

# Heuristic algorithm for joint replenishment problem with a multisupplier

Myungryun Yoo (Department of Computer Science, Tokyo City University, myoo@tcu.ac.jp)

## Abstract

Generally, the joint replenishment problem (JRP) means to decide a replenishment policy that minimizes the total cost of replenishing multiple items from a single supplier. In this paper, we deal with more complex JRP where items are procured from a multisupplier. For this problem, we used a heuristic algorithm, especially Genetic Algorithm (GA). The purpose of the proposed algorithm in this paper is to minimize the total relevant costs per unit time. The effectiveness of the proposed algorithm is shown through a simulation study.

## Keywords

joint replenishment problem, multisupplier, heuristic algorithm, genetic algorithm, minimize relevant costs

## 1. Introduction

The joint replenishment problem (JRP) means to decide a replenishment policy that minimizes the total cost of replenishing multiple items from a single supplier. In such a system, the ordering costs are decided by a major common ordering cost  $S$  incurred whenever an order is placed and a minor ordering cost  $s_i$  incurred if item  $i$  is ordered. In addition, there is also an individual inventory holding cost involved as well. The structure of an optimal strategy can be very complex, even for systems with only a few products. For all items, the demand is known and constant [Brown, 1967; Goyal, 1973; Goyal, 1974; Goyal, 1982; Goyal and Satir, 1989].

Arkin proved that the JRP is a NP-hard problem [Arkin et al., 1989]. Goyal proposed an enumeration approach to obtain an optimal solution [Goyal, 1974]. Kaspi and Rosenblatt proposed an approach based on trying several values of the basic cycle time between a minimum and maximum value [Kaspi and Rosenblatt, 1991]. Olsen proposed the direct grouping for JRP [Olsen 2005]. Especially, Roundy [Roundy, 1985] introduced two simple policies called q-optimal integer-ratio and optimal power-of-two. Jackson, Maxwell and Muckstadt [Jackson et al., 1985] presented an efficient procedure for the joint replenishment problem under the restriction that the reorder intervals must be power of two times a based period length. However, these papers concerned a JRP with a single supplier.

This JRP can be extended to the JRP with a multisupplier. Benton presented an efficient heuristic algorithm for evaluating alternative discount schedules under conditions of a multi-item and a multisupplier [Benton, 1991]. Moon and Cha proposed an approach for JRP with a multisupplier using a genetic algorithm (GA) [Moon and Cha, 2005]. However, that GA is very simple. The accuracy of searching for a solution can be improved.

In this paper, we propose a new approach for the JRP where items are procured from a multisupplier. In solution algorithms, a heuristic algorithm, especially a GA is used. The purpose of the proposed algorithm is to minimize the total relevant costs

per unit time.

The rest of this paper is organized as follows: In Section 2, the problem is mathematically formulated. Section 3 introduces the GA methods and describes implementations used for this problem. Then, the experimental results are shown in Section 4. Finally, Section 5 provides discussion and suggestions for further work on this problem.

## 2. Mathematical model for JRP

Several assumptions are defined.

- The demand rate for each item is constant and deterministic.
- The replenishment lead time is known duration.
- Shortages are not allowed.
- The entire order quantity is delivered at the same time.
- There are no quantity discounts.
- The inventory holding cost for each item of each supplier is known and constant, independent of the price of each item.

The JRP is formulated as follows:

$$\min F(x, T, k) = \frac{1}{T} \left( S + \sum_{i=1}^m \sum_{j=1}^n \frac{s_{ij}}{k_i} x_{ij} \right) + \sum_{i=1}^m \sum_{j=1}^n d_i p_{ij} x_{ij} + \frac{T}{2} \sum_{i=1}^m d_i k_i h_i \quad (1)$$

$$\text{s. t. } \sum_{j=1}^n x_{ij} = 1, \forall i \quad (2)$$

$$x_{ij} = 0 \text{ or } 1, \forall i, j \quad (3)$$

$$k_i : \text{integer}, \forall i \quad (4)$$

$$T \geq 1 \quad (5)$$

Where,  $i$  ( $i = 1, 2, \dots, m$ ) is index of item and  $j$  ( $j = 1, 2, \dots, n$ ) is index of supplier.  $m$  is total number of items and  $n$  is total number of suppliers.  $d_i$  means demand rate of item  $i$ ,  $S$  means major ordering cost,  $s_{ij}$  means minor ordering cost of item  $i$  purchased from supplier  $j$ ,  $h_i$  is inventory cost of item  $i$  per unit time,  $p_{ij}$  is price of item  $i$  ordered from supplier  $j$ . Besides these

parameters, we have three decision variables;  $x_{ij}$ ,  $T$  and  $k_i$ .  $x_{ij}$  is binary variable equal to 1 if item  $i$  is ordered from the  $j$ th supplier, otherwise 0.  $T$  is basic cycle time and  $k_i$  is integer number that decides the replenishment schedule of item  $i$ .

Equation (1) is our objective function and means to minimize the total relevant costs per unit time. The total relevant costs are composed of the ordering cost, inventory cost and purchase cost. From Equation (2) to (5) are the constraints of this problem.

### 3. GA approach

We use a heuristic algorithm, especially a GA, for JRP.

#### 3.1 Encoding

A chromosome involves the information of the selected supplier and the replenishment schedule of each item at the same time. The length of a chromosome is the same to the total number of items  $m$ . The  $i$ th gene has two meaning, the index of the supplier  $j$  where item  $i$  is purchased from and  $k_i$  which decides the replenishment schedule of item  $i$ . Each value of gene is generated randomly from 0 to 1.

#### 3.2 Decoding

In the decoding procedure, the decision variable  $x_{ij}$  is decided as follows:

$$x_{ij} = 1, x_{ij'} = 0, j \neq j', \forall i \quad (6)$$

Where,  $j$  can be computed from chromosome as follows:

$$j = 1 + \lfloor n \times v(i) \rfloor \quad (7)$$

Where,  $v(i)$  is the value of a  $i$ th gene.

The decision Variables  $T$  and  $k_i$  are decided as follows:

$$k_i = k_i^{\text{LB}} + \lfloor (k_i^{\text{UB}} - k_i^{\text{LB}} + 1) \times v(i) \rfloor \quad (8)$$

$$T^*(k) = \sqrt{\frac{2(S + \sum_{i=1}^m \sum_{j=1}^n \frac{s_{ij} x_{ij}}{k_i})}{\sum_{i=1}^m d_i k_i h_i}} \quad (9)$$

Where,  $k_i^{\text{UB}}$  and  $k_i^{\text{LB}}$  are the upper bound of  $k_i$  and the lower bound of  $k_i$ , respectively.  $k_i^{\text{UB}}$  and  $k_i^{\text{LB}}$  are calculated as follows:

$$k_i^{\text{LB}} = 1 \quad (10)$$

$$k_i^{\text{UB}} (k_i^{\text{UB}} - 1) \leq \frac{2 \max\{s_{ij}\}}{d_i h_i T_{\min}^2} \leq k_i^{\text{UB}} (k_i^{\text{UB}} + 1) \quad (11)$$

$$T_{\min} = \min \left( \sqrt{\frac{2s_{ij}}{d_i h_i}} \right) \quad (12)$$

### 3.3 Selection and GA operators

For selection, we used a roulette wheel selection [Gen and Cheng, 1997; Gen and Cheng, 2000] and for GA operators, we used the one cut point crossover and the swap mutation.

### 4. Validation

Several numerical tests were performed. We compared our proposed approaches with Moon's GA [Moon and Cha, 2005] and Khouja's algorithm [Khouja et al., 2000].

For the numerical test, price  $s_{ij}$ , minor ordering cost  $p_{ij}$ , inventory cost  $h_i$  and demand rate  $d_i$  are generated randomly based on exponential distribution and normal distribution. The total number of suppliers and major ordering cost  $S$  are set to 3 and 10, respectively.

Numerical tests were performed with two items set: 30 and 50 items.

Table 1 and 2 represent the results of the tests based on exponential distribution.

Table 1: The comparison (30 items, exponential)

Terms	$F(x, T, k)$	CT (msec)
Khouja's algorithm	279,003	333
Moon's GA	279,292	376
Proposed GA	278,717	341

Table 2: The comparison (50 items, exponential)

Terms	$F(x, T, k)$	CT (msec)
Khouja's algorithm	422,927	598
Moon's GA	423,399	649
Proposed GA	422,437	609

Table 3 and 4 represent the results of the tests based on normal distribution.

Table 3: The comparison (30 items, normal)

Terms	$F(x, T, k)$	CT (msec)
Khouja's algorithm	312,078	386
Moon's GA	314,003	392
Proposed GA	310,295	389

Table 4: The comparison (50 items, normal)

Terms	$F(x, T, k)$	CT (msec)
Khouja's algorithm	478,238	607
Moon's GA	492,003	641
Proposed GA	470,923	621

In these tables, CT means computation time. In these tables, the time of the proposed GA to find a solution is shorter than that of Moon's GA. Total relevant costs per unit time of the proposed GA are better than that of the other two algorithms.

## 5. Conclusion

The joint replenishment problem (JRP) involves determining a replenishment policy that minimizes the total relevant cost of replenishing multiple items from a single supplier. In this paper, we proposed a new approach for the JRP where items are procured from a multisupplier. In solution algorithms, the Genetic Algorithm (GA) was used. The performance of the algorithm is improved compared to that of Moon's algorithm by reducing the length of the chromosome. The purpose of the proposed algorithm in this paper is to minimize the total relevant costs per unit time. From the numerical results, the result of the proposed approach is better than Moon's GA and Khouja's algorithm. We plan to design a hybrid genetic algorithm for JRP with a multisupplier considering the resource limitations.

## References

- Arkin, E., Joneja, D., and Roundy, R. (1989). Computational complexity of uncapacitated multi-echelon production planning problems. *Operations Research Letters*, Vol. 8, No. 2, 61-66.
- Benton, W. (1991). Quantity discounts under conditions of multiple items, multiple suppliers and resource limitations, *Journal of Production Research*, Vol. 29, No. 10, 1953-1961.
- Brown, R. G. (1967). *Decision rules for inventory management*. Holt, Reinhart and Winston, New York.
- Gen, M. and Cheng, R. (1997). *Genetic algorithms & engineering design*. John Wiley & Sons.
- Gen, M. and Cheng, R. (2000). *Genetic algorithms & engineering optimization*. John Wiley & Sons.
- Goyal, S. K. (1973). Determination of economic packaging frequency for items jointly replenished. *Management Science*, Vol. 20, No. 1, 232-235.
- Goyal, S. K. (1974). Determination of optimum packaging frequency of items jointly replenished. *Management Science*, Vol. 21, No. 1, 436-443.
- Goyal, S. K. (1982). A note on formulation of the multi-item single supplier inventory model. *Journal of the Operational Research Society*, Vol. 33, No. 1, 287-288.
- Goyal, S. K. and Satir, A. T. (1989). Joint replenishment inventory control: Deterministic and stochastic models. *European Journal of Operational Research*, Vol. 38, No. 1, 2-13.
- Jackson, L. P., Maxwell, L. W., and Muckstadt, J. A. (1985). The joint replenishment problem with power-of-two restrictions. *AIIE Transactions*, Vol. 17, No. 1, 25-32.
- Kaspi, M. and Rosenblatt, M. (1991). On the economic ordering quantity for jointly replenished items. *International Journal of Production Research*, Vol. 29, No. 1, 107-114.
- Khouja, M., Michalewicz, Z., and Satoskar, S. (2000). A comparison between genetic algorithms and the RAND method for solving the joint replenishment problem. *Production Planning & Control*, Vol. 11, No. 6, 556-564.
- Moon, I. K. and Cha, B. C. (2005). The joint replenishment and supplier selection problem. *International Workshop on Intelligent Logistics Systems*, Korea, 229-238.
- Olsen, A. L. (2005). An evolutionary algorithm to solve the joint replenishment problem using direct grouping. *Computers & Industrial Engineering*, Vol. 48, No. 2, 223-235.
- Roundy, R. (1985). A 98% effective integer-ratio lot sizing for one-warehouse multi-retailer systems. *Management Science*, Vol. 31, No. 11, 1416-1430.

(Received March 26, 2018; accepted April 11, 2018)