Research Article

Optimization on Production-Inventory Problem with Multistage and Varying Demand

Duan Gang,1 Chen Li,2 Li Yin-Zhen,1,3 Song Jie-Yan,1 and Akhtar Tanweer1

1 School of Traffic and Transportation, Lanzhou Jiaotong University, Lanzhou 730070, China
2 Department of Mathematics, Lanzhou City University, Lanzhou 730070, China
3 Northwest Traffic Economy Research Center, Lanzhou Jiaotong University, Lanzhou 730070, China

Correspondence should be addressed to Duan Gang, dg_77@163.com

Received 25 June 2012; Revised 9 October 2012; Accepted 16 October 2012

Academic Editor: Yuri Sotskov

Copyright © 2012 Duan Gang 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.

This paper addresses production-inventory problem for the manufacturer by explicitly taking into account multistage and varying demand. A nonlinear hybrid integer constrained optimization is modeled to minimize the total cost including setup cost and holding cost in the planning horizon. A genetic algorithm is developed for the problem. A series of computational experiments with different sizes is used to demonstrate the efficiency and universality of the genetic algorithm in terms of the running time and solution quality. At last the combination of crossover probability and mutation probability is tested for all problems and a law is found for large size.

1. Introduction

Production-inventory control plays a vital role in the management of manufacturing enterprises. During production process we often hope that the volume of a given product is just enough to satisfy customers’ demand without overextending the production line and manufacturing too many. The redundant inventory will cut down our net profit. On the other hand, lacking supply will also make a heavy loss when a big order is placed. So there must be an optimal production quantity.

Many researchers are interested in this problem and it has been investigated from various perspectives. Goyal and Giri [1] develop two different models for the production-inventory problem, in which the demand, production, and deterioration rates of a product are assumed to vary with time over an infinite planning horizon. Shortages of a cycle...
are allowed to be backlogged partially. Huang [2] proposes an optimal integrated vendor-buyer inventory policy for flawed items in a just-in-time manufacturing environment. The production process is assumed to deteriorate during processing and produces a certain number of defective items. The objective is to minimize the total joint annual costs incurred by the vendor and the buyer. Hill [3] considers the problem of a vendor supplying a product to a buyer with the product manufactured in batches at a finite rate. And the structure of the globally-optimal solution is derived. Gayon et al. [4] formulate the joint production-control and inventory-allocation problem as a Markov decision process and characterize the structure of the optimal policy. They draw a conclusion that the optimal inventory-allocation policy is a state-dependent multilevel rationing policy, with the rationing level for each class nondecreasing in the number of announced orders. An adaptive control approach with a feedback is applied to track the inventory levels toward their goal levels by Alshamrani [5,6].

Considering uncertainty, Doğru et al. [7] develop a stochastic program to allow preferential component allocation for minimizing total inventory cost in assemble-to-order inventory systems with identical component lead times. Zijm and Houtum [8] compare the optimal base stock policy under stochastic circumstance with an MRP system in terms of cost effectiveness given a predefined target customer service level. And they also take stochastic lead times into consideration in a multistage production-to-order system. According to setting safety lead times, it leads to similar decomposition structures as those derived for multistage inventory systems. Kleywegt et al. [9] formulate the inventory routing problem as a Markov decision process. Gupta and Wang [10] investigate credit terms in inventory problem with random demand. They prove that the structure of the optimal inventory policy is not affected by credit terms under a discrete time of the retailer’s operations, although the value of the optimal policy parameter is.

Ouyang et al. [11] address the same problem based on a deterministic model by fuzzifying the rate of interest charges, the rate of interest earned, and the deterioration rate into the triangular fuzzy number. Hsieh [12] proposes two fuzzy inventory models with fuzzy parameters for crisp order quantity, or for fuzzy order quantity under decision maker’s preference. He demonstrates that the optimal order quantity or the optimal fuzzy order quantity of the two models are the real numbers. The optimal solutions can be specified to meet classical production inventory models under a given condition. References [13–16] also focus on the fuzzy optimal inventory problem.

Chao et al. [17] research operational decisions in dynamic inventory management are correlated with and constrained by financial flows of the firm. A general framework for incorporating financial states of an organization in multiperiod inventory models with lost sales is proposed. Hopp and Xu [18] propose a static approximation of dynamic demand substitution behavior based on a fluid network model and a service-inventory mapping. The interdependent inventory/service, price, and product assortment decisions in noncompetitive and competitive scenarios with demand substitution are analyzed. Olsen and Parker [19] investigate consumer behavior of potentially leaving the firm’s market when he encounters an inventory stockout at a retailer under the stochastic demand distribution in a time-dynamic context. A single firm and a duopoly case are modeled, respectively.

Many researches of perishable inventory problem are investigated. Broekmeulen and Donselaar [20] take into account of the age of the inventory for perishable products at retailers. The new policy they conclude leads to substantial cost reductions compared with a base policy that does not take into account the age of inventories. Duan et al. [21] propose two kinds of inventory models, with and without backlogging, respectively, for perishable items.
At last they draw a conclusion that the proposed model is a generalization of present ones. Gumasta et al. [22] consider the transportation of perishable goods from distributor to retailer. They simultaneously maximize the revenue and minimize transportation and inventory cost so as to maximize the net profit. Other related researches under stochastic circumstance can be referenced in [23, 24]. And Markov model for an inventory system with perishable items can be referenced in [25–27] and so on.

Some researchers are interested in discussing backorder problem in inventory. Yao and Chiang [28] take the storing cost $a$, backorder cost $b$, cost of placing an order $c$, total demand $r$, order quantity $q$, and shortage quantity as the triangular fuzzy numbers and use the signed distance method to defuzzify. Güler et al. [29] consider bayesian updating of demand and backorder distributions in a partial backorder newsvendor model. In order to model the relationship between risks and risk propagation in supply chain, Shin et al. [30] apply bayesian belief network to develop alternative backorder replenishment plan to minimize the total replenishment cost and expected risk cost. Hu et al. [31] and Shah and Soni [32] assume production elapsed time and demand to be a fuzzy random variable with backorder, respectively. We can also refer to [33–38] about backorder aspect.

The method for assembly line balancing problem is very similar to the inventory problem. Sotskov et al. [39, 40] minimize the number of stations $m$ for processing $n$ partially ordered operations within the given cycle time. They derive necessary and sufficient conditions when optimality of the line balance $b$ is stable with respect to sufficiently small variations of the operation times. Özcan and Toklu [41–43] and Purnomo et al. [44] research two-sided assembly lines balancing problem. Kellegöz and Toklu [45] address a branch and bound algorithm to solve the problem. Compared with the existing algorithm, the proposed algorithm outperforms it in terms of both CPU times and quality of feasible solutions found.

The traditional deterministic inventory models are based on the assumption that the demand rate is constant and order cycle is inflexible. As we all know the total cost is the integral of the production-inventory function because the production time is the same and the maximum inventory in every stage is also equal. But actually there is more than one customer. Demand cycles of these customers are different and the demand rate of every customer is also varying. This is multistage and varying demand production-inventory problem without stockout. In the planning horizon, which is the sum of all stages, what we face to decide is also when and how many quantity to produce in order to minimize the total cost subject to meeting every stage demand. Under the circumstance the traditional unconstrained optimization method is not suitable for the issue although it is still linear. Because according to every demand and its cycle there are different maximum inventory in every stage under different production schedules which means production frequency and production time. We have to define the maximum inventory in every stage for the sake of obtaining the objective function. And this definition relation is only determined by the constraint. So constrained optimization method will be applied to solve the problem. Motivated by the above case we model a hybrid integer programming to characterize the problem and solve it by using genetic algorithm.

The remainder of the paper is organized as follows. Section 2 is dedicated to the description of the modeling approach. Section 3 presents a genetic algorithm to solve the problem. Section 4 reports our computational results on test instance. Finally, conclusions are drawn in Section 5.
2. Production-Inventory Model

2.1. Problem Description

In multistage production-inventory system, every stage extent is versatile, demand rate is also different and production rate is a constant and is more than demand rate in every stage. The production-inventory status is drawn in Figure 1 according to the production rate and demand rate in every stage. The planning horizon is made up of five stages. $T_i$ denotes the extent of stage $i$. There are twice production in Figure 1. The production mode is as shown in Figure 1(a). The first production begins from Stage 1 and ends in Stage 2 lasting $t_1$ time long. And at time $t_1$ the inventory reaches maximum. The maximum inventory in the first production is consumed in the end of Stage 3. We call it a production cycle from Stage 1 to Stage 3. Stage 3 is called the end of one production cycle. The second production cycle includes Stage 4 and Stage 5 lasting $t_2$ time long. Figure 1(b) depicts another case of twice production. The first production cycle is from Stage 1 to Stage 4 and the second production cycle is in Stage 5 alone.

In fact there are four production schemes under the circumstance of twice production. Except the above two schemes mentioned, one is that the first production is in Stage 1 alone and the second production starts from Stage 2 to Stage 5. The other is that the first production begins from Stage 1 to Stage 3 and the second production starts from Stage 4 to Stage 5.
One extreme case is five times production in planning horizon, that is, production occurs in every stage and the quantity as inventory in every stage only satisfies the current stage demand. The other extreme case is only once production in planning horizon. The best production times and quantity are specified by the total cost which bears setup cost of production and carrying cost held in inventory. And we will find the best production scheme under all production times.

2.2. Model

2.2.1. Denotations Definition

First we define some denotations in our model to express the above conception.

Parameters are defined as follows.

\( P \): Production rate.

\( n \): The extent of stages.

\( m \): Production times, \( m = 1, 2, \ldots, n \).

\( T_i \): Demand extent of stage \( i, i = 1, 2, \ldots, n \).

\( R_i \): Demand rate in stage \( i, i = 1, 2, \ldots, n \).

\( K \): Setup cost of production once.

\( h \): Holding cost per unit volume per unit time held in inventory.

\( T \): Planning horizon. Obviously \( T \) equals to the sum of \( T_i \), that is, \( T = \sum_{i=1}^{n} T_i \).

Variables are defined as follows:

\[ x_i = \begin{cases} 
1, & \text{if the production begins from stage } i, \quad i = 1, 2, \ldots, n. \\
0, & \text{otherwise},
\end{cases} \tag{2.1} \]

\( t_{ik} \): The production time of one production cycle which begins from stage \( i \) and ends in stage \( k \), \( 1 \leq i \leq k \leq n \)

\[ v_{ijk} = \begin{cases} 
1, & \text{if } t_{ik} > 0, \\
0, & \text{otherwise},
\end{cases} \quad 1 \leq i \leq k \leq n. \tag{2.2} \]

\( Q_i \): The inventory level at the beginning of stage \( i, i = 1, 2, \ldots, n \)

\( Q_{ijk} \): The maximum inventory level at time \( t_{ik} \) in stage \( j \), \( 1 \leq i \leq j \leq k \leq n \). The variable maximum inventory \( Q_{ijk} \) is positive if and only if \( t_{ik} \) belongs to the interval \((\sum_{l=i}^{j-1} T_l, \sum_{l=i}^{j} T_l)\)

\[ w_{ijk} = \begin{cases} 
1, & \text{if } Q_{ijk} > 0, \\
0, & \text{otherwise},
\end{cases} \quad 1 \leq i \leq j \leq k \leq n. \tag{2.3} \]
2.2.2. Constraints

(1) Because the stockout is not permitted, the total quantity should be equal to the sum of demands in the planning horizon. And the quantity in every production in a certain stage interval should satisfy the demand in the same stage interval:

\[ P \sum_{i=1}^{n} \sum_{k \geq i} t_{ik} = \sum_{i=1}^{n} R_i T_i, \quad (2.4) \]

\[ Pt_{ik} = v_{ik} \sum_{l=i}^{k} R_l T_l, \quad 1 \leq i \leq k \leq n. \quad (2.5) \]

(2) Production times in the planning horizon are calculated by the following formula:

\[ \sum_{i=1}^{n} x_i = m \quad (2.6) \]

(3) Production time should be less than demand time:

\[ t_{ik} \leq \sum_{l=i}^{k} T_l, \quad 1 \leq i \leq k \leq n. \quad (2.7) \]

(4) The logic variables should satisfy the following:

\[ v_{ik} = x_i \left(1 - \Phi \left(\sum_{l=i+1}^{k} x_l\right)\right) x_{k+1}, \quad 1 \leq i \leq k \leq n. \quad (2.8) \]

The binary function \( \Phi(a) \) is defined as

\[ \Phi(a) = \begin{cases} 1, & \text{if } a > 0, \\ 0, & \text{otherwise}. \end{cases} \quad (2.9) \]

Formula (2.8) means \( v_{ik} = 1 \) if and only if \( x_i = 1, x_{k+1} = 1 \) and every \( x_l = 0, l = i, \ldots, k \) are met simultaneously.

(5) The inventory level \( Q_i \) at the beginning of stage \( i \) is derived by the following:

\[ Q_i = \max_{l \leq j \leq k} \left\{ w_{ijk} \left( \Psi(j-i) \sum_{s=1}^{i-1} (P - R_s) T_s + \Psi(i-(j+1)) \sum_{s=i}^{k} R_s T_s \right) \right\}, \quad (2.10) \]

where \( i \geq 2, 1 \leq l \leq i \leq k \leq n \) and

\[ \Psi(a) = \begin{cases} 1, & \text{if } a \geq 0, \\ 0, & \text{otherwise}. \end{cases} \quad (2.11) \]
It is the fact that a new production begins in stage $i$ when $Q_i$ becomes zero. In other cases $Q_i$ is always more than zero. When the production time lasts $t_{ik}$ time long in a production cycle, the first item $\Psi(j-i)\sum_{s=i}^{j-1}(P-R_s)T_s$ in formula (2.10) indicates the inventory at the beginning of stage $j$ which is before stage $i$. And the second item $\Psi(i-(j+1))\sum_{s=i}^{j}R_sT_s$ in formula (2.10) indicates the inventory at the beginning of stage $j$ which is after stage $i$. For different $j$ and $k$, the variable $w_{ijk}$ may be zero. Only when $w_{ijk}$ equals to one, $Q_i$ is the right inventory at the beginning of stage $i$. So we solve it by maximization.

(6) The relationship between the logic variables $w_{ijk}$ and $v_{ik}$ and continuous variable $t_{ik}$ is characterized as follows:

$$w_{ijk} = v_{ik} \Phi \left( t_{ik} - \sum_{l=i}^{j-1} T_l \right) \Psi \left( \sum_{l=i}^{j} T_l - t_{ik} \right), \quad 1 \leq i \leq j \leq k \leq n. \tag{2.12}$$

(7) The maximum inventory level $Q_{ijk}$ is deduced according to the following:

$$Q_{ijk} = w_{ijk} \left( Q_j + \left( t_{ik} - \sum_{l=i}^{j-1} T_l \right)(P - R_j) \right), \quad 1 \leq i \leq j \leq k \leq n, \tag{2.13}$$

(8) when $w_{ijk} = 1$ and production time is $t_{ik}$, the item $(t_{ik} - \sum_{l=i}^{j-1} T_l)(P - R_j)$ in formula (2.13) means the new inventory level during stage $j$. So $Q_{ijk}$ is the sum of the inventory $Q_j$ at the beginning of stage $j$ and $(t_{ik} - \sum_{l=i}^{j-1} T_l)(P - R_j)$. At last the variables are defined as:

$$x_{i}, t_{ik}, Q_{i}, Q_{ijk} \geq 0, \quad v_{ik}, w_{ijk} \in \{0, 1\}, \quad 1 \leq i \leq j \leq k \leq n. \tag{2.14}$$

The initial conditions are $t_{1} = 0$ and $S_1 = S_{n+1} = 0$.

2.2.3. Objective

The sum cost holds setup cost of production and carrying cost held in inventory. So the objective is to minimize the sum cost.

When producing $m$ times in the planning horizon the total setup cost is $mK$ where $K$ is setup cost producing once. In every production cycle we suppose that production time is $t_{ik}$ and $w_{ijk} = 1$, $1 \leq i \leq j \leq k \leq n$. Then we employ formula (2.4) to formula (2.13) to get the values of $Q_i$ and $Q_{ijk}$, $1 \leq i \leq j \leq k \leq n$.

$Q_i$ and $Q_{i+1}$ are the inventory levels at the beginning of stage $i$ and stage $i + 1$, respectively. Because the inventory figures from stage $i$ to stage $j - 1$ are all trapezoids. The area of a trapezoid is calculated by $(Q_i + Q_{i+1})T/2$. In fact the inventory figure of stage $i$ is a triangle for the inventory level $Q_i$ is zero such as inventory figure of Stage 1 in Figure 1(b).
We may take a triangle as a degenerate trapezoid in which one edge length is zero. So we use uniform area formula. From stage \( i \) to stage \( j - 1 \) when \( j \) is more than \( i \), the holding cost is

\[
\frac{h}{2} \sum_{l=i}^{j-1} (Q_l + Q_{l+1})T_l, \tag{2.15}
\]

where \( h \) is holding cost per unit volume per unit time held in inventory.

In stage \( j \) where production stops the inventory figure is made up of two trapezoids such as the inventory figure of the Stage 3 in Figure 1(b). So the holding cost in stage \( j \) is

\[
\frac{h}{2} \left( (Q_j + Q_{ijk}) \left( t_{ik} - \sum_{l=i}^{j-1} T_l \right) + (Q_{ijk} + Q_{j+1}) \left( j \sum_{l=i}^{j} T_l - t_{ik} \right) \right). \tag{2.16}
\]

The case from stage \( j + 1 \) to stage \( k \) is similar to stage \( i \) to stage \( j - 1 \). Therefore, the holding cost from stage \( j + 1 \) to stage \( k \) is

\[
\frac{h}{2} \sum_{l=j+1}^{k} (Q_l + Q_{l+1})T_l. \tag{2.17}
\]

So given production times \( m \), we could solve the minimal average total cost under every possible production mode in the planning horizon \( T \) as

\[
\min \left\{ \frac{1}{T} \left( mK + \frac{h}{2} \sum_{l=1}^{n} \sum_{j=2}^{l} \sum_{k=j}^{n} w_{ijk} \left( \sum_{l=i}^{j-1} (Q_l + Q_{l+1})T_l + (Q_j + Q_{ijk}) \left( t_{ik} - \sum_{l=i}^{j-1} T_l \right) + (Q_{ijk} + Q_{j+1}) \left( j \sum_{l=i}^{j} T_l - t_{ik} \right) \right) \right) \right\}. \tag{2.18}
\]

For every possible production times \( m \), we minimize the average total cost in the planning horizon. So the objective is

\[
\min F = \min \left\{ \frac{1}{T} \left( mK + \frac{h}{2} \sum_{l=1}^{n} \sum_{j=2}^{l} \sum_{k=j}^{n} w_{ijk} \left( \sum_{l=i}^{j-1} (Q_l + Q_{l+1})T_l + (Q_j + Q_{ijk}) \left( t_{ik} - \sum_{l=i}^{j-1} T_l \right) + (Q_{ijk} + Q_{j+1}) \left( j \sum_{l=i}^{j} T_l - t_{ik} \right) \right) \right) \right\}. \tag{2.19}
\]
3. Solution Procedure

We solve the solution by employing Genetic Algorithm (GA). GA is a stochastic search method that works on a population of the solutions simultaneously and searches large and complicated fields based on the mechanics of natural genetics and evolutionary principles. In addition, it is particularly suitable for optimization problems with an objective function subject to numerous constraints. GA has demonstrated considerable success in these optimization problems and received more and more attentions during the past decades.

3.1. Chromosome Encoding and Decoding

We encode each chromosome as an $m$ string of integers whose components are the production cycles. Suppose that there are $n$ stages in the planning horizon. We generate $m$ integers $a_i$, $i = 1, 2, \ldots, m$ randomly to satisfy $a_1 + a_2 + \cdots + a_m = n$. For example, let $n = 10$ and $m = 4$. A chromosome is may be (2 5 1 2). It means there is total 4 times production. The first production cycle is from Stage 1 to Stage 2. The second is from Stage 3 to Stage 7. The third is in Stage 8 alone. And the last production cycle is from Stage 9 to Stage 10.

The decoding procedure is as follows. When a chromosome is determined, we could decode this chromosome to get $v_{ijk}$ first. Then according to formula (2.5) we could obtain $t_{ik}$. By formula (2.10) and formula (2.13) we would find $w_{ijk}$, $Q_i$ and $Q_{ijk}$ finally.

3.2. Crossover

Crossover generates offspring by operating parent chromosomes. Let parameter $P_c$ be the crossover probability. A chromosome will be selected as a parent when $r < P_c$, where $r$ is a random number generated from the interval $[0, 1]$. We apply single parent crossover operator. For a selected parent chromosome with $m$ integers, $n_1$ and $n_2$ are stochastically generated from the interval $[1, m]$ and different. We exchange the value of location $n_1$ and the value of location $n_2$ in the parent chromosome. Figure 2 shows an example. The stochastically generated number is 2 and 3. An offspring (2 1 5 2) is obtained after exchanging 5 and 1. The crossover operator aims to change production scheme under a fixed production times.

3.3. Mutation

Mutation modifies a chromosome to form an offspring. Let parameter $P_m$ be the mutation probability. A chromosome will be selected as a parent when $r < P_m$, where $r$ is a random number generated from the interval $[0, 1]$. We randomly choose a location $p$ in a selected
chromosome. Then generate an integer stochastically, $q$, from the interval $[1, n]$. Then we compare the value of location $p$ in the chromosome and $q$. For the offspring $(2, 1, 5, 2)$, $p$ is 3 and $q$ is 6. Obviously 5 is less than 6. So we sum the values of right position $p$ and substitute the new sum, 7, for origin value, 5. The case of mutation is depicted in Figure 3(a). If $q$ is less than 5, let the sum of two numbers which are both more than 0 equals to 5. For example, $q = 2$, which is less than 5, we find two numbers $a_1$ and $a_2$ randomly satisfying $a_1 + a_2 = 5$. Then $a_1$ and $a_2$ replace 5. The case of mutation is drawn in Figure 3(b) where $a_1 = 4$ and $a_2 = 1$. The mutation operator aims to change production times by increasing or decreasing it.

### 3.4. Evaluation Function and Selection

Evaluation function is to evaluate the quality of a chromosome. We define evaluation value of a chromosome as the objective function of the corresponding solution.

The selection strategy means how to choose the chromosomes in the current population will create offspring for the next generation. We take the roulette wheel as selection mechanism, in which each chromosome is assigned a slice of a circular roulette wheel and the size of the slice is proportional to the chromosome’s evaluation value.

### 4. Computational Results

In this section we perform a series of computational experiments to evaluate the GA proposed in Section 3 in different size problems. The experiments have been coded in C programming and implemented on an Athlon 3.10 GHz PC, with 3.12 GB of RAM, running on Windows XP.

In order to test the efficiency and universality of the GA, we generate four groups of data randomly in which stages are 10, 20, 50, and 100, named problems 1, 2, 3, and 4, respectively. Tables 1, 2, 3 and 4 show the exact values of demand extent and demand rate of every problem. The production rate is set as 320 in these problems. The setup cost

![Figure 3: Example of mutation operating.](image-url)

**Table 1: Demand data in Problem 1.**

<table>
<thead>
<tr>
<th>Stage number</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Demand extent</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td>3</td>
<td>13</td>
<td>17</td>
<td>12</td>
<td>15</td>
<td>7</td>
<td>12</td>
</tr>
<tr>
<td>Demand rate</td>
<td>102</td>
<td>73</td>
<td>264</td>
<td>152</td>
<td>119</td>
<td>61</td>
<td>29</td>
<td>292</td>
<td>215</td>
<td>211</td>
</tr>
</tbody>
</table>
of production once and holding cost per unit volume per unit time are $2 and $10000, respectively. The parameters in our GA are chosen as follows. The population size $PS = 30$, crossover probability $P_c = 0.6$, and mutation probability $P_m = 0.5$. We execute the GA in 30 independent runs on each problem. The evaluation value is recorded for every trial in every problem and shown in Figure 4. Table 5 displays the computational results including number of generations, the best found evaluation value, average evaluation value, standard deviation, and average running time for the 30 trials in the four problems. Because the scale of every problem is different, the smaller the problem is the fewer the number of generations is. The results reveal that the proposed GA could effectively solve the problem in different sizes.

Because the parameters in the proposed GA, crossover probability and mutation probability, play a crucial role of the result, we also test the best combination of the two parameters for all problems. Both crossover probability and mutation probability vary from 0.1 to 0.9 when the step is 0.1. So there are 81 combinations in every problem. The average evaluation value is obtained from 30 independent trials under all combinations. Figure 5 displays the comparison results. The best combination is shown as Table 6. It is obvious that the average evaluation value in Table 6 is much better than that in Table 5 for the same problem.

For Problem 1 the best combination parameters are $(0.4, 0.5)$ and $(0.9, 0.8)$. This indicates that Problem 1 is not sensitive to the two parameters. Owing to the small size of Problem 1, if the number of generations increases we could find better solution than the
Table 4: Demand data in Problem 4.

<table>
<thead>
<tr>
<th>Stage number</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Demand extent</td>
<td>4</td>
<td>8</td>
<td>12</td>
<td>3</td>
<td>12</td>
<td>12</td>
<td>6</td>
<td>13</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Demand rate</td>
<td>70</td>
<td>202</td>
<td>53</td>
<td>22</td>
<td>78</td>
<td>231</td>
<td>203</td>
<td>22</td>
<td>219</td>
<td>239</td>
</tr>
<tr>
<td>Stage number</td>
<td>11</td>
<td>12</td>
<td>14</td>
<td>15</td>
<td>16</td>
<td>17</td>
<td>18</td>
<td>19</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>Demand extent</td>
<td>11</td>
<td>10</td>
<td>10</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>16</td>
<td>17</td>
<td>18</td>
<td>19</td>
</tr>
<tr>
<td>Demand rate</td>
<td>149</td>
<td>23</td>
<td>239</td>
<td>71</td>
<td>231</td>
<td>168</td>
<td>111</td>
<td>234</td>
<td>79</td>
<td>87</td>
</tr>
<tr>
<td>Stage number</td>
<td>21</td>
<td>22</td>
<td>23</td>
<td>24</td>
<td>25</td>
<td>26</td>
<td>27</td>
<td>28</td>
<td>29</td>
<td>30</td>
</tr>
<tr>
<td>Demand extent</td>
<td>14</td>
<td>12</td>
<td>11</td>
<td>5</td>
<td>7</td>
<td>12</td>
<td>13</td>
<td>3</td>
<td>14</td>
<td>6</td>
</tr>
<tr>
<td>Demand rate</td>
<td>203</td>
<td>210</td>
<td>106</td>
<td>145</td>
<td>32</td>
<td>24</td>
<td>136</td>
<td>115</td>
<td>245</td>
<td>93</td>
</tr>
<tr>
<td>Stage number</td>
<td>31</td>
<td>32</td>
<td>33</td>
<td>34</td>
<td>35</td>
<td>36</td>
<td>37</td>
<td>38</td>
<td>39</td>
<td>40</td>
</tr>
<tr>
<td>Demand extent</td>
<td>11</td>
<td>12</td>
<td>6</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>9</td>
<td>7</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>Demand rate</td>
<td>97</td>
<td>209</td>
<td>289</td>
<td>238</td>
<td>174</td>
<td>114</td>
<td>174</td>
<td>194</td>
<td>124</td>
<td>94</td>
</tr>
<tr>
<td>Stage number</td>
<td>41</td>
<td>42</td>
<td>43</td>
<td>44</td>
<td>45</td>
<td>46</td>
<td>47</td>
<td>48</td>
<td>49</td>
<td>50</td>
</tr>
<tr>
<td>Demand extent</td>
<td>6</td>
<td>3</td>
<td>8</td>
<td>14</td>
<td>9</td>
<td>13</td>
<td>7</td>
<td>13</td>
<td>7</td>
<td></td>
</tr>
<tr>
<td>Demand rate</td>
<td>238</td>
<td>149</td>
<td>80</td>
<td>95</td>
<td>93</td>
<td>161</td>
<td>229</td>
<td>58</td>
<td>198</td>
<td>188</td>
</tr>
<tr>
<td>Stage number</td>
<td>51</td>
<td>52</td>
<td>53</td>
<td>54</td>
<td>55</td>
<td>56</td>
<td>57</td>
<td>58</td>
<td>59</td>
<td>60</td>
</tr>
<tr>
<td>Demand extent</td>
<td>4</td>
<td>8</td>
<td>6</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>9</td>
<td>7</td>
<td>4</td>
<td>8</td>
</tr>
<tr>
<td>Demand rate</td>
<td>242</td>
<td>250</td>
<td>293</td>
<td>159</td>
<td>180</td>
<td>198</td>
<td>185</td>
<td>83</td>
<td>285</td>
<td>262</td>
</tr>
<tr>
<td>Stage number</td>
<td>61</td>
<td>62</td>
<td>63</td>
<td>64</td>
<td>65</td>
<td>66</td>
<td>67</td>
<td>68</td>
<td>69</td>
<td>70</td>
</tr>
<tr>
<td>Demand extent</td>
<td>8</td>
<td>7</td>
<td>5</td>
<td>7</td>
<td>10</td>
<td>14</td>
<td>8</td>
<td>10</td>
<td>13</td>
<td>4</td>
</tr>
<tr>
<td>Demand rate</td>
<td>273</td>
<td>183</td>
<td>228</td>
<td>50</td>
<td>93</td>
<td>211</td>
<td>121</td>
<td>70</td>
<td>78</td>
<td>236</td>
</tr>
<tr>
<td>Stage number</td>
<td>71</td>
<td>72</td>
<td>73</td>
<td>74</td>
<td>75</td>
<td>76</td>
<td>77</td>
<td>78</td>
<td>79</td>
<td>80</td>
</tr>
<tr>
<td>Demand extent</td>
<td>6</td>
<td>7</td>
<td>9</td>
<td>10</td>
<td>3</td>
<td>4</td>
<td>12</td>
<td>5</td>
<td>5</td>
<td>13</td>
</tr>
<tr>
<td>Demand rate</td>
<td>133</td>
<td>218</td>
<td>167</td>
<td>30</td>
<td>127</td>
<td>48</td>
<td>195</td>
<td>222</td>
<td>213</td>
<td>238</td>
</tr>
<tr>
<td>Stage number</td>
<td>81</td>
<td>82</td>
<td>83</td>
<td>84</td>
<td>85</td>
<td>86</td>
<td>87</td>
<td>88</td>
<td>89</td>
<td>90</td>
</tr>
<tr>
<td>Demand extent</td>
<td>4</td>
<td>10</td>
<td>6</td>
<td>5</td>
<td>4</td>
<td>10</td>
<td>12</td>
<td>11</td>
<td>9</td>
<td>14</td>
</tr>
<tr>
<td>Demand rate</td>
<td>22</td>
<td>298</td>
<td>170</td>
<td>28</td>
<td>283</td>
<td>127</td>
<td>72</td>
<td>210</td>
<td>274</td>
<td>59</td>
</tr>
<tr>
<td>Stage number</td>
<td>91</td>
<td>92</td>
<td>93</td>
<td>94</td>
<td>95</td>
<td>96</td>
<td>97</td>
<td>98</td>
<td>99</td>
<td>100</td>
</tr>
<tr>
<td>Demand extent</td>
<td>10</td>
<td>8</td>
<td>10</td>
<td>13</td>
<td>3</td>
<td>8</td>
<td>3</td>
<td>7</td>
<td>8</td>
<td>6</td>
</tr>
<tr>
<td>Demand rate</td>
<td>75</td>
<td>59</td>
<td>222</td>
<td>202</td>
<td>239</td>
<td>105</td>
<td>113</td>
<td>16</td>
<td>100</td>
<td>299</td>
</tr>
</tbody>
</table>

Table 5: Computational results statistics.

<table>
<thead>
<tr>
<th>Problem number</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>The number of generations</td>
<td>100</td>
<td>500</td>
<td>1000</td>
<td>2000</td>
</tr>
<tr>
<td>The best found evaluation value</td>
<td>1409.4977</td>
<td>1366.0474</td>
<td>1665.7842</td>
<td>1715.5158</td>
</tr>
<tr>
<td>The average evaluation value</td>
<td>1411.9887</td>
<td>1417.7661</td>
<td>1695.2008</td>
<td>1746.4505</td>
</tr>
<tr>
<td>Standard deviation</td>
<td>1.6308</td>
<td>32.4708</td>
<td>21.1338</td>
<td>13.45</td>
</tr>
<tr>
<td>The average running time (sec.)</td>
<td>0.45</td>
<td>1.8</td>
<td>4.22</td>
<td>8.48</td>
</tr>
</tbody>
</table>

Table 6: The best combination of \((P_c, P_m)\).

<table>
<thead>
<tr>
<th>Problem number</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>The best combination</td>
<td>(0.4, 0.5), (0.9, 0.8)</td>
<td>(0.4, 0.2)</td>
<td>(0.6, 0.1)</td>
<td>(0.6, 0.1)</td>
</tr>
<tr>
<td>The average evaluation value</td>
<td>1410.8025</td>
<td>1398.782</td>
<td>1596.6894</td>
<td>1676.5948</td>
</tr>
</tbody>
</table>
former under any combination parameters. For example, when the number of generations reaches 1000, the average evaluation value is 1419.8273 from the 30 trials under the worst combination (0.1, 0.2) whose average evaluation value is 1482.9856 and number of generations is 100 in Figure 5(a). At the same time the average running time is only one second.

We find a law in Figures 5(c) and 5(d) that when given a value of crossover probability, the average evaluation value almost increases with the increase of mutation probability. And we test other experiments generated randomly where stage is from 60 to 90 (in which step is 10) under all combination parameters. The results reflect the same law. However, it is not the same case in Figures 5(a) and 5(b). This demonstrates that when the problem size is large the mutation probability should be small. So we could only find the best crossover probability. The best combination in Problems 3 and 4 is the same in Table 6. But it is just by chance because in the latter experiments we do not find the same best combination.

Table 7 reports the partial best solutions and evaluation values found during search under the best combination parameters for the four problems. The stages in a bracket in Table 7 indicate that they are in one continuous production cycle. And the stage which does not appear in Table 7 for every problem means a production cycle. For example, the first production cycle is from Stage 1 to Stage 2 in Problem 2. The second and the third production cycle are in Stage 3 and Stage 4, respectively. And the last production cycle starts from Stage 17 to Stage 20. So there are total twelve production cycles in Problem 2.

The most cases are that one production cycle includes two stages in Table 7. When one production cycle includes two stages, the sum cost is obviously smaller if the demand rate in the first stage is more than that in the second stage. So we can conclude from Table 7 that all demands rate of two successional stages in one production cycle satisfy the above condition.

The best combination in Problems 3 and 4 is the same in Table 6. But it is just by chance that when given a value of crossover probability, the average evaluation value almost increases with the increase of mutation probability. And we test other experiments generated randomly where stage is from 60 to 90 (in which step is 10) under all combination parameters. The results reflect the same law. However, it is not the same case in Figures 5(a) and 5(b). This demonstrates that when the problem size is large the mutation probability should be small. So we could only find the best crossover probability. The best combination in Problems 3 and 4 is the same in Table 6. But it is just by chance because in the latter experiments we do not find the same best combination.

Table 7 reports the partial best solutions and evaluation values found during search under the best combination parameters for the four problems. The stages in a bracket in Table 7 indicate that they are in one continuous production cycle. And the stage which does not appear in Table 7 for every problem means a production cycle. For example, the first production cycle is from Stage 1 to Stage 2 in Problem 2. The second and the third production cycle are in Stage 3 and Stage 4, respectively. And the last production cycle starts from Stage 17 to Stage 20. So there are total twelve production cycles in Problem 2.

The most cases are that one production cycle includes two stages in Table 7. When one production cycle includes two stages, the sum cost is obviously smaller if the demand rate in the first stage is more than that in the second stage. So we can conclude from Table 7 that all demands rate of two successional stages in one production cycle satisfy the above condition.

Table 7: The partial best solutions and evaluation values.

<table>
<thead>
<tr>
<th>Problem number</th>
<th>The stages in a continuous production cycle</th>
<th>Evaluation value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(3, 4), (6, 7), (8, 9)</td>
<td>1409.4977</td>
</tr>
<tr>
<td>2</td>
<td>(1, 2), (5, 6), (7, 8), (14, 15, 16), (17, 18, 19, 20)</td>
<td>1332.6984</td>
</tr>
<tr>
<td>3</td>
<td>(1, 2, 3), (4, 5), (7, 8), (16, 17), (19, 20), (21, 22, 23, 24), (26, 27, 28), (30, 31), (32, 33), (36, 37), (44, 45), (46, 47), (49, 50)</td>
<td>1487.9055</td>
</tr>
<tr>
<td>4</td>
<td>(2, 3), (6, 7, 8), (11, 12), (13, 14), (18, 19), (24, 25), (29, 30), (33, 34), (41, 42), (53, 54), (57, 58), (63, 64), (70, 71), (75, 76), (80, 81), (82, 83, 84), (95, 96, 97, 98)</td>
<td>1584.0483</td>
</tr>
</tbody>
</table>

Figure 4: Computational results.
There is similar regular pattern for demands rate of three and four successional stages in one production cycle except a case in Problem 2 from Stage 14 to Stage 16. For demand rate in Stage 1 is almost equivalent to that in Stage 2 in Problem 3, we consider that it does not violate the rule. The demand rate in Stage 14 in Problem 2 is much less than those in the next two stages, but the three stages are still included in one production cycle. The main reason is that the demand extent in Stage 14 is very short (only 3). Among the four problems, the maximum number of stages included in a continuous cycle are four, whereas the last three

Figure 5: Comparison results with combination of $P_c$ and $P_m$. 
problems it occurs only once. And demand rate in the first stage is greater than the sum of those in the latter three stages in Problem 2 and 4.

By increasing the number of stages and demands rate, the total cost must increase. But the average cost may not always be so. Although the stage number in Problem 2 is twice of it in Problem 1 and the demand rate in Problem 2 is also more than that in Problem 1, the average cost of Problem 2 is less than that of Problem 1. This suggests that the best production mode (production times and volume of production) could reduce the average cost. So the total cost must cut down.

5. Conclusion

In this paper it is found that the efficient and effective production-inventory strategy could reduce unnecessary funding, inventory cost, and production cycle. Furthermore, the production-inventory problem with multistage and varying demands, in which every stage extent is different, is investigated. Moreover, we generate a hybrid integer model to optimize the production-inventory process and a GA to find a better solution. Computational results illustrate that the proposed GA yields a high-quality solution relatively fast. According to the tests of combination parameters we find the law between the average evaluation value and mutation probability under the large size.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (Grant nos. 60870008 and 61164003) and the Lanzhou Jiaotong University Young Scientific Research Fund Project (2011020).

References

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