid and A. H. Seddighi, “A location-routing-inventory model for designing multisource distribution networks,” *Engineering Optimization*, vol. 43, no. 10, pp. 1–19, 2012.
- [14] M. Fleischmann, P. Beullens, J. M. Bloemhof-Ruwaard, and L. N. van Wassenhove, “The impact of product recovery on logistics network design,” *Production and Operations Management*, vol. 10, no. 2, pp. 156–173, 2001.
- [15] V. Jayaraman, R. A. Patterson, and E. Rolland, “The design of reverse distribution networks: models and solution procedures,” *European Journal of Operational Research*, vol. 150, no. 1, pp. 128–149, 2003.
- [16] K. Lieckens and N. Vandaele, “Reverse logistics network design with stochastic lead times,” *Computers and Operations Research*, vol. 34, no. 2, pp. 395–416, 2007.
- [17] K. Sahyouni, R. C. Savaskan, and M. S. Daskin, “A facility location model for bidirectional flows,” *Transportation Science*, vol. 41, no. 4, pp. 484–499, 2007.
- [18] G. Easwaran and H. Üster, “Tabu search and Benders decomposition approaches for a capacitated closed-loop supply chain network design problem,” *Transportation Science*, vol. 43, no. 3, pp. 301–320, 2009.
- [19] S. K. Srivastava, “Network design for reverse logistics,” *Omega*, vol. 36, no. 4, pp. 535–548, 2008.
- [20] D. Vlachos and R. Dekker, “Return handling options and order quantities for single period products,” *European Journal of Operational Research*, vol. 151, no. 1, pp. 38–52, 2003.
- [21] S. Jakobs, “On genetic algorithms for the packing of polygons,” *European Journal of Operational Research*, vol. 88, no. 1, pp. 165–181, 1996.
- [22] M. Gen, R. Cheng, and L. Lin, *Network Models and Optimization: Multi-Objective Genetic Algorithm Approach*, Springer, Heidelberg, Germany, 2008.
- [23] T. W. Leung, C. H. Yung, and M. D. Troutt, “Applications of genetic search and simulated annealing to the two-dimensional non-guillotine cutting stock problem,” *Computers and Industrial Engineering*, vol. 40, no. 3, pp. 201–214, 2001.
- [24] “Location-Routing Problems (LRP),” [http://sweet.ua.pt/~iscf143/\\_private/SergioBarretoHomePage.htm](http://sweet.ua.pt/~iscf143/_private/SergioBarretoHomePage.htm).
- [25] H. H. Orkcu, “Subset selection in multiple linear regression models: a hybrid of genetic and simulated annealing algorithms,” *Applied Mathematics and Computation*, vol. 129, no. 23, pp. 11018–11028, 2013.
- [26] Y. Xu, R. Qu, and R. F. Li, “A simulated annealing based genetic local search algorithm for multi-objective multicast routing problems,” *Annals of Operations Research*, vol. 206, no. 1, pp. 527–555, 2013.
- [27] L. Wang, Q. L. Fu, C. G. Lee, and Y. R. Zeng, “Model and algorithm of fuzzy joint replenishment problem under credibility measure on fuzzy goal,” *Knowledge-Based Systems*, vol. 39, pp. 57–66, 2013.
- [28] L. Wang, C. X. Dun, C. G. Lee, Q. L. Fu, and Y. R. Zeng, “Model and algorithm for fuzzy joint replenishment and delivery scheduling without explicit membership function,” *International Journal of Advanced Manufacturing Technology*, vol. 66, no. 9–12, pp. 1907–1920, 2013.
- [29] L. Wang, H. Qu, Y. H. Li, and J. He, “Modeling and optimization of stochastic joint replenishment and delivery scheduling problem with uncertain costs,” *Discrete Dynamics in Nature and Society*, vol. 2013, Article ID 657465, 12 pages, 2013.