碩 士 學 位 論 文

指導敎授 金 大 弘

JIT 구매 하에서의 다품목의 발주정책에 관한 연구

An Integrated Inventory Model for Multi-Item in Just-In-Time Purchasing

2 0 0 1 年 12 月

漢 城 大 學 校 大 學 院

産 業 工 學 科

産 業 工 學 專 攻

金 容 喆

碩 士 學 位 論 文

指導教授 金 大 弘

JIT 구매 하에서의 다품목의 발주정책에 관한 연구

An Integrated Inventory Model for Multi-Item in Just-In-Time Purchasing

위 論文을 工學 碩士學位 論文으로 提出함

2 0 0 1 年 12 月

漢 城 大 學 校 大 學 院

産 業 工 學 科

産 業 工 學 專 攻

金 容 喆

金容喆의　　　工學 碩士學位 論文을 認定함

2 0 0 1 年 12 月

審査 委員長　　　　　(印)

審 査 委 員　　　　　(印)

審 査 委 員　　　　　(印)

<목 차>

<표 목 차>

<그 림 목 차>

제 1 장 서 론

국제경제환경 하에서 경쟁력을 갖추기 위하여 상당수의 제조업에서는 생산혁신방법중의 하나인 JIT 생산방식을 도입하는 기업들이 늘어나고 있으며, 자동차 제조업계를 중심으로 전자, 기계 등 일본과의 직·간접적 경쟁관계에 직면하여 있는 기업에서의 도입 움직임이 더욱 활발한 실정이다. 특히 JIT 생산방식의 핵심요소중의 하나인 수요업체와 공급업체의 협력관계가 국내 제조업체간에 많은 관심의 대상이 되고 있는데, 이는 완제품 생산을 위하여 상당수의 부품을 부품공급업체로부터 구매하게 되며 이로 인하여 수요업체의 생산성 및 제품의 품질이 상당부분 구매업체에 의하여 결정되기 때문인 것으로 판단된다. JIT 생산방식의 성공적인 정착을 위하여 외부 JIT라 할 수 있는 JIT 구매방식의 도입이 필수적이며 이는 수요업체 뿐만 아니라 공급업체들의 노력과 협조가 필요한 사항이다. JIT 구매의 성공적 도입은 자재구매제도의 변화(필요한 시점에 꼭 필요한 양의 부품을 필요한 장소에 제공)를 유발할 뿐만 아니라 자재재고수준, 생산성, 원가 및 제품품질에 상당한 영향을 미친다 [2,5].

이러한 JIT 철학은 일본 제조업이 수많은 성공을 이끌어 내고, 더욱이 생산 및 재고관리의 연구에 많은 발전을 이루게 하였다. JIT 생산체계에서 구매는 회사의 운영을 성공적으로 이끌기 위해 매우 필요하다. 구매자는 필요한 시간에 필요한 양만큼 좋은 품질의 품목을 배송 할 수 있는 신뢰할만한 공급업자를 필요로 하며 이 공급업자와 장기적인 계약관계를 구축하게 된다. 그러나 과거의 전통적인 구매환경에서는 양 집단이 서로 적대적인 관계인 경우가 많으며, 한 집단의 이익을 최대화하려는 재고정책이나 협상이 다른 집단에게는 더 많은

비용이나 손실을 가져다 주는 경우가 자주 발생한다. JIT 구매 환경 하에서는 이 두 집단을 하나의 공급사슬로 묶어 발주량을 다빈도 소량으로 납품하게 되며 공급사슬 전체의 이익을 위해 서로 효과적인 협상을 해야 한다. 이러한 시스템이 구축되면 장기적으로 공급사슬 전체의 이익이 증가하게 되며 양 집단에게 동시에 이익을 가져오게 된다[6].

따라서 본 연구의 목적은 JIT 구매환경 하에서 구매자-공급업자간의 다품목 발주정책에 관한 유형을 살펴보고 다양한 발주정책이 공급사슬 전체에 어떠한 영향을 미치는지 살펴보고자 한다. 또한 상호간에 바람직한 협력관계가 형성될 때 서로에게 이익이 되는 성과를 살펴보고자 한다.

제 2 장 연구배경

JIT 구매의 기본 이념은 JIT 생산을 위해 품목 혹은 자재를 소량으로 조달할 것을 구매자와 체결하는 것이며 핵심적인 요소는 목 크기(lot-size)의 감소, 빈번하고 신뢰성 높은 조달계획, 조달기간의 단축과 높은 신뢰도, 그리고 구매품목의 높은 품질수준유지 등이다. JIT 시스템 하에서 구매자-공급업자간의 관계가 동반자 혹은 협력적인 관계에서 수행이 되면 재고비용의 절감, 품질의 향상, 생산성 향상, 효율적인 경영관리 등의 효과를 이룰 수 있다[1].

JIT 구매 하에서의 구매자-공급업자의 조달정책에 관한 연구는 몇몇 문헌에서 이미 다루어졌다. Pan and Liao [7]는 전통적 EOQ를 수정하여 JIT 구매하에서 발주량을 목 분할하여 납품하는 경우에 총비용에 미치는 효과를 처음으로 분석하였다. 그러나 그들의 연구에서는 다빈도로 납품하는 경우의 고정 운송비를 고려하지 않았으며, Ramasesh [8]는 앞의 연구를 확장하여 총발주비용을 순수 발주비용부분과 소롯트로 분할하여 납품하는 데 드는 운송비로 나누어 고려하였으나 구매자에게서 발생하는 총비용만을 고려하였고 공급업자에게서 발생하는 비용은 고려하지 못하였다. Goyal[3]은 제품의 수요가 알려져 있고 일정한 상황에서 구매자와 공급업체의 통합 총비용을 최소화하는 발주량을 찾을 수 있는 해법을 제시하였으며, Miller and Kelle[6] 및 Ha and Kim[4]은 단일품목의 경우 JIT 구매 하에서 단일구매자-단일공급업자의 통합재고모형을 제시하였고 운송비를 통합모형에서 고려하였다.

본 연구에서는 단일품목인 경우의 JIT 구매에 관한 정량발주모형을 확장하여 목 분할 정책(lot-splitting policy) 하에서 다품목을 고려한 각 품목별 발주주기 및 각 품목별 공통운송횟수를 결정하는 통합 모형(integrated model)을 제시하고자 한다. 일반적으로 구매자와 공급업체간에 다품목을 거래하는 경우에

발주간격을 조절함으로써 일부 품목의 공통발주 및 공통운송이 가능하게 되며 전체적으로 발주비용이나 운송비를 공유하여 절감하게 된다. 따라서 본 연구에서는 구매자와 공급업자에서 발생하는 발주비용, 재고유지비용, 생산준비비용 및 운송비용을 모두 고려한 총비용을 구하고, 다른 모형에 비해 총비용을 절감할 수 있는 발견적 해법(heuristic method)을 제시하려고 한다. 또한 예제를 통하여 통합된 발주정책이 구매자와 공급업자 모두에 의해 협력적인 방법으로 적용될 때, 기존의 정책에 비해서 양 집단 모두에게 이익이 될 수 있다는 것을 보였다.

제 3 장 다품목 발주정책의 개발

구매 측면에서 다품목을 묶어서 공동발주를 시행하는 것은 고정발주비용과 운송비용을 통합시키는 것이므로 각 품목당 목 크기를 감소시킨다. 구매자와 공급업자간에 목 크기를 결정하는 방법은 세 가지 경우가 있다. 첫째, 각 품목을 각각 독립적으로 발주하는 경우이다. 이러한 방법은 전혀 품목간에 발주를 통합시키지 않는 경우이며 결과적으로 발주비용이 가장 많이 발생한다. 둘째, 모든 품목을 합쳐서 동시에 발주하는 경우이다. 모든 품목을 매 주문마다 통합을 시키기 때문에 첫 번째 방법보다는 비용이 더 낮다. 그러나 이러한 방법은 수요가 큰 품목이든 작은 품목이든, 혹은 단위기간 당 재고유지비용이 큰 품목이든 작은 품목이든 모두 통합을 시키기 때문에 실제로 발주빈도가 적을 수 있는 품목에도 개별발주비용이 부과된다. 마지막으로 통합해서 발주하되 매 발주에 모든 품목이 포함되는 것이 아니라 각 품목들을 고려해서 선택적으로 통합발주하는 경우이다. 실제로 각 품목들의 연간수요나 생산율이 각기 다르기 때문에 발주빈도가 많은 제품은 매 발주주기마다 발주하고 적은 제품은 두 번, 혹은 세 번의 발주주기마다 발주하는 경우가 더 현실적이다. 본 논문에서는 이 세 가지 모형에 대하여 살펴보고 동일한 예제를 적용하여 비교해 보았다.

3.1 기호정의 및 가정

본 연구에서는 JIT 구매 하에서 단일품목의 구매자-공급업자의 통합모형 [4,6]을 확장하여 구매자와 공급업자간에 다품목을 발주하는 경우에 품목별 발주주기 및 목 분할 납품 횟수를 결정하는 통합재고모형을 유도하려고 한다. 여기서 고려되는 비용으로는 구매자의 발주비용, 공급업자의 생산준비비용, 양측

에서 발생하는 재고유지비용 및 목 분할로 운송하는 데 소요되는 운송비용 등이다. 먼저 품목별 발주주기가 동일한 경우, 즉 모든 품목이 공통발주주기 T마다 함께 발주되는 경우를 살펴보고 더 나아가 현실적으로 각 품목별로 발주주기를 달리할 수 있으며 공통발주주기(T)의 정수 배의 시점에 주문이 가능하도록 발주주기를 정할 수 있도록 하였다. 공급업자는 주문이 들어오면 생산준비비용을 들여서 생산을 하고 JIT 구매 하에서 소롯트(small-lot) 납품을 위하여 발주량을 N번에 나누어서 목 분할 납품을 하게 되며 여러 품목간에 운송수단을 공유하여 한 번 운송하는데 일정한 공통운송비용(Z)이 발생하게 된다.

기호정의

D_i : 품목 i 의 연간수요,

P_i : 품목 i 의 연간생산율, (P_i 〉 D_i)

s_i : 품목 i 의 공급업자의 생산준비비용,

A : 통합주문의 공통발주비용,

a_i : 품목 i 의 개별품목 발주비용,

H_{Bi} : 품목 i 의 구매자 재고유지비용,

H_{Si} : 품목 i 의 공급업자 재고유지비용,

Q_i : 품목 i 의 발주량,

T : 공통 발주주기,

Z : 운송 건당 운송비용,

N : 공통 발주주기 T기간동안의 공통 운송횟수

기본 가정은 전통적인 경제적 발주량(EOQ) 모형처럼 다음의 상황에 기반

을 두고 있다.

1) 각 품목의 수요는 일정하고 확정적이다.
2) 선행기간이 일정하고 알려져 있다.
3) 각 품목의 주문당 발주비용과 단위당 유지비용은 고정되어 있다.
4) 품절을 허용하지 않는다.

3.2 단일품목의 정량발주 모형

Miller and Kelle[6]의 단일품목일 경우의 총비용(jointed total relevant cost)을 계산하는 모형 $JTRC(Q,N)$은 다음과 같다. 식의 유도는 부록 1에 정리되어 있다.

$$JTRC(Q,N) = a\frac{D}{Q} + H_B\frac{Q}{2N} + Z\frac{ND}{Q}$$

$$+ s\frac{D}{Q} + H_S\frac{Q}{2}\left(1 - \frac{D}{P} - \frac{1}{N} + \frac{2D}{NP}\right)$$

위의 총비용을 최소화하는 Q와 N을 구하기 위해 각각에 대하여 편미분을 하고 0으로 놓아 정리하면 다음과 같이 최적 Q^*와 N^*을 구할 수 있다.

$$Q^* = \sqrt{\frac{2DP(a+s)}{H_S(P-D)}}$$

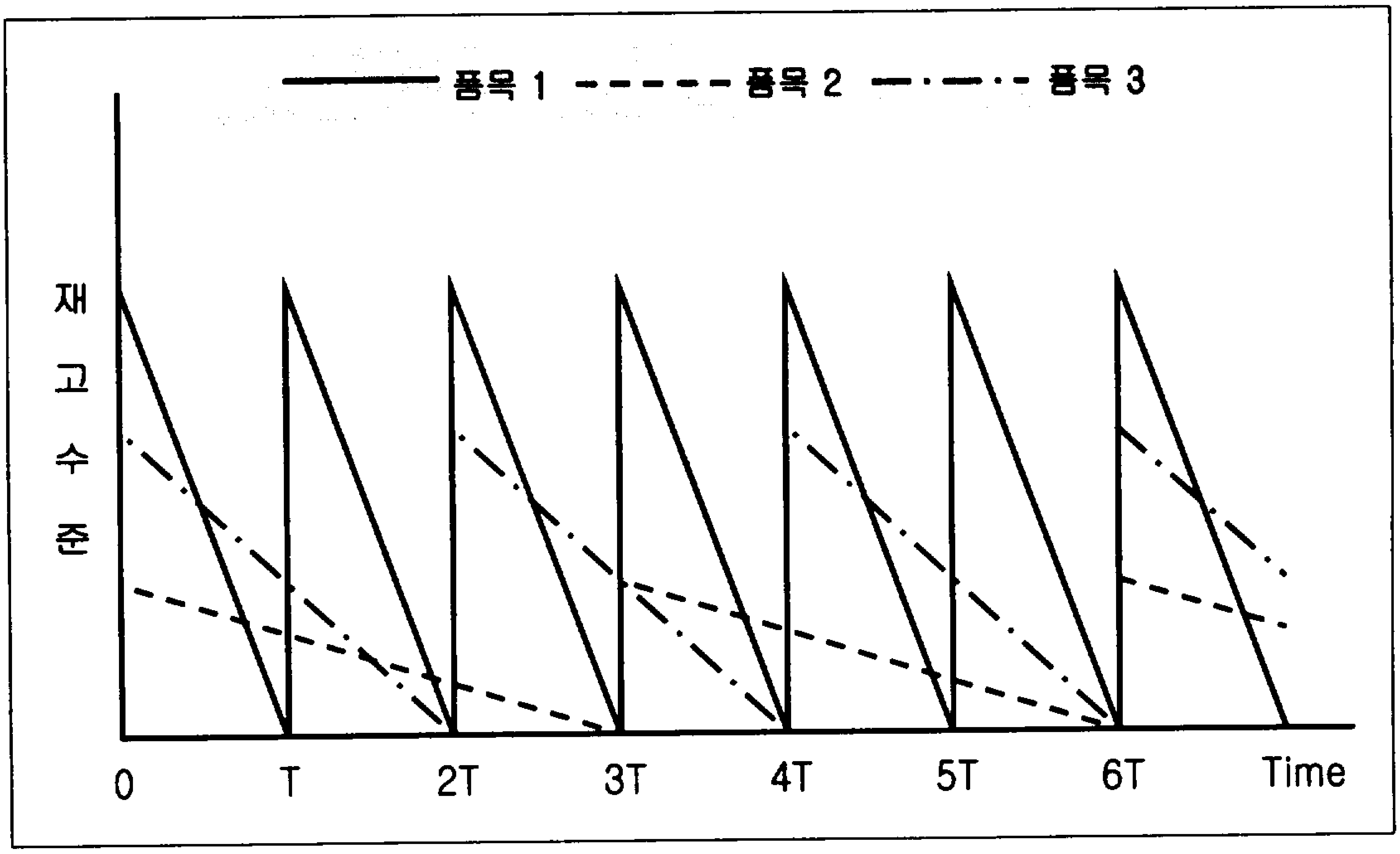
$$N^* = \sqrt{\frac{(a+s)\{(H_B - H_S)P + 2DH_S\}}{ZH_S(P-D)}}$$

3.3 다품목 정기발주 모형

위의 모형을 다품목인 경우로 확장하기 위하여 발주간격 = 발주량/연간수요 ($T = Q/D$)이라는 관계를 이용할 수 있다. T는 공통 발주주기로서 품목이 공통적으로 기간 T마다 발주됨을 의미한다. 그러나 실제적으로는 품목별 발주주기가 상이할 수 있으므로 각 품목별 발주주기는 T의 배수인 $m_i T$가 된다. 총 발주비용은 통합주문의 공통발주비용과 개별 품목당 발주비용의 합이 되어 $A + \sum(a_i/m_i)/T$ 이 되며, 생산준비비용 또한 마찬가지로 $\sum(s_i/m_i)/T$이 된다. 구매자와 공급업자의 각 품목별 재고유지비용을 고려하면 다음과 같이 다품목의 경우로 확장된다.

$$\begin{aligned} JTRC(T, \overline{\boldsymbol{m}}, N) = & \frac{A + \sum a_i/m_i}{T} + \frac{\sum H_{Bi} m_i T D_i}{2N} + \frac{ZN}{T} \\ & + \frac{\sum s_i/m_i}{T} + \frac{\sum H_{Si} m_i T D_i}{2}\left(1 - \frac{D_i}{P_i} - \frac{1}{N} + \frac{2D_i}{NP_i}\right) \end{aligned} \qquad (1)$$

여기서 m_i 는 정수로서, 공통 발주주기 (T)의 배수이다. 즉 다품목 상황에서 품목별로 발주주기를 달리할 수 있으며 품목 i 는 매 m_i 번째 발주주기에 발주된다는 것이다. <그림 1>에서 보는바와 같이 예를 들어 $m_2 = 3$ 이면 품목 2는 3 T기간마다 발주가 된다는 것이며 $m_1 = 1$ 이면 품목 1은 매 T기간마다 발주한다는 의미이다.



<그림 1> m_i 의 값에 따른 품목별 재고 형태

($m_1 = 1$, $m_2 = 3$, $m_3 = 2$ 인 경우)

<그림 1>에서 나타난 것처럼 품목별로 상이한 발주주기를 가지는 경우, 식 (1)을 최소화하는 T를 구하기 위해 $\partial JTRC(T, \overline{\boldsymbol{m}}, N) / \partial T = 0$으로 두어 특정 $\overline{\boldsymbol{m}}$ 및 N에 대한 최적 T를 다음과 같이 구할 수 있다. 즉,

$$-\frac{1}{T^2}\left\{A + ZN + \sum \frac{(a_i + s_i)}{m_i}\right\}$$

$$+\frac{1}{2}\sum\left\{\frac{H_{Bi} m_i D_i}{N} + \frac{H_{Si} m_i D_i}{N}\left(N - \frac{ND_i}{P_i} - 1 + \frac{2D_i}{P_i}\right)\right\} = 0$$

또는

$$T^*(\overline{\boldsymbol{m}}, N) = \sqrt{\frac{2\{A + ZN + \sum(a_i + s_i)/m_i\}}{\sum\left\{\frac{H_{Bi}m_iD_i}{N} + \frac{H_{Si}m_iD_i}{N}\left(N - \frac{ND_i}{P_i} - 1 + \frac{2D_i}{P_i}\right)\right\}}}$$

이 된다. 식을 간략화하기 위하여

$$I_i = \frac{H_{Bi}D_i}{N} + \frac{H_{Si}D_i}{N}\left(N - \frac{ND_i}{P_i} - 1 + \frac{2D_i}{P_i}\right)$$

이라고 두면

$$T^*(\overline{\boldsymbol{m}}, N) = \sqrt{\frac{2\left\{A + ZN + \sum\frac{a_i + s_i}{m_i}\right\}}{\sum m_iI_i}} \qquad (2)$$

으로 정리되고, 식(2)를 식(1)에 대입하면 총비용은 다음과 같이 $\overline{\boldsymbol{m}}$와 N에 대한 수식만으로 간단하게 정리할 수 있다.

$$JTRC(\overline{\boldsymbol{m}}, N) = \sqrt{2\left(A + ZN + \sum\frac{a_i + s_i}{m_i}\right)\sum m_iI_i} \qquad (3)$$

식(3)을 최소화하는 $\overline{\boldsymbol{m}}$과 N을 구하는 것은 다음의 수식을 최소화하는 $\overline{\boldsymbol{m}}$과 N을 구하는 것과 동일하다.

$$F(\overline{\boldsymbol{m}}, N) = \left(A + ZN + \sum \frac{a_i + s_i}{m_i}\right) \sum m_i I_i \tag{4}$$

식(4)를 최소화하는 것은 다음의 이유로 어려운 문제(비선형 정수계획법)이다 : (1) m_i 간에는 서로 상호 작용이 있다. 즉, m_i값이 다른 m_j값에 영향을 준다. (2) m_i값은 정수이어야 한다. 따라서 실용적으로 쉽게 해를 찾는 발견적 해법(heuristic method)으로 $\overline{\boldsymbol{m}}$의 정수제약을 무시하고 식(4)를 최소화하는 $\overline{\boldsymbol{m}}$를 구하기 위해 다음과 같이 식(4)를 편미분 하면,

$$\frac{\partial F(\overline{\boldsymbol{m}}, N)}{\partial m_j} = -\frac{a_j + s_j}{m_j^2} \sum m_i I_i + \left(A + ZN + \sum \frac{a_i + s_i}{m_i}\right) I_j = 0$$

또는

$$m_j^2 = \frac{(a_j + s_j)}{I_j} \frac{\sum m_i I_i}{A + ZN + \sum \frac{a_i + s_i}{m_i}} \tag{5}$$

으로 정리가 가능하다.

$$[C(N)]^2 = \frac{\sum m_i I_i}{A + ZN + \sum \frac{a_i + s_i}{m_i}} \tag{6}$$

이라고 하면, $j \neq k$에 대하여 다음과 같은 식을 구할 수 있다.

$$m_k^2 = \frac{(a_k + s_k)}{I_k}[C(N)]^2 \qquad (7)$$

만일 품목 j 에 대해서

$$\frac{(a_j + s_j)}{I_j} < \frac{(a_k + s_k)}{I_k}$$

이면 m_j(실수해)는 m_k(실수해)보다 작을 것이다. 따라서 각 품목의 $(a_j + s_j)/I_j$값을 계산하여 그 중에서 가장 작은 값을 가지는 품목의 m_j값은 가장 작은 값을 가지며 또한 정수라는 조건 때문에 $m_j = 1$을 가지게 된다. 만약 품목 번호 1번이 $(a_j + s_j)/I_j$ 의 계산결과 값이 가장 작도록 순서가 정해졌다면 $m_1 = 1$이다.

식(5)와 식(6)에 의해서 m_j는

$$m_j = \sqrt{\frac{a_j + s_j}{I_j}}\, C(N) \qquad (8)$$

이 된다.

$C(N)$을 계산하기 위하여 식(8)을 $m_1 = 1$이라는 것을 이용하여 식(6)에 대입하여 정리하면 분자부분은 다음과 같이 정리가 된다.

$$\sum_{i=1}^{n} m_i I_i = I_1 + C(N) \sum_{i=2}^{n} \sqrt{\frac{a_i + s_i}{I_i}} \times I_i$$

$$= I_1 + C(N) \sum_{i=2}^{n} \sqrt{(a_i + s_i) I_i} \tag{9}$$

마찬가지로 분모부분은

$$\sum_{i=1}^{n} \frac{a_i + s_i}{m_i} = (a_1 + s_1) + \frac{1}{C(N)} \sum_{i=2}^{n} \sqrt{\frac{I_i}{a_i + s_i}} (a_i + s_i)$$

$$= (a_1 + s_1) + \frac{1}{C(N)} \sum_{i=2}^{n} \sqrt{(a_i + s_i) I_i} \tag{10}$$

이 되므로 식(9)과 식(10)을 식(6)에 대입하면

$$[C(N)]^2 = \frac{I_1 + C(N) \sum_{i=2}^{n} \sqrt{(a_i + s_i) I_i}}{A + ZN + (a_1 + s_1) + \frac{1}{C(N)} \sum_{i=2}^{n} \sqrt{(a_i + s_i) I_i}}$$

을 얻게된다. 이를 정리하면 아래와 같이 $C(N)$을 얻을 수 있다.

$$C(N) = \sqrt{\frac{I_1}{A + ZN + (a_1 + s_1)}} \tag{11}$$

따라서 식(11)을 식(8)에 대입하여 정리하면 나머지 품목들의 m_j값을 식 (12)와 같이 얻게 된다.

$$m_j = \sqrt{\frac{(a_j + s_j)}{I_j} \cdot \frac{I_1}{A + ZN + (a_1 + s_1)}} \quad j = 2, 3, \ldots, n \tag{12}$$

m_j는 1보다 같거나 큰 정수값을 가져야 하므로, 1보다 작은 경우는 1이라고 두고 1보다 큰 경우는 반올림을 하여 값을 취한다.

3.4 품목별 발주주기가 동일한 정기발주 모형

품목별 발주주기가 동일한 경우는 3.3의 일반적인 경우에 비해 특수한 경우라고 할 수 있다. 식(1)에서 모든 품목에 대하여 m_i가 모두 1 일 때, 즉 모든 품목이 매 T기간마다 동시에 함께 발주되는 경우이다. 그렇게 되면 총비용 식 (1)은 다음과 같이 수정된다.

$$\begin{aligned} JTRC(T,N) = & \frac{A + \sum a_i}{T} + \frac{\sum H_{Bi} T D_i}{2N} + \frac{ZN}{T} \\ & + \frac{\sum s_i}{T} + \frac{\sum H_{Si} T D_i}{2}\left(1 - \frac{D_i}{P_i} - \frac{1}{N} + \frac{2D_i}{NP_i}\right) \end{aligned} \tag{13}$$

식(13)에서 특정 N에 대한 최적 T를 구하기 위하여 식(13)을 T에 대하여 편미분 하여 0으로 두면,

$$T^*(N) = \sqrt{\frac{2\{A + ZN + \sum(a_i + s_i)\}}{\sum I_i}} \tag{14}$$

이 되고 식(14)를 식(13)에 대입하여 정리하면 총비용은 다음과 같이 N에 대해서 간단하게 정리된다.

$$JTRC(N) = \sqrt{2\left\{A + ZN + \sum(a_i + s_i)\right\}\sum I_i} \tag{15}$$

N이 정수이기 때문에 품목별 발주주기가 동일한 경우 식(15)의 최적해를 다음과 같이 반복적인 절차를 통해 쉽게 구할 수 있다.

1) 운송횟수의 초기치로 $N=1$이라 둔다.

2) $JTRC(N) < JTRC(N+1)$ 이면 $JTRC(N)$이 최적이 된다.

3) 그렇지 않으면 $N=N+1$로 하여 단계 2)로 간다.

제 4 장 반복적 해법

품목별 공통발주 및 공통운송을 하는 정책에서 품목별 발주주기가 일정하지 않은 일반적인 경우 총비용인 식(1)을 최소화하는 T, $\overline{\boldsymbol{m}}$, N을 찾는 것은 비선형 정수계획법으로 최적해를 구하는 것은 매우 어려운 문제이며, 따라서 본 연구에서는 쉽게 해를 찾을 수 있는 발견적 해법(heuristic method)을 제시하는데 주안점을 두었다. 다음의 반복적 해법을 이용하여 해를 계산 할 수 있으며, 계산은 엑셀과 같은 스프레드시트에서 가능하다.

1) 운송횟수의 초기치로 $N=1$을 가정한다.

2) $(a_i+s_i)/I_i$를 구하고 가장 작은 값을 가지는 품목을 $m_1=1$로 한다.

3) m_j, $j=2,3,...n$ 을 식(12)를 이용하여 계산한 후 반올림으로 정수값을 구한다.

4) 식 (1)을 이용하여 $JTRC(\overline{\boldsymbol{m}},N)$를 계산한다.

5) 충분히 N을 증가시키고 난 후, $JTRC(\overline{\boldsymbol{m}},N)$ 중에서 가장 작은 값을 취하고, 이때의 T^*, Q_i^* 를 구한다.

제 5 장 수치 예

개발된 통합모형의 유용성을 보이기 위하여 Ha and Kim[4]의 수치 예를 확장하여 4개의 품목에 대한 모수가 <표 1>과 같다고 하자.

<표 1> 수치 예(단위 : $)

공통 발주비용 A = 25, 운송건당 운송비용 Z = 25

	품목 1	품목 2	품목 3	품목 4
연간수요율(D_i)	12,000	5,000	8,000	300
연간생산율(P_i)	48,000	20,000	32,000	1,200
공급업자의 1회 생산준비비용(s_i)	300	600	600	500
구매자의 1회 발주비용(a_i)	50	20	10	25
구매자 연간 재고유지비용(H_{Bi})	25	15	20	30
공급업자 연간 재고유지비용(H_{Si})	10	5	10	15

<표 1>의 예를 이용하여 반복적 해법에 의해 해를 구한 결과가 <표 2>에 나타나 있다.

<표 2> 수치 예의 반복적 절차에 의한 최적해

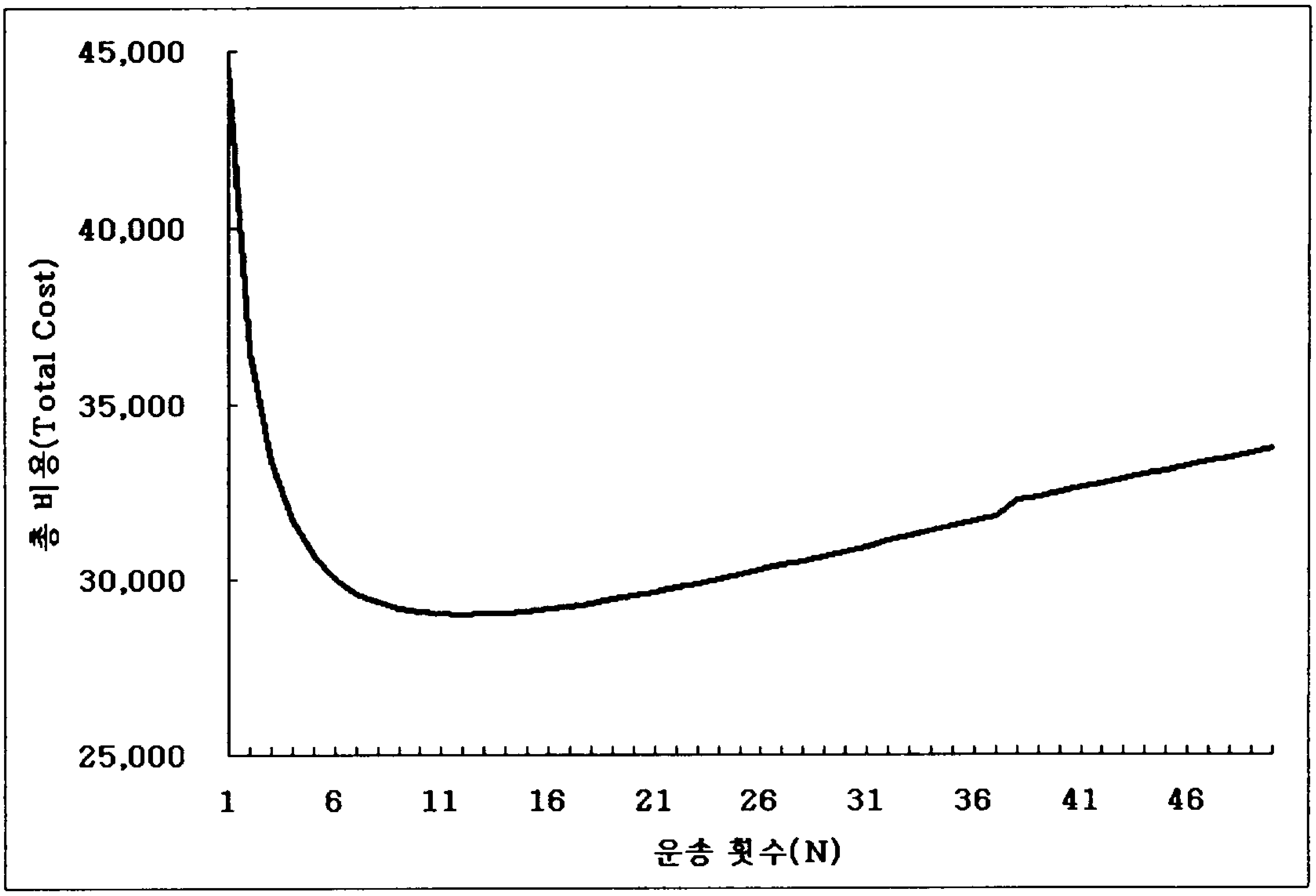
	품목 1	품목 2	품목 3	품목 4
m_i	1	2	1	5
T	0.1172			
Q_i	1406	1172	937	176
$JTRC$	$ 29,014.72			

<표 3> 수치 예의 반복적 절차에 의한
총 비용, 공통 발주주기, 품목별 발주량의 변화

N	m_1	m_2	m_3	m_4	$JTRC$	T	Q_1	Q_2	Q_3	Q_4
1	1	3	2	7	44524.82	0.0443	532	665	709	93
2	1	2	2	6	36490.79	0.0618	742	618	989	111
3	1	2	2	6	33363.50	0.0691	829	691	1105	124
4	1	2	1	6	31709.71	0.0935	1122	935	748	168
5	1	2	1	6	30712.26	0.0982	1178	982	785	177
6	1	2	1	5	30026.38	0.1032	1239	1032	826	155
7	1	2	1	5	29617.88	0.1064	1276	1064	851	160
8	1	2	1	5	29351.32	0.1090	1308	1090	872	164
9	1	2	1	5	29181.25	0.1114	1336	1114	891	167
10	1	2	1	5	29079.63	0.1135	1362	1135	908	170
11	1	2	1	5	29028.25	0.1154	1385	1154	923	173
12	**1**	**2**	**1**	**5**	**29014.72**	**0.1172**	**1406**	**1172**	**937**	**176**
13	1	2	1	5	29030.36	0.1188	1426	1188	951	178
14	1	2	1	4	29051.20	0.1223	1467	1223	978	147
15	1	2	1	4	29104.43	0.1238	1485	1238	990	149
16	1	2	1	4	29172.46	0.1252	1502	1252	1002	150
17	1	2	1	4	29252.57	0.1266	1519	1266	1013	152
18	1	2	1	4	29342.64	0.1279	1535	1279	1023	153
19	1	2	1	4	29440.96	0.1292	1550	1292	1033	155
20	1	2	1	4	29546.18	0.1304	1565	1304	1043	156

<표 3>은 수치 예의 반복적 절차에 의해 N에 따라 총 비용 및 공통 발주주기 및 개별품목의 발주량의 변화를 나타낸 것이다. $N=1$일 때 품목 1이 다른 품목에 비하여 $(a_i+s_i)/I_i$의 값이 가장 적기 때문에 $m_1=1$로 두고 나머지 품목에 대하여 m_i를 각각 계산한다. 이런 절차를 통해 결과적으로 공통 발주주기 T기간동안의 공통 운송횟수가 $N=12$일 때 총비용이 가장 적게 나오는 것을 알 수 있다. 그리고 이때의 공통 발주주기는 0.1172년, 약 6주가 되고 품목 1과 품목 3은 6주마다, 그리고 품목 2는 12주, 품목 4는 30주마다 발주가 된

다. 공통 운송횟수별 총 비용의 변화를 그래프로 표현하면 <그림 2>와 같다.



<그림 2> 운송 횟수별 총비용 변화곡선

동일한 예제에 대하여 Miller and Kelle[6]가 제안한 단일품목인 경우의 모형을 적용하면, 각 품목별 개별발주 및 개별운송에 대한 비용을 모두 통합한 결과 총비용이 $ 32,059.07 으로 <표 4>의 결과가 나온다. 따라서 본 논문의 해법을 적용한 공통발주 및 공통운송 정책이 총 비용 면에서 약 9.50%($ 3,044.35)가 감소하는 것으로 나왔다.

<표 4> 품목별 개별발주 및 개별운송인 경우의 정량발주모형 적용 결과

	품목 1	품목 2	품목 3	품목 4	합계
T_i	0.0913	0.2622	0.1455	0.5700	
Q_i	1095	1311	1164	171	
N	6	9	7	7	
$JTRC_i$	11,684.67	6,686.62	11,179.15	2,508.63	**$ 32,059.07**

또한 품목별 발주주기가 동일한 정기발주모형의 경우, 즉 모든 품목이 공통 발주주기 T 마다 함께 발주되는 경우에는 모든 품목에 대하여 $m_i = 1$로 하여 해를 구하면 공통 운송횟수가 $N = 15$일 때 총 비용이 가장 적으며 이 때의 총비용은 $JTRC =$ $ $ 31,713.02 이다. 모든 품목이 기간 T마다 함께 발주 되었을 때는 품목별 발주주기가 일정하지 않은 공통발주 및 공통운송모형보다 총비용이 약 9.30%($ 2,698.30) 증가하였다. <표 5>에 결과가 나와있다.

<표 5> 품목별 발주주기가 동일한 경우의 정기발주모형 적용 결과

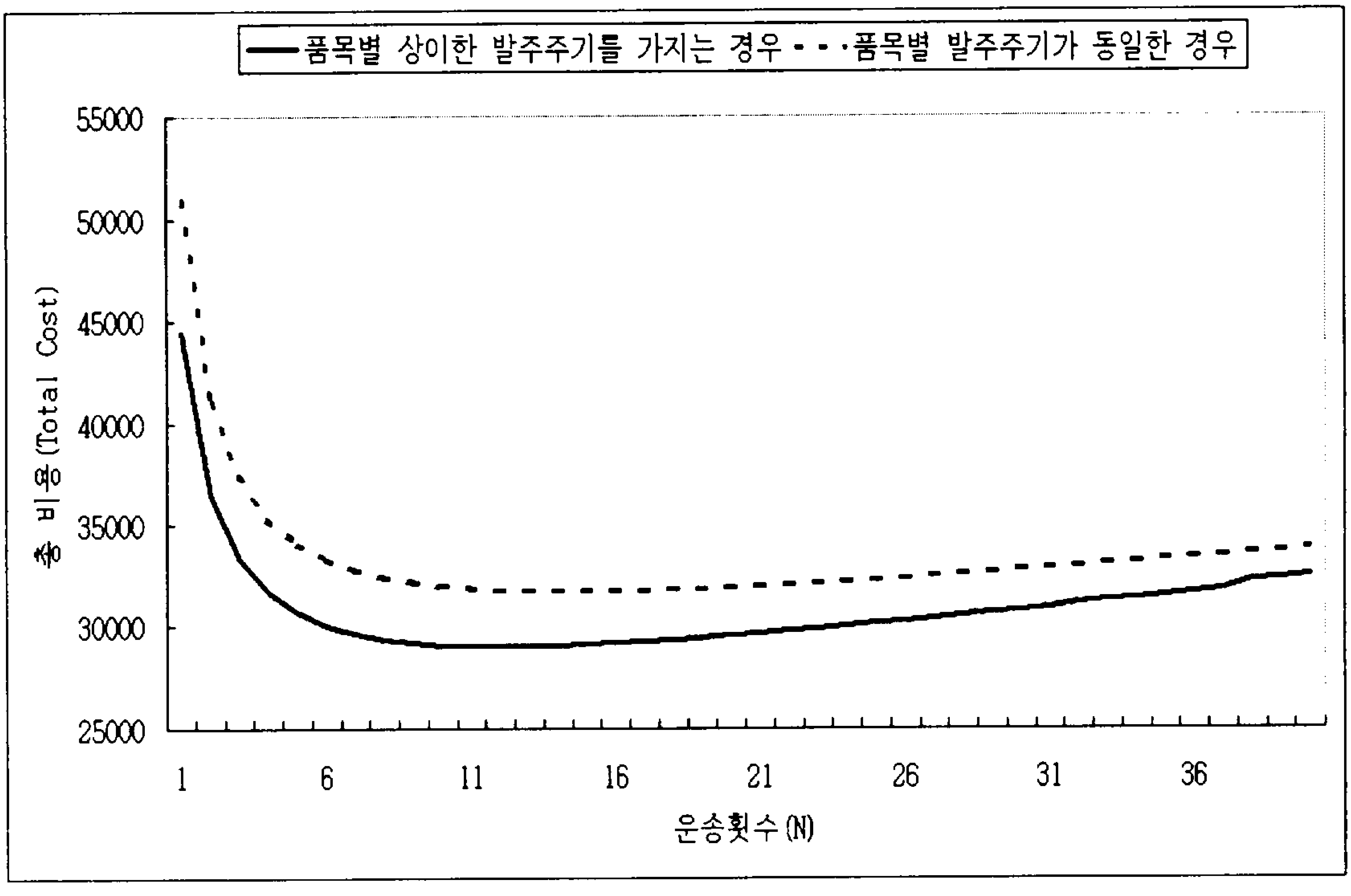
	품목 1	품목 2	품목 3	품목 4
T	0.1580			
Q_i	1896	790	1264	47
$JTRC$	**$ 31,713.02**			

다품목으로 발주하는 세 가지 정책을 각각 적용하여 비교해보니 본 모형이 가장 낮은 비용을 가지고 있음을 알 수 있다. <표 6>은 수치 예에 대한 세 가지 정책별 총 비용의 차이를 비교한 것이고 <그림 3>은 공통발주 및 공통운송 정책에서 발주주기가 일정한 경우와 일정하지 않은 경우 대하여 총 비용의 변

화를 도식적으로 표현한 것이다.

<표 6> 다품목 발주정책별 총 비용 비교 (백분율)

	개별발주 및 개별운송	공통발주 및 공통운송	
		발주주기 동일	상이한 발주주기
비용($)	32,059.07	31,713.02	29,014.72
본모형 대비 증가율	10.49 %	9.30 %	-



<그림 3> 다품목 발주정책에 따른 총비용의 변화
(공통운송 및 공통발주인 경우)

JIT 구매환경에서는 구매자-공급업자가 하나의 공급사슬을 구성하고 공급사슬 전체의 이익을 위해 두 집단이 협력해야 한다. 그러나 일반적으로 전통적인 구매환경에서는 구매자 또는 공급업자가 다른 집단에 대하여 협상력(구매력)을 가지는 경우, 자신의 이익을 최대화 하다보면 상대방에게는 더 많은 비용이나 손실을 가져다 주는 경우가 자주 발생한다. 이러한 경우는 공급사슬 전체의 비용이 본 모형보다 오히려 증가하게 된다.

본문의 동일한 예제를 구매자가 협상력(구매력)을 가지는 경우에 적용하면, <표 7>에 나와 있는 것처럼 구매자 측의 비용을 최소화하는 공통발주주기 T에 의하여 계산한 결과 $N=14$일 때 구매자의 비용은 $ 6,107.61이 되고 공급업자의 비용은 $ 25,607.41이 되며 이 정책의 총 비용은 $ 31,715.02이 된다. 본 모형과 비교해보면 구매자의 비용은 약 7.62%($ 503.97)가 감소하였으나 공급업자의 비용은 오히려 약 14.3%($ 3.204.27)가 증가하였고 통합비용도 약 9.31%($ 2,700.30)가 증가하였다.

<표 7> 전통적인 구매정책과의 비교

( )안은 통합모형 대비 증가율

	통합모형인 경우	구매자가 협상력을 가지는 경우		공급업자가 협상력을 가지는 경우	
구매자 비용	6,611.59	6,107.61	(-7.62 %)	7,882.99	(19.23 %)
공급업자 비용	22,403.14	25,607.41	(14.3 %)	21,714.74	(-3.07 %)
통합 비용	29,014.72	31,715.02	(9.31 %)	29,597.73	(2.01 %)

반대로 공급업자가 협상력을 가지는 경우, $N=10$일 때 공급업자의 비용은 $ 21,714.74 이며 구매자의 비용은 $ 7,882.99 이다. 이 정책의 총 비용은 $ 29,597.73 이다. 본 모형에 비해, 공급업자의 비용은 약 3.07%($ 688.40) 감소하였으나 구매자의 비용은 약 19.23%($ 1,271.41)가 증가하였고 통합비용도 약

2.01%($ 583.01) 증가하였다. (이 모형의 유도는 부록 4에 있다.) 결론적으로 전통적인 구매정책에 비해 JIT 구매환경 하에서의 통합정책이 공급사슬 전체의 이익을 증가시킴으로써 오히려 양 집단에게 유용함을 가져다 주는 것을 알 수 있다.

위의 수치 예는 Ha and Kim[4]의 한가지 예를 확장하여 각 모형을 비교한 것이다. 다음 장에서는 이 모형을 좀 더 다양한 수치 예를 가지는 경우로 확장하여 모의문제를 이용해 각 모형의 유용성을 비교해보았다.

제 6 장 모의문제를 이용한 비교분석

본 논문에서 제안한 개발된 통합모형은 총비용을 줄일 수 있는 발견적인 해법이다. 본 모형이 유용하다는 것을 보이기 위해 여기서는 50개의 모의문제에 대하여 앞에서 언급한 다른 모형의 정책들과 비교하여 보았다. 모의문제를 작성하는데 있어서, 통합주문의 공통발주비용 A와 운송 건당 운송비용 Z를 고정시키고, 개별 품목에 해당되는 모수들을 다음의 보기에서 무작위추출(random sampling)하는 방식을 채택하였다.

D_i : {5,000, 10,000, 15,000, 20,000},

P_i : {2, 3, 4, 5} × D_i,

s_i : {250, 500, 750, 1,000},

a_i : {5, 10, 15, 20},

H_{Si} : {5, 10, 15, 20},

H_{Bi} : {2, 3, 4, 5} × H_{Si}

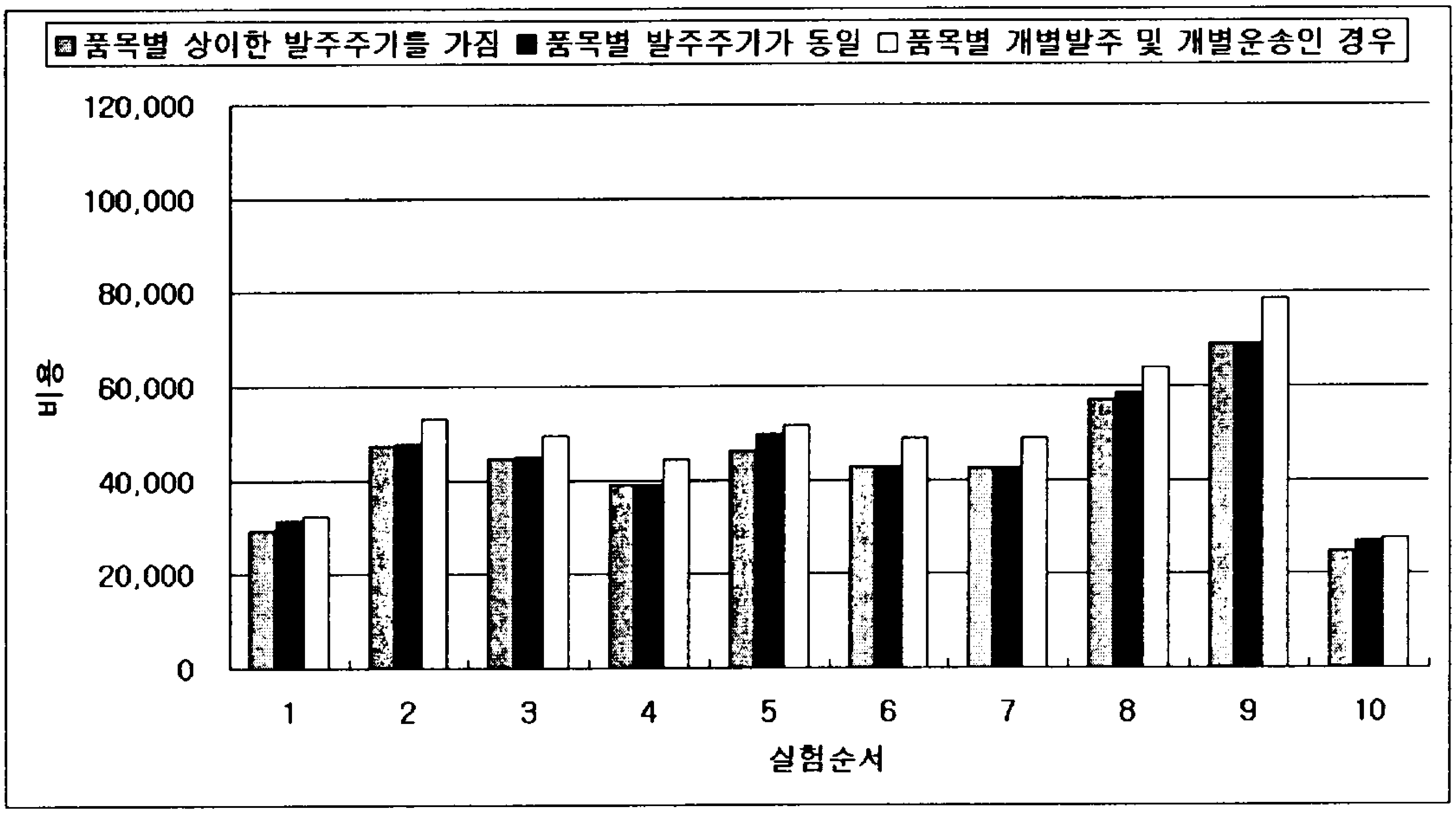
위의 모수들을 품목의 수가 각각 i = {3, 5, 10, 20, 40}인 경우에 적용하여 모의문제를 실험한 후, 동일한 모수가 각 모형에 적용되었을 때 모형마다 최적해가 어떻게 산출되는지를 살펴보았다. 각 조건에서 10번씩 실험을 한 후 각각의 결과에 대한 평균 및 모형별 비용 비교가 <표 8>에 나타나있다. 대부분의 문제에서 품목별 공통발주 및 공통운송시 상이한 발주주기를 가지는 경우를 고려한 정책이 다른 정책에 비해 더 나은 비용절감을 가져온다는 것을 알 수 있다. 또한 본 모형과 비교하여보면 품목 수가 3일 때, 비용 증가율이 품목별 발

주간격이 동일한 경우는 약 2.6% 정도이지만 개별품목 및 개별발주하는 경우에는 약 12.4%가 증가한다. 차선의 정책은 품목별 발주간격이 동일한 경우이다. 그러나 품목의 수가 적을 때는 본 모형과 동일한 결과 값을 가지거나 조금 많을 뿐이지만 품목의 수가 어느 정도로 증가하면 비용 증가율도 늘어난다. 모의실험에서도 품목의 수가 20인 경우, 품목의 수가 3인 경우에 비해 비용 증가율이 3배 이상 커짐을 알 수 있다.

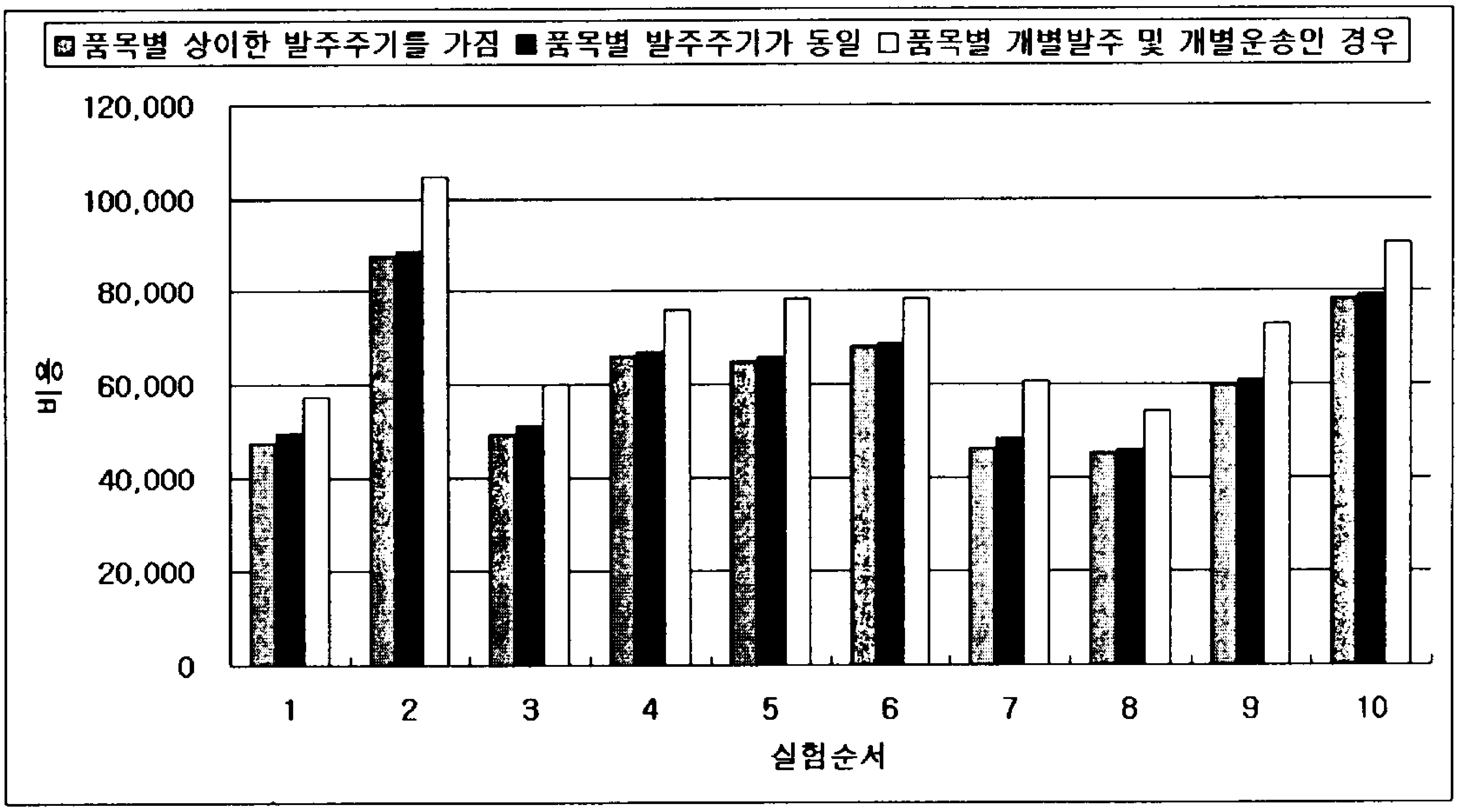
<표 8> 다른 정책과의 총비용 비교(본 모형 기준)

품목 수	정 책	평 균	비용 증가율
3	상이한 발주주기를 가짐	44320.13	-
	발주주기가 동일	45295.72	2.64%
	개별발주 및 개별운송	49901.98	12.36%
5	상이한 발주주기를 가짐	61295.41	-
	발주주기가 동일	62526.61	2.26%
	개별발주 및 개별운송	73279.55	20.17%
10	상이한 발주주기를 가짐	119001.13	-
	발주주기가 동일	128055.97	8.02%
	개별발주 및 개별운송	148217.72	24.55%
20	상이한 발주주기를 가짐	242067.81	-
	발주주기가 동일	258768.38	6.99%
	개별발주 및 개별운송	314262.89	29.95%
40	상이한 발주주기를 가짐	452476.43	-
	발주주기가 동일	483526.77	6.87%
	개별발주 및 개별운송	604762.35	33.66%

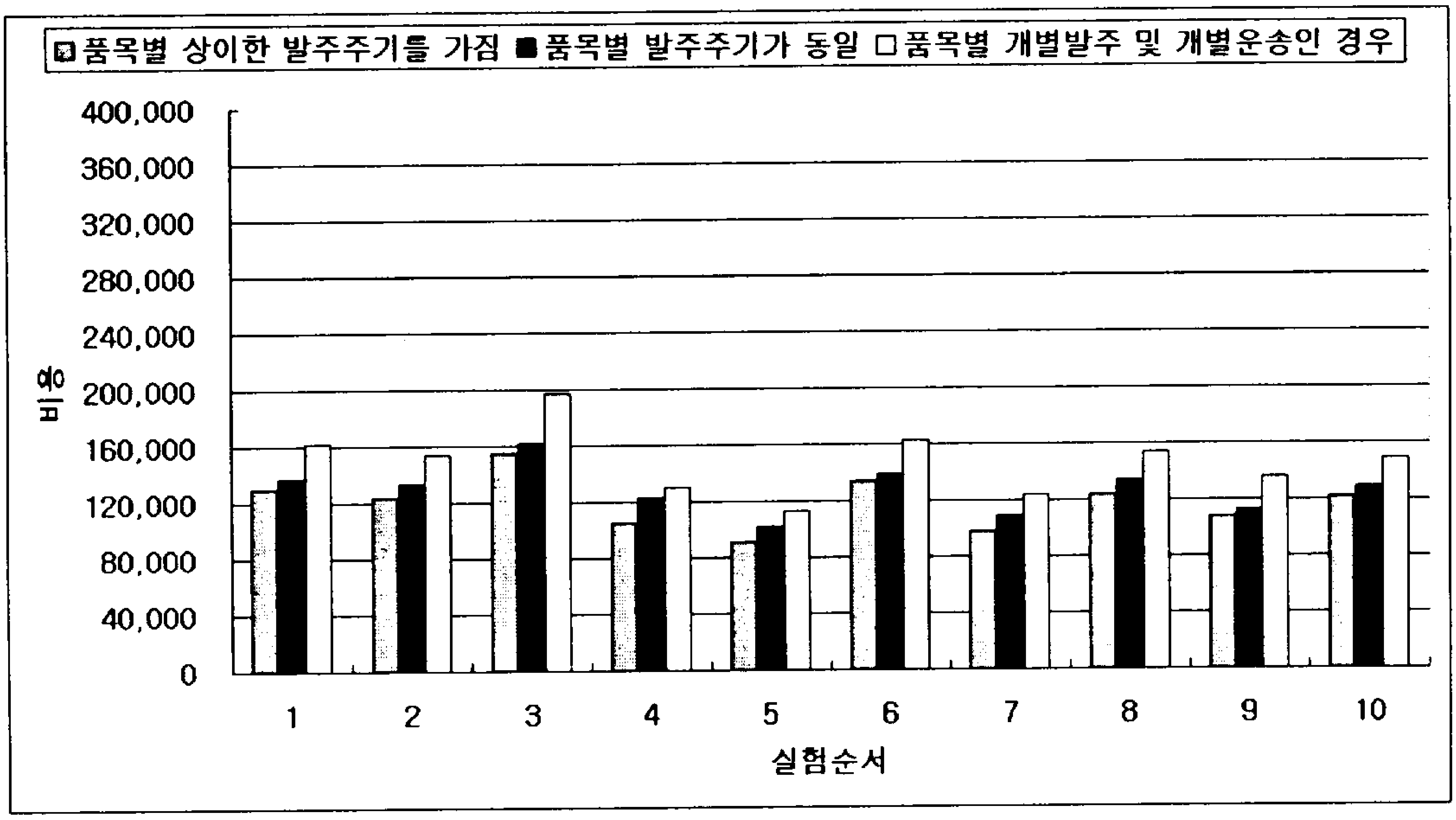
다음의 <그림 4>에서 <그림 8>까지는 각각 품목의 수가 3, 5, 10, 20, 그리고 40인 경우에 모의문제 순서대로 각 모형별 비용 비교를 도식적으로 표현한 것이다. 각 모의문제의 실험결과는 부록 2에 있다.



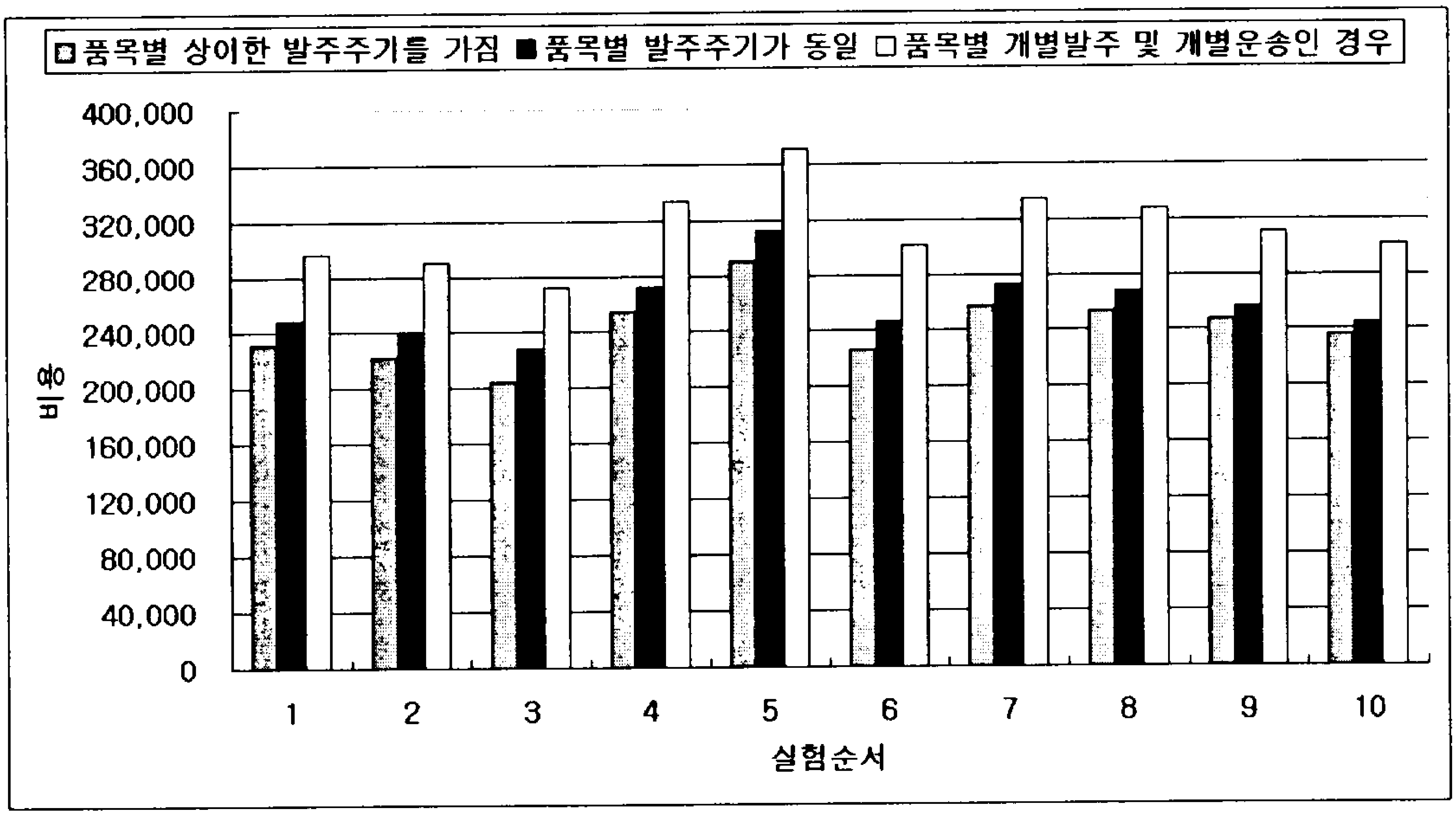
<그림 4> 각 모형별 총비용 비교 (품목 수가 3인 경우)



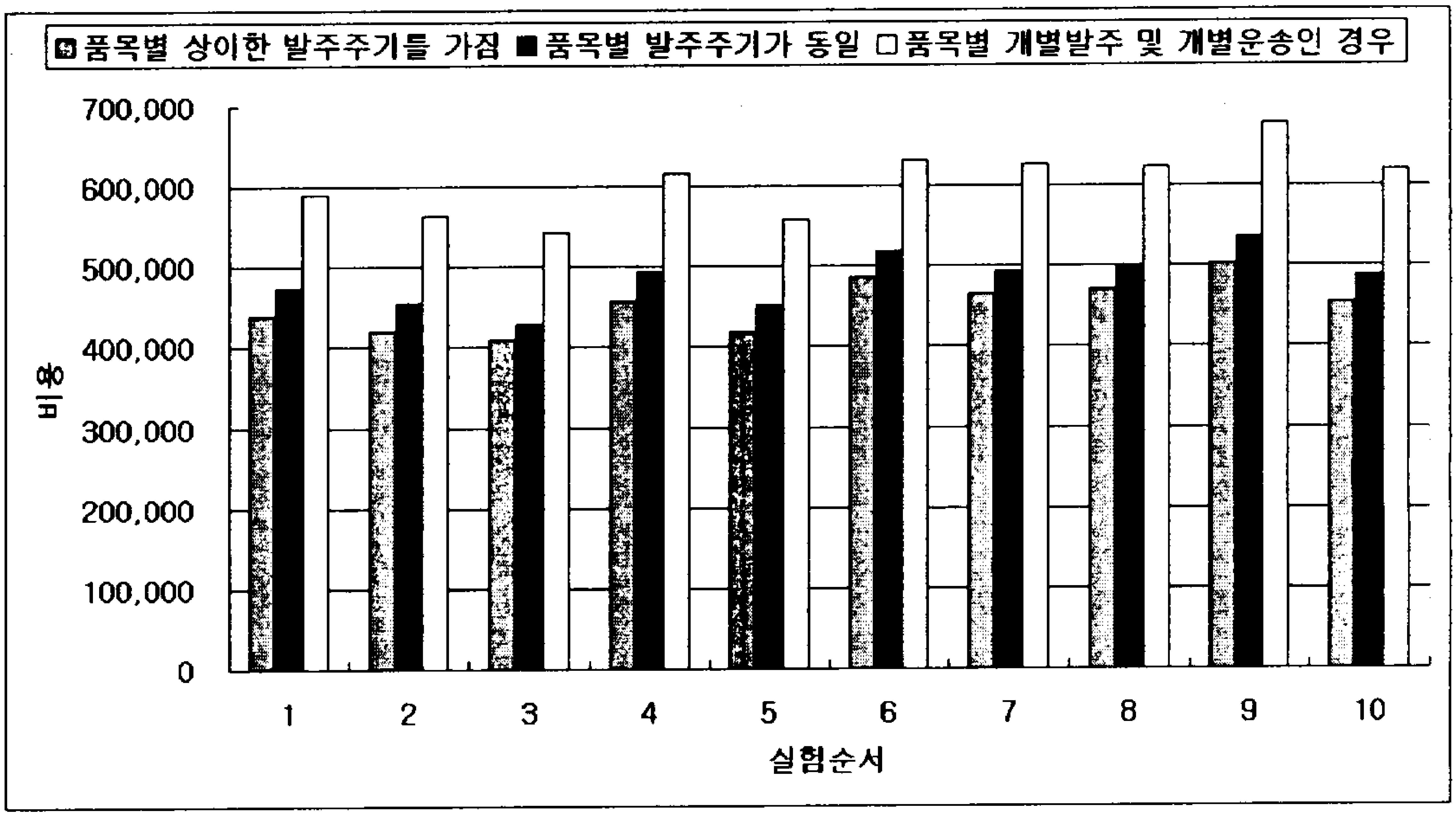
<그림 5> 각 모형별 총비용 비교 (품목 수가 5인 경우)



<그림 6> 각 모형별 총비용 비교 (품목 수가 10인 경우)



<그림 7> 각 모형별 총비용 비교 (품목 수가 20인 경우)



<그림 8> 각 모형별 총비용 비교 (품목 수가 40인 경우)

제 7 장 결　론

본 논문은 효과적인 JIT 구매체계를 구축하기 위해 다품목을 발주하는데 있어 구매자와 공급업자의 다빈도 소량구매정책을 수립하기 위해 필요한 수학적 모형을 제시하였다. 구매자와 공급업자들 간에 다품목을 발주하고 소량 다빈도로 분할하여 운송하는 경우 공급업자에게 개별적인 발주보다는 여러 품목을 공통발주하는 것이 총 발주비용을 절감할 수 있으며, 또한 품목별 개별운송보다는 정기순회혼합적재를 이용하여 여러 품목을 공통운송하는 것이 총운송비를 절감할 수 있다. 본 모형에서는 구매자와 공급업체들에서 발생하는 총발주비용, 총재고유지비용 및 총운송비용을 통합한 총비용(jointed total relevant cost)을 절감하는 통합적 접근법을 사용하였다. 목(lot) 크기를 작게 하여 몇 회에 걸쳐 발송하는 통합된 목분할(lot-splitting) 방식의 정기발주모형이 개발되었고 총비용을 줄일 수 있는 발견적 해법(heuristic method)을 제시하였다. 그리고 이 모형의 유용함을 보이기 위하여 품목별로 개별 발주 및 개별 운송하는 경우나 품목별 발주간격이 모두 동일한 경우와 비교하여 총비용의 절감효과가 크다는 것을 예제 및 모의실험을 통하여 보였다. 또한 구매자와 공급업체들이 바람직한 협력적인 관계를 형성할 때 서로에게 이익이 되는 성과를 살펴보았다.

참 고 문 헌

[1] 안천의 · 이양원, "JIT시스템 하에서 구매-공급자의 관계 유형과 성과에 관한 연구", 논문집 Vol.25 No.1 pp. 171-176, 군산대학교, 1997

[2] Burton, T., JIT/Repetitive Sourcing Strategies: Tying the Knot with Your Suppliers, ,*International Journal of production Research,* Vol. 29, No 4 pp. 38-41, 1988.

[3] Goyal, S.K., A Joint Economic-Lot Size Model for Purchaser and Vendor: A Comment, *Decision Sciences*, Vol. 19, pp.236-241, 1988.

[4] Ha, D. and Kim, S.L., Implementation of JIT purchasing: an integrated approach. *Production Planning and control*, Vol. 8, No. 2, 152-157, 1997.

[5] Inman, R., Quality Certification of Suppliers by JIT manufacturers, *Production & Inventory Management Journal*, Vol. 31, No 2, pp. 58-61, 1990.

[6] Miller, P.A. and Kelle, P., Quantitative Support for Buyer-Supplier Negotiation in Just-In-Time Purchasing, *International Journal of Purchasing and Materials Management*, Spring98, Vol. 34 Issue 2, pp. 25-29, 1998.

[7] Pan, A.C., and Liao, C., An Inventory Model under Just-In-Time

Purchasing Agree, 1989

[8] Ramasesh, R.V., Recasting the Traditional Inventory Model to Implement Just-In-Time Purchasing, *Production and Inventory Management*, Vol. 31, No.1, 1990

부 록 1. 단일품목의 정량발주 모형*

수요가 일정하고 알려져 있는 단일품목을 고려할 때, 수요가 알려져 있기 때문에 안전재고는 고려하지 않는다. 일반적으로 EOQ 모형은 구매자나 공급업자 한쪽만의 총비용을 최소화하는 것이다. Miller and Kelle[6]와 Ha and Kim[4]이 제안한 SSMD(single-setup-multiple-deliveries) 경우, 구매자의 총비용 식은 다음과 같이 구매비용과 재고유지비용 및 운송비용으로 표현되고(식 A1), 공급업자의 총비용 식은 다음과 같이 준비비용과 재고유지비용으로 표현된다(식 A2).

$$JTRC(Q,N)_{Buyer} = a\frac{D}{Q} + H_B\frac{Q}{2N} + Z\frac{ND}{Q} \qquad \text{식 (A1)}$$

$$JTRC(Q,N)_{Supplier} = s\frac{D}{Q} + H_S\frac{Q}{2}\left\{1 - \frac{D}{P} - \frac{1}{N} + \frac{2D}{P}\right\} \qquad \text{식 (A2)}$$

식(A2)는 다음으로부터 유도된다. <그림 A1>에서 공급업자의 주기재고는 다음의 네 개 영역의 합이 된다.

영역 I : $\frac{q^2}{2P}$

영역 II : $\frac{(Q-q)^2}{2P} - (i-2)(i-3)\frac{q^2}{2D} - (i-2)q\left\{\frac{Q-q}{P} - (i-2)\frac{q}{D}\right\}$

영역 III : $\{N-(i-1)\}q\left\{\frac{iq}{D} - \frac{Q-q}{P}\right\}$

영역 IV : $(N-i)\{N-(i-1)\}\frac{q^2}{2D}$

---

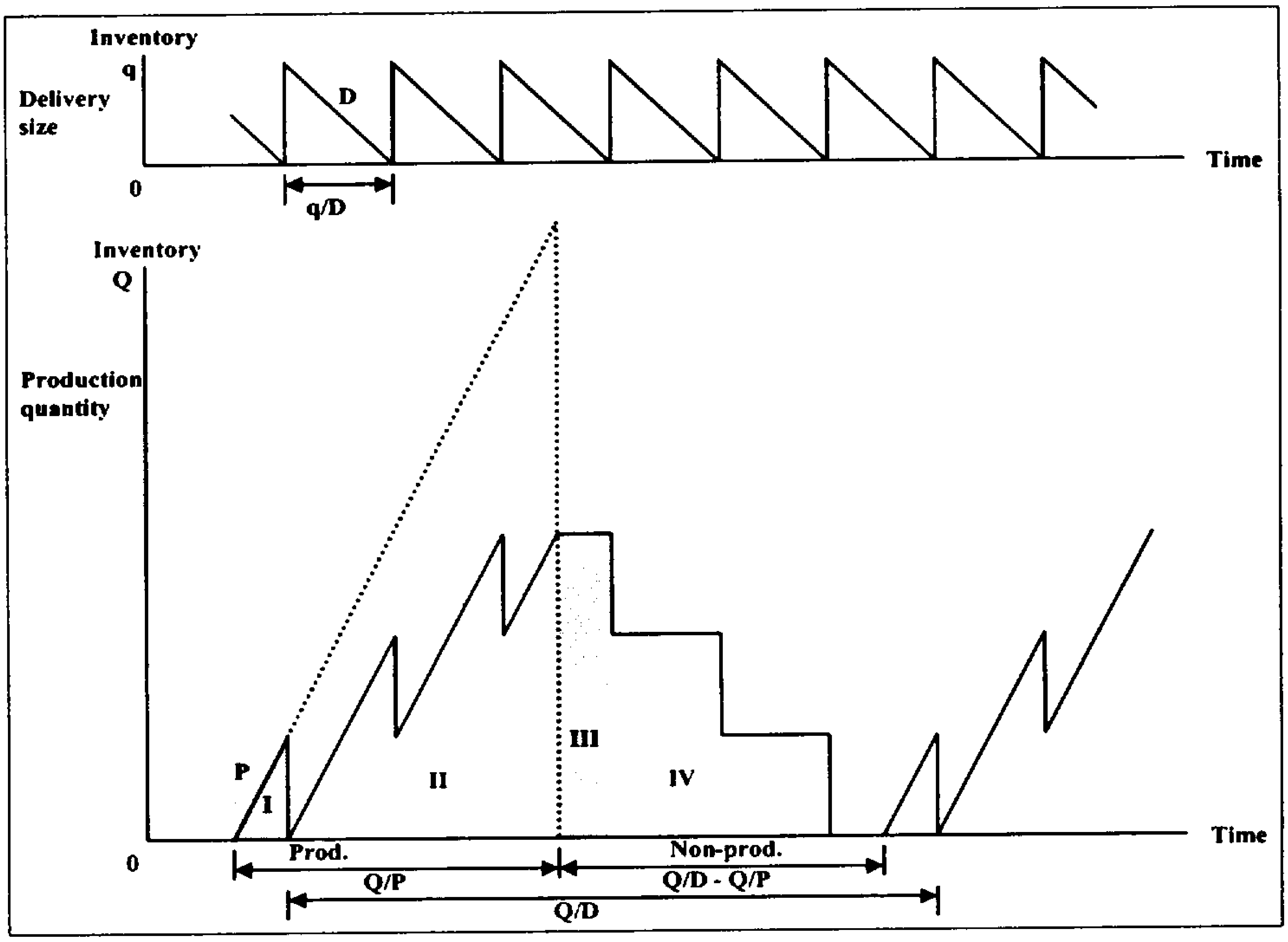
* Ha, D. and Kim, S.L.,Implementation of JIT purchasing: an integrated approach, *Production & Inventory Management Journal*, Vol 31, No 2, 156-157, 1997

그러면 공급업자의 연간 재고유지비용은 다음과 같다.

$$HC_{Supplier} = \frac{D}{Q} H_S(\text{영역}I + \text{영역}II + \text{영역}III + \text{영역}IV) \quad \text{식 (A3)}$$

그러므로 $q = Q/N$ 이라 하면, 식(A3)에 대입하여 정리하면 다음과 같이 단순화된다.

$$HC_{Supplier} = \frac{QH_S}{2}\left\{1 - \frac{D}{P} - \frac{1}{N} + \frac{2D}{NP}\right\} \quad \text{식 (A4)}$$



<그림 A1> SSMD에서의 재고형태

부 록 2. 정책별 모의문제 실험결과

각 모형별 최적해

품목 수	정책	시뮬레이션 번호 1	2	3	4	5	6	7	8	9	10	평균
3	조달주기가 일정하지 않음	29432.65	47298.81	44716.33	39011.31	46155.99	42846.24	42737.28	56957.15	69068.95	24976.66	44320.13
	조달주기가 일정	31453.79	47822.13	45073.47	39011.31	49760.46	42846.24	42737.28	58525.54	69068.95	26858.08	45295.72
	개별발주 및 개별운송	32271.51	53178.62	49620.59	44438.52	51526.09	48796.01	48907.21	63961.24	78739.05	27580.93	49901.98
5	조달주기가 일정하지 않음	47339.22	87988.84	49128.81	65975.26	64813.46	68245.99	46100.42	45300.83	59711.27	78350.04	61295.41
	조달주기가 일정	49618.09	88817.94	51099.63	66760.27	65629.29	68800.07	48258.83	45954.18	61022.41	79305.35	62526.61
	개별발주 및 개별운송	57232.28	104719.56	60095.73	75775.07	78204.05	78226.23	60595.15	54400.10	72964.16	90583.16	73279.55
10	조달주기가 일정하지 않음	129574.17	123818.83	154735.22	105223.71	90782.93	134342.11	98177.75	123844.80	107909.94	121601.84	119001.13
	조달주기가 일정	136855.57	133326.04	162055.41	122794.37	102294.14	139119.15	109007.45	133960.40	112564.95	128582.26	128055.97
	개별발주 및 개별운송	162362.31	154228.10	196658.81	130197.23	113068.50	163306.56	124211.47	153721.91	135601.20	148821.06	148217.72
20	조달주기가 일정하지 않음	230681.73	222198.57	204237.90	254196.68	291242.58	225761.08	257219.26	252675.18	246654.66	235810.45	242067.81
	조달주기가 일정	247786.19	239067.53	228243.20	272256.52	313295.65	245649.78	272557.96	268019.13	256491.47	244316.41	258768.38
	개별발주 및 개별운송	296898.47	291387.89	272239.13	334272.58	370957.59	301584.47	334691.16	327788.65	311152.65	301656.31	314262.89
40	조달주기가 일정하지 않음	439556.92	419619.20	410054.78	456794.94	419079.44	486428.64	466323.42	470114.40	502117.75	454674.84	452476.43
	조달주기가 일정	473968.46	454491.46	428134.31	493159.69	451201.05	517554.88	492684.96	500144.93	536099.57	487828.36	483526.77
	개별발주 및 개별운송	590587.98	562787.33	541829.48	615168.49	559094.05	632014.77	627029.50	622671.07	676417.66	620023.14	604762.35

본 모형 기준 총 비용 증가율(백분율)

품목 수	정책	시뮬레이션 번호 1	2	3	4	5	6	7	8	9	10	평균
3	조달주기가 일정하지 않음	-	-	-	-	-	-	-	-	-	-	-
	조달주기가 일정	6.87%	0.68%	0.80%	0.00%	7.81%	0.00%	0.00%	2.75%	0.00%	7.53%	2.64%
	개별발주 및 개별운송	9.65%	12.43%	10.97%	13.91%	11.63%	13.89%	14.44%	12.30%	14.00%	10.43%	12.36%
5	조달주기가 일정하지 않음	-	-	-	-	-	-	-	-	-	-	-
	조달주기가 일정	4.81%	0.94%	4.01%	1.19%	1.26%	0.81%	4.68%	1.44%	2.20%	1.22%	2.26%
	개별발주 및 개별운송	20.90%	19.01%	22.32%	14.85%	20.66%	14.62%	31.44%	20.09%	22.19%	15.61%	20.17%
10	조달주기가 일정하지 않음	-	-	-	-	-	-	-	-	-	-	-
	조달주기가 일정	5.62%	7.68%	4.73%	16.70%	12.68%	3.56%	11.03%	8.17%	4.31%	5.74%	8.02%
	개별발주 및 개별운송	18.64%	15.68%	21.35%	6.03%	10.53%	17.39%	13.95%	14.75%	20.46%	15.74%	15.45%
20	조달주기가 일정하지 않음	-	-	-	-	-	-	-	-	-	-	-
	조달주기가 일정	7.41%	7.59%	11.75%	7.10%	7.57%	8.81%	5.96%	6.07%	3.99%	3.61%	6.99%
	개별발주 및 개별운송	28.70%	31.14%	33.30%	31.50%	27.37%	33.59%	30.12%	29.73%	26.15%	27.92%	29.95%
40	조달주기가 일정하지 않음	-	-	-	-	-	-	-	-	-	-	-
	조달주기가 일정	7.83%	8.31%	4.41%	7.96%	7.66%	6.40%	5.65%	6.39%	6.77%	7.29%	6.87%
	개별발주 및 개별운송	34.36%	34.12%	32.14%	34.67%	33.41%	29.93%	34.46%	32.45%	34.71%	36.37%	33.66%

부 록 3. 모의문제 프로그램

```
Option Explicit
Dim N As Integer                    '공통발주주기 T기간동안의 공통운송횟수
Private Const A = 25                '통합주문 공통발주비용
Private Const Z = 25                '운송건당 운송비용
Dim Rnd_Array(50) As Double         '품목수 최대 50까지
Dim Item(50), Di(50), Pi(50), Si(50), Ai(50), HSi(50), HBi(50), Ii(50), Ki(50), Mi(50)   As
Double
Dim item_i(50), Di_i(50), Pi_i(50), Si_i(50), Ai_i(50), HSi_i(50), HBi_i(50), Ii_i(50), Ki_i(50),
Mi_i(50), Ni(50), Qi(50), TC(50) As Double
Dim SortCheck(50, 9) As Double
Dim JTRC_Value, Fi_Value, TC_SUM As Double
Dim I_Value As Integer
Dim N_Value As Integer
Dim i As Integer
Dim Min_JTRC As Double

'Array에 품목번호 할당
Private Sub Item_Solve(index As Integer)
    For i = 0 To index
        DoEvents
        Item(i) = Str(i)
    Next i
End Sub

'random 변수 발생
Private Sub RAND_Place(index As Integer)
    Dim Random_Value As Double
    Randomize
```

```
    For i = 0 To index
        DoEvents
        If Rnd > 0# Then
        Random_Value = Format(Rnd, "0.#####")
        Rnd_Array(i) = Random_Value
        End If
    Next i
End Sub

'random 변수에 따른 각 모수 sampling
Private Sub Di_Solve(index As Integer)
    For i = 0 To index
        DoEvents
        If Rnd_Array(i) > 0# And Rnd_Array(i) <= 0.25 Then
            Di(i) = 5000
        ElseIf Rnd_Array(i) > 0.25 And Rnd_Array(i) <= 0.5 Then
            Di(i) = 10000
        ElseIf Rnd_Array(i) > 0.5 And Rnd_Array(i) <= 0.75 Then
            Di(i) = 15000
        Else: Di(i) = 20000
        End If
    Next i
End Sub
Private Sub Pi_Solve(index As Integer)
    For i = 0 To index
        DoEvents
        If Rnd_Array(i) > 0# And Rnd_Array(i) <= 0.25 Then
            Pi(i) = Di(i) * 2
        ElseIf Rnd_Array(i) > 0.25 And Rnd_Array(i) <= 0.5 Then
            Pi(i) = Di(i) * 3
        ElseIf Rnd_Array(i) > 0.5 And Rnd_Array(i) <= 0.75 Then
```

```
            Pi(i) = Di(i) * 4
        Else: Pi(i) = Di(i) * 5
        End If
    Next i
End Sub
Private Sub Si_Solve(index As Integer)
    For i = 0 To index
        DoEvents
        If Rnd_Array(i) > 0# And Rnd_Array(i) <= 0.25 Then
            Si(i) = 250
        ElseIf Rnd_Array(i) > 0.25 And Rnd_Array(i) <= 0.5 Then
            Si(i) = 500
        ElseIf Rnd_Array(i) > 0.5 And Rnd_Array(i) <= 0.75 Then
            Si(i) = 750
        Else: Si(i) = 1000
        End If
    Next i
End Sub
Private Sub Ai_Solve(index As Integer)
For i = 0 To index
        DoEvents
        If Rnd_Array(i) > 0# And Rnd_Array(i) <= 0.25 Then
            Ai(i) = 5
        ElseIf Rnd_Array(i) > 0.25 And Rnd_Array(i) <= 0.5 Then
            Ai(i) = 10
        ElseIf Rnd_Array(i) > 0.5 And Rnd_Array(i) <= 0.75 Then
            Ai(i) = 15
        Else: Ai(i) = 20
        End If
    Next i
End Sub
```

```
Private Sub HSi_Solve(index As Integer)
    For i = 0 To index
        DoEvents
        If Rnd_Array(i) > 0# And Rnd_Array(i) <= 0.25 Then
            HSi(i) = 5
        ElseIf Rnd_Array(i) > 0.25 And Rnd_Array(i) <= 0.5 Then
            HSi(i) = 10
        ElseIf Rnd_Array(i) > 0.5 And Rnd_Array(i) <= 0.75 Then
            HSi(i) = 15
        Else: HSi(i) = 20
        End If
    Next i
End Sub
Private Sub HBi_Solve(index As Integer)
    For i = 0 To index
        DoEvents
        If Rnd_Array(i) > 0# And Rnd_Array(i) <= 0.25 Then
            HBi(i) = HSi(i) * 2
        ElseIf Rnd_Array(i) > 0.25 And Rnd_Array(i) <= 0.5 Then
            HBi(i) = HSi(i) * 3
        ElseIf Rnd_Array(i) > 0.5 And Rnd_Array(i) <= 0.75 Then
            HBi(i) = HSi(i) * 4
        Else: HBi(i) = HSi(i) * 5
        End If
    Next i
End Sub

'Ii 값 계산
Private Sub Ii_Solve(index As Integer, N_Value As Integer)
    Dim temp As Double
    N = N_Value
```

```
    For i = 0 To index
        DoEvents
        temp = ((HBi(i) * Di(i)) / N) + (HSi(i) * Di(i) / N) * (N - (N * Di(i) / Pi(i)) - 1 +
(2 * Di(i) / Pi(i)))
        Ii(i) = Format(temp, "#.####")
    Next i
End Sub

'(ai+si)/Ii 값 산출
Private Sub Ki_Solve(index As Integer)
    Dim temp As Double
    For i = 0 To index
        DoEvents
        temp = (Ai(i) + Si(i)) / Ii(i)
        Ki(i) = Format(temp, "0.######")
    Next i
End Sub

'(ai+si)/Ii 값 정렬 - 작은순서대로
Private Sub sort(index As Integer)
    Dim j, k As Integer
    Dim temp As Double
    Dim temp_sell(9) As String
    For i = 0 To index
        DoEvents
        SortCheck(i, 0) = Item(i)
        SortCheck(i, 1) = Di(i)
        SortCheck(i, 2) = Pi(i)
        SortCheck(i, 3) = Si(i)
        SortCheck(i, 4) = Ai(i)
        SortCheck(i, 5) = HSi(i)
```

```
        SortCheck(i, 6) = HBi(i)
        SortCheck(i, 7) = Ii(i)
        SortCheck(i, 8) = Ki(i)
    Next i
        For i = 0 To index
        For j = 0 To index
            DoEvents
            If Ki(i) < Ki(j) Then
                temp = Ki(j)
                Ki(j) = Ki(i)
                Ki(i) = temp
                For k = 0 To 9
                    temp_sell(k) = SortCheck(j, k)
                    SortCheck(j, k) = SortCheck(i, k)
                    SortCheck(i, k) = temp_sell(k)
                Next k
            End If
        Next j
    Next i
End Sub
Private Sub Change_Sort(index As Integer)
    For i = 0 To index
        DoEvents
        Item(i) = SortCheck(i, 0)
        Di(i) = SortCheck(i, 1)
        Pi(i) = SortCheck(i, 2)
        Si(i) = SortCheck(i, 3)
        Ai(i) = SortCheck(i, 4)
        HSi(i) = SortCheck(i, 5)
        HBi(i) = SortCheck(i, 6)
        Ii(i) = SortCheck(i, 7)
```

```
        Ki(i) = SortCheck(i, 8)
    Next i
End Sub

'C(N) 계산
Private Sub CN(index As Integer, N_Value As Integer)
    Dim CN_Value, Mi_Value As Double
    N = N_Value
    Mi(0) = 1                          '(ai+si)/Ii 값중 가장 작은 품목i를 mi=1로 설정
    CN_Value = (Ii(0) / (A + (Z * N) + (Ai(0) + Si(0)))) ^ (1 / 2)
    For i = 1 To index
        DoEvents
        Mi_Value = ((Ai(i) + Si(i)) / Ii(i)) ^ (1 / 2) * CN_Value         '각 mi 계산
        Mi(i) = Round(Mi_Value, 1)
    Next i
End Sub
'FC 계산
Private Sub Fi_Solve(index As Integer, N_Value As Integer)
    Dim ASMi, Mii As Double
    Dim temp As Double
    N = N_Value
    i = ASMi = Mii = 0
    Do
        temp = CDbl((Ai(i) + Si(i)) / Mi(i))
        ASMi = temp + ASMi
        Mii = Mi(i) * Ii(i) + Mii
        i = i + 1
        DoEvents
    Loop While i < (index + 1)
    Fi_Value = (A + Z * N + ASMi) * Mii
End Sub
```

```
'JTRC 최적해 계산 - 본모형
Private Sub JTRC_Solve()
    JTRC_Value = (2 * Fi_Value) ^ (1 / 2)
    If JTRC_Value < Min_JTRC Then
        Min_JTRC = JTRC_Value
    End If
End Sub
Private Sub Form_Load()
    Mi_Result.Enabled = False
    Rst_TC.Enabled = False
End Sub
'입력변수 N, I 및 결과 초기화
Private Sub refresh_Click()
    Txt_Rst.Text = ""
    Txt_N.Text = ""
    Txt_I.Text = ""
    TxtN_Rst.Text = ""
    TxtM_Rst.Text = ""
    Txt_Sum = ""
    TC_SUM = 0
End Sub
'입력된 N값에 따른 각 함수 호출
Private Sub Process(index As Integer, N_Value As Integer)
    N = N_Value
    Call Item_Solve(index)
    Call RAND_Place(index)
    Call Di_Solve(index)
    Call RAND_Place(index)
    Call Pi_Solve(index)
    Call RAND_Place(index)
    Call Si_Solve(index)
```

```
    Call RAND_Place(index)
    Call Ai_Solve(index)
    Call RAND_Place(index)
    Call HSi_Solve(index)
    Call RAND_Place(index)
    Call HBi_Solve(index)
    Call Ii_Solve(index, N)
    Call Ki_Solve(index)
    Call sort(index)
    Call Change_Sort(index)
    Call CN(index, N)
    Call Fi_Solve(index, N)
    Call JTRC_Solve
End Sub
'입출력부분
Private Sub Rst_N_Click()
     Dim N_Value, index As Integer
     Dim temp, label As String
     Min_JTRC = 1000000000000000#
     If Txt_N.Text = "" Then
          MsgBox "N의 숫자를 입력해주세요", vbOKOnly, "Error Message"
          Mi_Result.Enabled = False
          Rst_TC.Enabled = False
          Exit Sub
     Else: N_Value = CInt(Txt_N.Text)
     End If
     If Txt_I.Text = "" Then            '품목의 수 입력
          MsgBox "I의 숫자를 입력해주세요", vbOKOnly, "Error Message"
          Mi_Result.Enabled = False
          Exit Sub
     Else: I_Value = CInt(Txt_I.Text) - 1
```

```
        Call Process(I_Value, 1)
    End If
    Mi_Result.Enabled = True
    Rst_TC.Enabled = True
    label = "i = " + Txt_I.Text + "일 때" + "," & vbCrLf
    For i = 0 To I_Value              ' 품목 수에 따른 random sampling된 모수 generate
        DoEvents
        label = label + "ITem" + Str(Item(i)) + ","
        item_i(i) = Item(i)
    Next i
    label = label & vbCrLf + "Di,"
    For i = 0 To I_Value
        DoEvents
        label = label + Str(Di(i)) + ","
        Di_i(i) = Di(i)
    Next i
    label = label & vbCrLf + "Pi,"
    For i = 0 To I_Value
        DoEvents
        label = label + Str(Pi(i)) + ","
        Pi_i(i) = Pi(i)
    Next i
    label = label & vbCrLf + "Si,"
    For i = 0 To I_Value
        DoEvents
        label = label + Str(Si(i)) + ","
        Si_i(i) = Si(i)
    Next i
    label = label & vbCrLf + "Ai,"
    For i = 0 To I_Value
        DoEvents
```

```
        label = label + Str(Ai(i)) + ","
        Ai_i(i) = Ai(i)
    Next i
    label = label & vbCrLf + "HSi,"
    For i = 0 To I_Value
        DoEvents
        label = label + Str(HSi(i)) + ","
        HSi_i(i) = HSi(i)
    Next i
    label = label & vbCrLf + "HBi,"
    For i = 0 To I_Value
        DoEvents
        label = label + Str(HBi(i)) + ","
        HBi_i(i) = HBi(i)
    Next i
    label = label & vbCrLf
    For i = 0 To I_Value
        DoEvents
        Ii_i(i) = Ii(i)
        Ki_i(i) = Ki(i)
        Mi_i(i) = Mi(i)
    Next i
    For index = 1 To N_Value                    'N에 따른 JTRC 출력
        DoEvents
        If index = 1 Then
            temp = "JTRC (i = " & Str(I_Value + 1) & " and " + " N = " & Str(index) &
"),"  & Str(JTRC_Value) + "," & vbCrLf
        Else
        Call Ii_Solve(I_Value, index)
        Call Ki_Solve(I_Value)
        Call sort(I_Value)
```

```
        Call Change_Sort(I_Value)
        Call CN(I_Value, index)
        Call Fi_Solve(I_Value, index)
        Call JTRC_Solve
        temp = temp + "JTRC (i = " & Str(I_Value + 1) & " and " + "N = " & Str(index)
& ")," & Str(JTRC_Value) + "," & vbCrLf
        End If
    Next index
    TxtN_Rst.Text = Str(Min_JTRC)
    Txt_Rst.Text = Txt_Rst.Text & vbCrLf & vbCrLf + label + temp
End Sub
'결과를 텍스트파일로 저장
Private Sub Save_Btn_Click()
    Dim Str
    CommonDialog1.DialogTitle = "파일 저장"
    CommonDialog1.Filter = "텍스트 파일(*.txt)|*.txt"
    CommonDialog1.ShowSave
    Open CommonDialog1.FileName For Output As #1
        Str = Txt_Rst.Text
        Print #1, Str
    Close #1
End Sub
```

부 록 4. 전통적 구매정책인 경우의 모형

본문의 식(1)에서, 구매자의 비용과 공급업자의 비용을 나누면 각각 식(A5)와 식(A6)으로 나눌 수 있다.

$$TC_B(T, \overline{\boldsymbol{m}}, N) = \frac{A + \sum a_i/m_i + ZN}{T} + \frac{\sum H_{Bi} m_i TD_i}{2N} \quad \textbf{(A5)}$$

$$TC_S(T, \overline{\boldsymbol{m}}, N) = \frac{\sum s_i/m_i}{T} + \frac{\sum H_{Si} m_i TD_i}{2}\left(1 - \frac{D_i}{P_i} - \frac{1}{N} + \frac{2D_i}{NP_i}\right) \quad \textbf{(A6)}$$

(a) 구매자가 협상력을 가지는 경우

구매자의 비용을 최소화하는 $T, \overline{\boldsymbol{m}}, N$ 을 구하기 위해 식(A5)를 T에 대하여 편미분 하여 0으로 놓고 정리하면,

$$T_B^* = \sqrt{\frac{(A + ZN + \sum a_i/m_i)2N}{\sum H_{Bi} m_i D_i}} \quad \textbf{(A7)}$$

이 된다. 식(A7)을 식(A5)에 대입해서 정리하면,

$$TC_B(\overline{\boldsymbol{m}}, N) = \sqrt{\frac{2(A + ZN + \sum a_i/m_i)\ \sum H_{Bi} m_i D_i}{N}} \quad \textbf{(A8)}$$

을 얻게 된다. 식(A8)을 최소화하는 $\overline{\boldsymbol{m}}, N$을 구하는 것은 아래의 식을 최소화하는 $\overline{\boldsymbol{m}}, N$를 구하는 것과 마찬가지이다.

$$F_B(\overline{\boldsymbol{m}}, N) = (A + ZN + \sum a_i / m_i) \frac{\sum H_{Bi} m_i D_i}{N} \tag{A9}$$

운송횟수의 초기치를 $N=1$로 두고 본문에서의 알고리즘을 동일하게 적용하면 다음과 같다.

1) 식(A9)를 편미분한 결과 ; $m_j^2 = \dfrac{a_j}{H_{Bj} D_j} \dfrac{\sum H_{Bi} m_i D_i}{A + ZN + \sum a_i / m_i}$

2) $C_B(N)^2 = \dfrac{\sum m_i H_{Bi} D_i}{A + ZN + \sum a_i / m_i}$ 이라고 둔다.

3) j ≠ k 일 때, $m_k^2 = \dfrac{a_k}{H_{Bk} D_k} C_B(N)^2$

4) $\dfrac{a_i}{H_{Bi} D_i}$를 구하고 가장 작은 값을 가지는 품목을 $m_1 = 1$로 한다.

5) $C_B(N) = \sqrt{\dfrac{H_{B1} D_1}{A + ZN + a_1}}$을 구한다.

6) $m_j = \sqrt{\dfrac{a_j}{H_{Bj} D_j}} C_B(N)$이므로 $m_j, j = 2, 3, \ldots, n$을 다음 식을 이용하여 계산한 후 반올림으로 정수값을 구한다.

$$m_j = \sqrt{\frac{a_j}{H_{Bj}D_j}\frac{H_{B1}D_1}{A+ZN+a_1}} \quad \textbf{(A10)}$$

7) 식(A10)을 식(A7)에 대입하여 $T_B{}^*$를 계산한다.

8) 단계 6)과 7)을 이용하여 식(A5)와 식(A6)을 합하여 통합비용을 계산한다.

9) 충분히 N을 증가시키고 난 후 통합비용이 가장 작은 값을 취한다.

(b) 공급업자가 협상력을 가지는 경우

공급업자의 비용을 최소화하는 $T, \overline{\boldsymbol{m}}, N$ 을 구하기 위해 식(A6)을 T에 대하여 편미분 하여 0으로 놓고 정리하면,

$$T_S{}^* = \sqrt{\frac{2\sum \frac{s_i}{m_i}}{\sum H_{Si}m_iTD_i\left(1-\frac{D_i}{P_i}-\frac{1}{N}+\frac{2D_i}{NP_i}\right)}} \quad \textbf{(A11)}$$

이 된다. 식을 간략화 하기 위해

$$K_i = H_{Si}D_i\left(1-\frac{D_i}{P_i}-\frac{1}{N}+\frac{2D_i}{NP_i}\right)$$

이라고 하고 식(A11)을 식(A6)에 대입해서 정리하면,

$$TC_S(\overline{\boldsymbol{m}}, N) = \sqrt{2\sum \frac{s_i}{m_i}\sum m_iK_i} \quad \textbf{(A12)}$$

을 얻게 된다. 식(A12)를 최소화하는 $\overline{\boldsymbol{m}}, N$을 구하는 것은 아래의 식을 최소화하는 $\overline{\boldsymbol{m}}, N$를 구하는 것과 마찬가지이다.

$$F_S(\overline{\boldsymbol{m}}, N) = \sum \frac{s_i}{m_i} \sum m_i K_i \qquad \textbf{(A13)}$$

운송횟수의 초기치를 $N=1$로 두고 본문에서의 알고리즘을 동일하게 적용하면 다음과 같다.

1) 식(A13)을 편미분한 결과 ; $m_j^2 = \dfrac{s_j}{K_j} \dfrac{\sum m_i K_i}{\sum s_i / m_i}$

2) $C_S(N)^2 = \dfrac{\sum m_i K_i}{\sum s_i / m_i}$ 이라고 둔다.

3) j ≠ k 일 때, $m_k^2 = \dfrac{s_k}{K_k} C_S(N)^2$

4) $\dfrac{s_i}{K_i}$를 구하고 가장 작은 값을 가지는 품목을 $m_1 = 1$로 한다.

5) $C_S(N) = \sqrt{\dfrac{K_1}{s_1}}$을 구한다.

6) $m_j = \sqrt{\dfrac{s_j}{K_j}} C_S(N)$이므로 $m_j, j = 2,3, \ldots, n$을 다음 식을 이용하여 계산한 후 반올림으로 정수값을 구한다.

$$m_j = \sqrt{\frac{s_j}{K_j} \frac{K_1}{s_1}} \qquad \textbf{(A14)}$$

7) 식(A14)를 식(A11)에 대입하여 $T_S{}^*$를 계산한다.

8) 단계 6)과 7)을 이용하여 식(A5)와 식(A6)을 합하여 통합비용을 계산한다.

9) 충분히 N을 증가시키고 난 후 통합비용이 가장 작은 값을 취한다.

ABSTRACT

An Integrated Inventory Model for Multi-Item in Just-In-Time Purchasing

Kim, Yong Chul
Major in Industrial Engineering
Dept. of Industrial Engineering
The Graduate School of
Hansung University

This paper addresses the necessity of integration between buyer and suppliers for effective implementation of JIT purchasing in a multi-item environment.

An integrated inventory model of facilitating multiple shipments in small lots is developed. Also, an iterative solution procedure is developed to find the order(contract) interval for each item and number of shipments between buyers and suppliers simultaneously.

We show by example that when the integrated policy is adopted by both buyer and suppliers in a cooperative manner, both parties can be benefited.

감사의 글

그리 쉽지 않은 대학원 생활을 애써 이끌어주신 많은 분들에게 감사의 마음을 전합니다.

다양한 지식과 학문으로 지도해주시고 여기까지 이끌어주신 김대홍 교수님에게 깊은 감사를 드립니다. 또한 대학원 생활동안 충고와 조언으로 마지막까지 관심을 기울여주신 박명환 교수님, 정병용 교수님, 이재득 교수님, 원형규 교수님, 유재건 교수님, 홍윤기 교수님, 위남숙 교수님, 홍정완 교수님께도 감사드립니다.

지난 2년 동안 같이 생활해주고 도와준 대학원 동기 곽대월, 이정욱, 양계령, 김영준, 양승태에게도 고맙다는 말을 전합니다. 그리고 나중에나마 같이 생활하고 학회까지 참여해준 후배 김재영에게도 고마움을 전합니다. 또한 선배로서 조교의 자리에서 많은 도움을 주신 안수영 선배님에게도 고맙다는 말을 전합니다.

선배로서 도움을 주지 못했음에도 많은 일을 도와주고 마지막 논문까지 열심히 응원해준 진영, 기범, 성용, 영일, 현규, 민경, 미이를 비롯한 많은 후배들에게 정말로 감사를 드립니다.

마지막으로 저의 결정을 믿고 묵묵히 도와주시고 지원해주신 아버지, 어머니, 누님, 동생에게 깊은 애정을 드립니다.

2001년 12월

김 용 철