혼합정수계획법 및 유전자 알고리즘을 이용한 다품목 재고 시스템의 주문 주기 상쇄에 관한 연구

문일경, 차병철, 김선권
부산대학교 산업공학과
ikmoon@pusan.ac.kr

Offsetting Inventory Cycle of Items Sharing Storage using Mixed Integer Programming & Genetic Algorithm

IL Kyeong Moon, Byung Chul Cha, Sun Kwon Kim
Department of Industrial Engineering, Pusan National University

Abstract

The ability to determine the optimal frequencies and offsets for independent and unrestricted ordering cycles for multiple items can be very valuable for managing storage capacity constrained facilities in a supply chain. The complexity of this problem has resulted in researchers focusing on more tractable surrogate problems that are special cases of the base problem. Murthy *et al.*(European Journal of Operation Research 2003) developed insights leading to solution of the original problem and present a heuristic for offsetting independent and unrestricted ordering cycles for items to minimize their joint storage requirements. However, their study cannot find optimal solution due to the Greedy Heuristic solution procedure. In this paper, we present a complete procedure to find the optimal solution for the model with a integer programming optimization approach and genetic algorithm. Numerical examples are included to compare each model with that of Murthy et al. Research of this type may prove useful in solving the more general problem of selecting order policies to minimize combined holding, ordering, and storage costs.

1. 서론

다품목을 취급하는 발주업체에서는 다수의 품목을 개별적으로 주문하기보다는 공동발주(joint replenishment)를 선호하는데 이는 주문 및 운송에 소요되는 비용을 줄일 수 있을 뿐만 아니라 구매, 수송, 검사에 따른 여러 운영상의 장점도 존재하기 때문이다. 그러나 이와 같이 공동발주를 하게 되면 불필요한 품목을 미리 보유하게 됨으로써 재고비용이 상승하게 되고 요구되는 창고의 최대 저장공간도 증가하게 되는 단점이 있다. 본 연구는 이러한 다품목을 취급하는 재고시스템을 대상으로 기존의 연구에서 고려되지 못했던 창고의 최대 저장공간을 최소화 하기위한 최적의 주문정책을 구하고자 한다. 이는 개별발주에 비해 여러 가지 제반비용 및 운영상의 장점에도 불구하고 공동발주를 함으로써 발생되는 창고의 저장공간에 대한 문제를 해결하기 위한 것이다.

다품목에 대한 주문정책으로 첫 번째 주목할 만한 접근법은 Parsons [7]의 연구로서 독립적 개별 품목을 위한 경제적 주문 간격을 구하였다. 또한 Pirkul and Aras [8], Benton [1] 등의 연구에서는 창고의 저장공간에 대한 제약하에서 다품목, 다수공급자, 가격할인이 고려된 주문정책에 관한 연구를 하였다.

두 번째 접근법으로 Krone [5], Hall [3], 및 그 외 다수의 연구에서 몇몇 품목들을 그룹화 하거나 전체 품목을 공통주기를 가지고 주문하는 방법을 연구하였다.

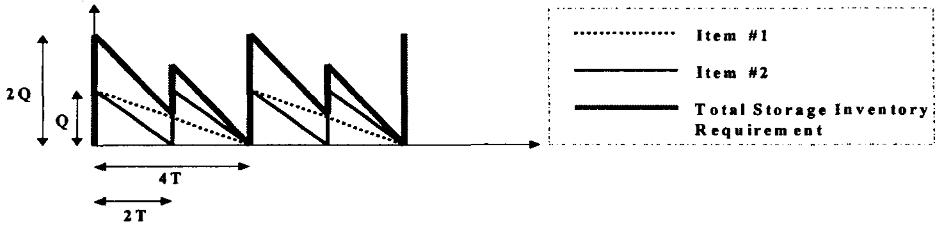
세 번째 접근법은 Goyal and Belton [2], Kaspi and Rosenblatt [4] 등의 연구에서 기본주문주기를 바탕으로 하여 각 품목들을 정수배 또는 2의 승수배(a Power of Two Multiple)로 공동발주 하기 위한 주문정책을 제안하였다. 그러나 이러한 접근 방법은 공동발주에 대한 비용상의 잇점만을 고려하였을 뿐 공동발주를 함으로써 발생되는 창고의 저장공간에 대한 문제를 간과하였다.

최근 Murthy et al. [6]의 연구에서는 이러한 공동발주로 파생되는 창고의 저장공간에 대한 문제를 해결하기 위한 통찰력 있는 방법론을 제시 하였으며, 제시된 휴리스틱 모형을 통해 요구되는 창고의 최대 저장공간을 15.46%나 절감 시킬 수 있었다. 그러나 제시된 알고리즘은 최대최소방법(MiniMax: 비관적 의사결정)에 입각한 그리디 휴리스틱(Greedy Heuristic)을 사용함으로써 많은 개선 가능성을 안고 있다. 이에 본 연구에서는 문제의 특성을 반영한 혼합정수계획모형을 통해 최적해를 구하고, 본 대상 문제가 NP-hard 문제임을 감안하여 유전자 알고리즘을 이용한 시간 효율적인 알고리즘을 개발하였다

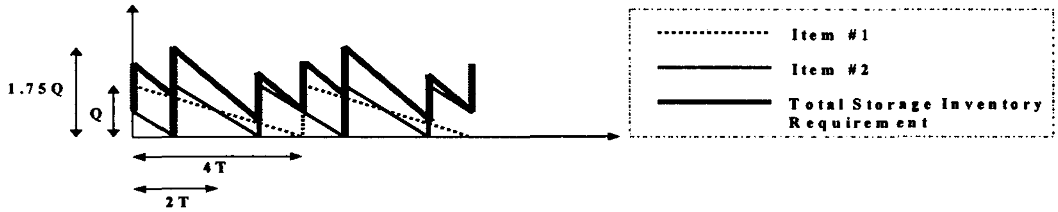
2. 연구내용

2.1 옵셋팅(Offsetting)

공동발주(joint replenishment)에서 발생되는 창고의 최대 저장공간에 대한 문제를 해결하기 위하여 간단한 예제를 통해 재고주기를 옵셋팅(Offsetting)하는 방법을 소개한다. 여기서 옵셋팅(Offsetting)이란 다품목을 묶어서 기본주문주기의 정수배로 주문하되 재고 주기상의 여러 시점에서



< 그림 1. 옵셋팅을 실시하지 않을 경우의 예제 >



< 그림 2. 옵셋팅을 실시하였을 경우의 예제 >

발생되는 재고의 모임현상을 감소시키기 위해 개별 품목의 주문시점을 변경하는 방법이다.

< Table 1. 예제 >

품 목	기본주문주기	주문주기	주문량
Item #1	T	4 T	Q
Item #2	T	2 T	Q

1) 옵셋팅이 없을 경우 : <그림 1>과 같이 두 가지 품목의 주기는 같은 시간에 시작되고 요구되는 최대 저장공간은 2Q가 된다는 것을 알 수 있다.

2) 옵셋팅이 있을 경우 : <그림 2>와 같이 첫 번째 품목은 변경 없이 그대로 둔 상태에서 두 번째 품목(#2)을 시작시점으로부터 T기간 만큼 이후에 주문을 시작하게 된다(Offsetting)면 요구 저장 공간의 최고점은 1.75Q가 될 것이다. 따라서 <그림 1>과 비교하여 볼 때 <그림 2>에서 요구되는 최대 저장공간은 12.5%가 감소하게 된다.

2.2 혼합정수계획법 모형

본 연구는 서론에서 언급한 발주정책 중 세번째 접근법으로써, 각 품목을 기본주문주기의 정수배(TBO_i)로 주문하는 공동발주(joint replenishment)에서 구해진 최적의 기본주문주기와 각 품목의 주문 주기(TBO_i)를 기본 입력으로 창고의 최대 저장공간을 최소화하기 위한 옵셋팅 주문주기를 구하는 것이 목적이다. 정수계획법모형에 사용되는 기호들과 모형은 아래와 같다.

Z	정수인 t=0 시점의 재고수준
Z^+	0보다 큰 정수
n	품목의 수
i	품목의 번호
LCM	최소공배수
$TBOi \in Z^+$	품목i의 주문주기
I_{it}	품목i의 t시점에서의 재고수준
D_i	품목i의 1기간동안의 수요
$Q_i = D_i TBO_i$	품목i의 주문량

$$X = \begin{cases} 1, & t\text{시점에주문하면} \\ 0, & otherwisee \end{cases}$$

$$Min \quad Z$$

subject to

$$\sum_{t=0}^{TBOi-1} X_{it} = 1 \qquad i = 1, 2, \ldots, n \quad \square(1)$$

$$I_{i0} = Q_i X_{i0} + \frac{Q_i}{TBO_i} \sum_{t=1}^{TBOi-1} tX_{it}$$

$$i = 1, 2, \ldots, n \quad (2)$$

$$I_{it} = I_{i(t-1)} + Q_i X_{it} - D_i$$

$$i = 1, 2, \ldots, n,\ t = 1, \ldots, \mathrm{TBO}_i\text{-}1 \quad (3)$$

$$Z \geq \sum_{i=1}^{n} I_{ik},\ k \begin{cases} t, & t < TBO_i \\ Mod\,(t/TBO_i), & t \leq TBO_i \end{cases}$$

$$t = 1, 2 \ldots, LCM\text{-}1 \quad (4)$$

$$Z = \sum_{i=1}^{n} I_{i0} \qquad i = 1, 2, \ldots, n \quad (5)$$

이 정수계획모형의 의사결정변수는 재고를 옵셋팅 하기위한 X_{it}로서 i품목의 최초의 발주시점 t를 결정하는 것이다.

목적함수는 무한의 기간동안 요구되는 최대 저장 공간을 최소화하는 것이다. 이것은 t=0의 시점으로부터 시작해서 LCM(모든 품목들의 주문주기에 대한 최소공배수) 기간동안 창고의 저장공간이 최대가 되는 시점은 한번 발생 되며, 전체 저장 공간상의 유형은 LCM을 주기적으로 반복하게 된다. 이것은 무한 기간이라고 해도 LCM을 반복적으로 재고 주기가 유지하므로 그 시작과 끝은 모든 품목들의 주문주기에 대한 최소공배수로 구해진다. 제약식 (1)은 각 품목의 최초의 발주시점을 결정하기 위한 것이다. 제약식 (2)는 각 품목의 t=0에서의 재고수준으로 각 품목의 최초의 발주시점에 의해 결

정되어진다. 제약식 (3)은 $t=0$에서의 재고수준과 최초의 발주시점에 따라 달라지는 $t=1$부터 $t=TBO_i-1$까지의 재고수준을 나타낸다. 제약식 (4)는 $t=1$ 이후 $LCM-1$까지 각 시점에서 발생되는 요구되는 저장 공간의 재고수준에 대한 합을 나타낸다. 여기서 모든 시점에서의 재고수준은 Z보다 작거나 같아야 한다. 제약식 (5)는 각 시점에서 모든 품목들에 대한 재고들의 합이 최대가 되는 시점을 $t=0$시점이 되도록 했다.

아래의 <Table 2> 예제는 Murthy et al. [6]의 연구에서 사용되었던 예제이다. 9개의 품목으로 구성된 다품목 재고이며 각각 주문량 Q와 주문주기 TBO_i를 가지고 있다. 예를 들면, 첫 번째 품목 #1은 *4*기간의 주문주기와 *100*의 주문량을 가지고 있으며 이는 곧 기본 주문주기 $T=1$ 기간동안 *25*만큼의 재고를 소진하고, 이를 무한의 기간동안 반복하게 된다는 의미이다.

< Table 2. Murthy et al. 예제 >

품목	1	2	3	4	5	6	7	8	9
Qi	100	200	81	144	150	160	90	60	50
TBOi	4	5	9	12	15	8	6	12	2

아래의 <Table 3> 모델 비교는 각각의 해법에서 제시한 값의 비교를 보여주고 있다.

< Table 3. 모델 비교 >

품목	Without Offsetting		Murthy *et al.* Heuristic		Optimal, MIP Model	
	Xit	*Ii0*	*Xit*	*Ii0*	*Xit*	*Ii0*
1	*X10*	100	*X10*	*100*	*X10*	100
2	*X20*	200	*X24*	160	*X20*	200
3	*X30*	81	*X30*	81	*X38*	72
4	*X40*	144	*X40*	144	*X47*	84
5	*X50*	150	*X50*	150	*X512*	120
6	*X60*	160	*X66*	120	*X66*	120
7	*X70*	90	*X75*	75	*X74*	60
8	*X80*	60	*X84*	20	*X811*	5
9	*X90*	50	*X91*	25	*X91*	25
합계		1035		875		786

옵셋팅을 실시하지 않았을 경우 한번에 모든 품목을 주문하게 되므로 재고의 최대 요구 저장공간은 1035 만큼 공간이 필요하다는 것을 알 수 있다. 그러나 Murthy et al. [6]의 연구에서는 옵셋팅을 사용하여 875만큼의 최대 저장공간만을 사용하게 되므로 옵셋팅을 실시하지 않았을 경우와 비교하여 15.46% 감소했다. 그러나 본 연구의 혼합정수계획 모델을 통해 이 문제에 대한 최적해를 구해본 결과 786으로 옵셋팅을 실시하지 않았을 경우 대비 24.06%, Murthy et al. [6]의 휴리스틱 대비 10.17% 감소함을 알 수 있었다.

3. 유전자 알고리즘

많은 최적화 문제들은 탐색영역에 있어서 변수와 비선형성이 복합된 복잡한 문제들로 구성되어 있으며, 본 문제와 같은 전형적인 NP-hard의 문제는 해를 구하는데 많은 어려움이 있다. 정수계획모형의 방법으로 작은 크기의 문제에 대한 최적해를 구하였다 하더라도 보다 큰 문제에 대해서는 최적해를 찾기 위한 시간적인 제약이 발생하게 된다. 이를 극복하기 위한 다른 대안은 빠른 시간 내에 최적해에 가까운 값을 구하는 것이라 하겠다. 그래서 Murthy *et al.* [6]의 연구에서 보다 성능이 우수한 근사 최적해를 구하기 위하여 유전자 알고리즘을 이용한 휴리스틱을 개발하였다.

일반적으로 유전자 알고리즘은 대상이 되는 후보해들을 문자나 기호의 배열인 염색체(chromosome)로 표현하며 염색체를 구성하는 문자나 기호들을 유전자(gene)라 부른다. 표현된 염색체에서 새로운 염색체를 생성하고 우수한 염색체만을 선택하여 새로운 염색체를 생성하는 과정을 반복하여 점진적인 갱신을 함으로써 전역적인 최적해를 탐색하게 된다. 모집단의 갱신은 자연계의 진화 과정인 선택(selection), 교차(crossover) 그리고 돌연변이(mutation)에 의해 주어진 문제에 대해 최적점에 근접하는 개체를 생성하여 최적해를 찾아가는 절차를 수행한다. 일반적인 유전자 알고리즘의 해법 절차는 아래 <Table 3>과 같다.

<Table 3. 일반적 유전자 알고리즘의 해법 절차>

```
Begin
    t: =0
    P(t)의 초기화(초기모집단 생성)
    P(t)의 적응도 평가
    While(종료조건이 만족되지 않으면) Do
        Begin
            t: =t+1
            P(t-1)로부터 P(t)를 선별
            P(t)의 유전연산(교차와 돌연변이)
            P(t)의 적응도 평가
        End
    End
```

3.1 염색체 표현

본 유전자 알고리즘에서 사용된 염색체는 랜덤키 표현(Random Key Representation)을 이용하였다. 유전자 알고리즘을 이용하여 목적값을 최소로 하는 개별 품목에 대한 최초 주문시점의 배치가 염색체(chromosome)로서 표현되어야 하고(encoding process), 반대로 알고리즘 수행과정에서 얻어진 염색체가 최초 주문시점으로 표현되어야 한다(decoding process). 만약 염색체를 최초 주문시점으로 그대로 사용할 경우 유전 파라메타의 연산과정을 거치면서 불필요한 값들이 만들어 지게 되는데 이것은 비효율적인 유전자를 생성시키는 것과 같은 결과를 초래하기 때문에 유전자의 발생값을 0과 1 사이의 실수인 난수를 사용한 랜덤키 염색체를 이용하였다. 그 후 다양한 유전 연산자인 유전 파라메타를 통해서 최적점에 근접하는 절차를 수행하게 된다. 표현된 염색체는 아래의 <그림 3>과 같으며 유전자 개수는 전체 재고 품목의 수로 나타낼 수 있고, 하나의 유전자는 각각의 품목에 대한 최초 발주 시점이 된다. <그림 3>은 최적의 해를 도출하였을때의 string을 표시하였다. 위쪽그림의 랜덤키 에서 생성된 랜덤한 유전자는 표현식을 통해 하나의 염색체로서 역할을 하게 된다. 표현식에 의하여 Decording을 하게 되면, 최적점에 근접하는 개별 품목의 최초 주문시점을 얻을 수 있으며, 염색체의 Decording 표현식은 아래와 같다.

< Chromosome >

0.0138	0.1121	0.9841	0.6448	0.8534	0.7991	0.7176	0.1167	0.5637

< Decording >

$$Round\,(Chromosome\ \times (\,TBOi-1\,))$$

A=(g[1] g[2] g[3] g[4] g[5] g[6] g[7] g[8] g[9])

0	0	8	7	12	6	4	1	1

< 그림 3. Chromosome & Decording >

사전에 결정된 개별 품목의 재고 주기는 TBO_i에 1을 차감하게 되는데, 이것은 최초 주문시점이 0의 시점부터 시작될 수 있기 때문이다. 여기에서 Random Number는 랜덤키에서 발생된 값이 연산자를 통해 진화하고 이를 $TBOi-1$ 값에 곱하게 되면 개별 품목에서 발생 가능한 최초 주문시점중 하나가 선택된다. Round는 숫자를 지정한 자릿수로 반올림하게 되는데 요구하는 정수값을 구하기 위한 것이다. 여기서 주목할 점은 결정된 개별 품목의 최초 주문 가능 시점에서 벗어난 임의의 수치가 발생되지 않게 함으로써 불필요하게 생성되는 값들을 사전에 차단시키려는 이유이다.

3.2 적합도 평가

적합도 평가는 생성된 개체집단(population)에서 염색체들의 우수성을 평가하기 위한 수단으로 사용된다. 본 연구에서 사용된 적합도 함수(fitness fuction)은 아래와 같다.

$$eval(A) = Min\,Z$$

각 염색체를 위한 목적함수는 제시된 수학적 모형에서의 목적식을 이용하였다. 생성된 개체집단의 염색체들이 제시하는 최초발주시점에 따라 0시점부터 LCM기간까지 각 시점에서 발생되는 재고수준의 최대 저장 공간을 최소로하는 염색체를 평가하기 위한 것이다.

3.3 교배연산자(crossover)

교배연산은 재편성된 모집단으로부터 일정한 교배확률(Pc)에 따라 교배를 위한 염색체의 쌍을 만들어 각 염색체의 유전자 교환을 통해 모체보다 우수한 염색체를 생성시키기 위한 단계이다. 이를 통해 생성되는 자식염색체들이 더욱 높은 적합도를 가진 염색체를 형성할 가능성을 높일 수 있다. 임의의 두 염색체를 선택하고 지정한 유전자를 기점으로 교배점과 길이는 임의로 선택할 수 있다. 본 연구에서는 균등교차연산자를 이용하였다. 균등교차는 임계확률을 가지고 임계확률 값에 따라서 부모염색체로부터 복사를 해오는 방법이다. 균등교차는 일점교차나 다점교차에 비해 스키마의 결합 형태가 다양하고 구조를 잘 유지하는 장점이 있다.

3.4 돌연변이 연산자(mutation)

돌연변이란 교배에 참여한 두 부모 개체의 형질과는 상관없이 주어진 확률에 따라 임의 위치의 유전자 값을 변이 시키는 조작을 돌연변이라 한다. 초기 모집단 생성 후 복제 및 도태에 의한 교차만으로는 광범위한 최적개체의 탐색에 한계가 있다. 이와 같은 한계를 극복하고 한정된 탐색범위를 벗어나 새로운 탐색공간으로의 전이가 가능하도록 함으로써 국소 최적치 수렴 구역을 탈출할 수 있도록 하기 위한 인위적인 조작을 돌연변이라고 할 수 있다. 본 연구에서는 비균등 돌연변이 연산자(non-uniform crossover operator)를 이용하였다. 비균등 돌연변이 연산자는 유전자 알고리즘이 진행됨에 따라 점점 변이의 강도를 줄여가는 방법이다.

3.5 실험결과

제시된 모형의 시간 효율적인 성능 및 실현가능성을 검증하기 위하여 실험을 실시하였다. 본 연구에 사용된 유전자 알고리즘의 적용은 상업용 유전자 알고리즘 최적화 solution인 Evolver 4.0을 이용하였다. 사용한 parameter로서 교차율(crossover rate)과 돌연변이율(mutation rate)은 각각 50%와 10%로 고정하였다. 실험은 Murthy et al.의 논문에서 사용된 예제를 이용하였다. 유전자 알고리즘의 실험결과로서 상당히 빠른 시간 내에 혼합정수계획으로 구한 최적해를 얻을 수 있었으며 이를 통해서 제시한 유전자 알고리즘의 시간효율성 및 유효성이 있음을 알 수 있었다.

4. 결론

본 연구는 다품목 재고시스템의 공동발주 모형에서 요구되는 창고의 최대 저장공간을 최소화 하기위한 주문정책에 관한 연구이다. 최근에 Murthy et al. [6]의 논문에서는 창고의 최대 저장공간을 최소화하기 위하여 옵셋팅 주문주기에 대한 휴리스틱을 제시하였다. 그러나 그들의 연구는 우수한 해를 얻지 못하였다. 이에 본 연구에서는 문제의 특성을 반영하여, 혼합정수계획법을 사용하여 최적해를 얻었고, 이 문제가 전형적인 NP-hard 문제임을 감안하여 유전자 알고리즘에 기초한 시간 효율적인 휴리스틱 방법을 개발하고, 기존 논문에서 사용된 예제를 이용하여 그 결과를 서로 비교하였다. 본 논문에서 개발된 기법들은 주문비용, 재고유지비용 및 저장비용 등의 제반비용을 최소화하기 위하여 최적의 주문정책을 선택하기 위한 보다 일반적인 문제를 해결하는데 매우 유용할 것으로 사료된다. 향후 연구과제는 이런 일반적인 문제를 고려한 재고 정책에 관한 연구를 진행할 예정이다.

참고 문헌

[1] Benton, W. C. (1991), "Quantity discount decisions under conditions of multiple items, multiple suppliers and resource limitations", *International Journal of Production Research* 29, 1953-1961.

[2] Goyal, S. K. and Belton, A. S. (1979), "On a simple method of determining order quantities in joint replenishment for deterministic demand", *Management Science* 25, 604.

[3] Hall, N. G. (1988), "A multi-item EOQ model with inventory cycle balancing", *Naval Research Logistics Quarterly* 35, 319-325.

[4] Kaspi, M. and Rosenblatt, M. J. (1983), "An improvement of silvers algorithm for the joint replenishment problem", *IIE Transactions* 15, 264-267.

[5] Krone, L. H. (1964), "A note on economic lot sizes for multi-purpose equipment", *Management Science* 10, 461-464.

[6] Murthy, N. N., Benton, W. C. and Rubin, P. A. (2003), "Offsetting inventory cycles of items sharing storage", *European Journal of Operational Research* 150, Issue 2, 304-319.

[7] Parsons, J. A. (1966), "Multi-product lot size determination when certain restrictions are active", *Journal of Industrial Engineering* 17, 360-365.

[8] Pirkul, H. and Aras, O. A. (1985), "Capacitated multiple item ordering problem with quantity discounts", *IIE Transactions* 17, 206-211.