

의류 인터넷 쇼핑몰에서 브랜드를 고려한 상품 입고 및 재배치 방법 연구

송용욱
연세대학교 원주캠퍼스 정경대학 경영학부 교수
(yusong@yonsei.ac.kr)

안병혁
경상대학교 경영정보학과 교수
(bahn@gnu.ac.kr)

.....

의류 인터넷 쇼핑몰들은 판매 상품의 포장과 배송을 위한 상품 창고를 운영하고 있다. 상품들은 동일 브랜드끼리 상자에 담긴 후 선반에 일렬로 보관된다. 상품의 반출 및 상품 관리의 편의상, 상자들은 동일 브랜드끼리 묶여져 선반에 진열되어 있어야 한다. 따라서, 새로운 상품들이 입고될 경우 상품들은 동일 브랜드끼리 상자에 담긴 후 선반 위에 있는 기존 상자들 중에서 동일 브랜드의 상자 옆에 배치되어야 한다. 그런데, 선반 위의 빈 곳이 새로 입고되는 상자를 넣을 수 있을 만큼 충분하지 않다면, 옆의 다른 브랜드의 상자들을 옆으로 밀어서 공간을 확보한 후 새로운 상자를 배치함으로써 동일 브랜드 상자끼리 붙어있도록 해야 한다.

우리의 문제는 이와 같이 새로운 상품을 입고할 때 동일 브랜드의 상자들끼리 붙어 있도록 하면서 다른 브랜드의 상자들을 옆으로 옮길 경우, 그 횟수를 최소화하는 것이다. 이 문제의 최적해를 구하기 위해서 우리는 이 문제를 우선 정수계획법으로 모형화하였다. 그런데, 정수계획법 문제는 분기한정법(Branch and Bound) 기법으로 해결하여야 하나, 그 경우 문제해결 시간이 너무 오래 걸리는 문제가 발생한다. 따라서, 본 연구에서는 위 재배치 문제를 할당 문제(Assignment Problem)로 완화하여 모형화함으로써 만족할만한 준최적해를 구하는 방법론을 제시하고, 실험에 의하여 그 타당성을 검토하였다. 또한, 이 방법론 하에서 실제 의류 인터넷 쇼핑몰의 컴퓨팅 환경을 고려할 때 해결 가능한 문제의 최대 크기를 도출하고, 그 크기 이내에서 입고 계획을 생성하는 시스템을 구현하였다.

.....

논문접수일 : 2010년 04월 30일 논문수정일 : 2010년 05월 24일 게재확정일 : 2010년 06월 15일 교신저자 : 안병혁

1. 서론

의류 인터넷 쇼핑몰 상품 창고에서 상품들은 동일 브랜드끼리 상자에 담긴 후 선반에 일렬로 보관된다. 상품 관리의 편의상, 상자들은 동일 브랜드끼리 묶여져 선반에 진열되어 있어야 한다. 따라서 새로운 상품들이 입고될 경우 상품들은 동일 브랜드끼리 상자에 담긴 후 선반 위에 있는 기존 상자들 중에서 동일 브랜드의 상자 옆에 배치되어

야 한다. 그런데, 선반 위의 빈 곳이 새로 입고되는 상자를 넣을 수 있을 만큼 충분하지 않다면, 옆의 다른 브랜드의 상자들을 옆으로 밀어서 공간을 확보한 후 새로운 상자를 배치함으로써 동일 브랜드 상자끼리 붙어 있도록 해야 한다. 우리의 문제는 이와 같이 새로운 상품 상자를 입고할 때 동일 브랜드의 상자들끼리 붙어 있도록 하면서, 다른 브랜드의 상자를 옆으로 옮기는 횟수를 최소화하는 것이다.

위와 같은 본 연구의 문제는 기본적으로 상품의 반출을 용이하게 하기 위한 상품의 배치를 정하는 문제이지만, 상품 배치 기준이 브랜드 별로 상품을 묶는데 있다는 점에서 상품 반출 빈도, 상품 크기나 무게 등을 기준으로 하는 기존 연구들과는 차별성을 갖는다. 이러한 문제를 풀기 위한 방법론을 제시하고, 그 효율성을 검토하기 위해 본 논문은 다음과 같이 구성된다.

제 2장에서는 우선 관련 문헌들을 살펴보고, 제 3장에서는 문제정의, 최적화 모델링, 문제완화 등을 포함한 상품 배치 방법론을 설명한다. 제 4장에서는 제안된 방법론의 타당성을 실험에 의하여 검토하고, 마지막으로 제 5장에서 결론을 맺도록 한다.

2. 문헌 연구

주문된 상품들을 창고에서 꺼내는 과정을 효율화하기 위하여 주문 상품반출(Order picking)에 관한 기존 연구들은 크게 두 가지 측면에서 접근하고 있다. 하나는 재고 상품(SKU, Stock keeping unit)을 창고의 특정 위치에 배치(assign)하는 배치문제이며, 다른 하나는 SKU들의 배치가 주어진 상태에서 각 주문 상품을 반출하기 위해 창고를 돌아다니고, 상품 정보를 파악하거나, 기타 필요한 행동들을 하기 위해 소요되는 시간을 최소화하기 위해 유사한 주문끼리 묶는(Order batching) 문제이다(Brynzer, 1996; Koster et al., 2007).

주문묶음(Order batching) 문제는 창고 내 상품 적재 선반의 배치 및 이에 따른 복도의 위치와 길이 등을 고려하여 주문 상품반출 지시서에 따른 한번의 작업동안 반출자(Order picker)가 움직여야 하는 동선의 길이의 평균을 최소화하는 문제로 모형화되며, 이러한 모형은 일반적으로 풀기가 매우 어려운 문제로 알려져 있다. 따라서, 지금까지의

연구들은 주로 주어진 상황을 고려하여 만족스러운 시간 내에 만족스러운 해를 찾아내는 휴리스틱의 개발에 초점을 맞추어 왔다. 주문묶음(Order batching) 휴리스틱과 관련하여, Elsayed and Unal (1989)은 EQUAL 알고리즘, SL 알고리즘, MAX-SAV 알고리즘, CWright 알고리즘 등을 기술하였으며, Goetschalckx and Ratliff(1988)는 최적의 정지 회수와 정지 위치, 그리고 매 정지마다 꺼내야 할 상품을 정하기 위한 알고리즘을 제시하였다. 또한 실제 상황에서는 종종 배치 정책과 주문묶음 정책을 조합하기도 한다.

주문 상품반출 시간은 각 상품의 특징에 따라 SKU를 배치함으로써 감소될 수 있다. SKU를 배치하는 전통적인 방법은 상품의 사용률이나 회전율을 바탕으로 배치하는 것이다. 예를 들어 가장 자주 주문되는 상품을 입고 및 출고 장소의 제일 앞에 배치하는 것이다. Heskett(1963)은 SKU 당 필요한 저장 공간(cube)과 SKU의 주문 빈도 간의 비율로 표현되는 주문 당 입방면적 지수(COI, Cube-per-order index)를 제안하였다. COI 배치 정책에 따르면, SKU를 COI에 따라 정렬한 후, 가장 작은 COI를 갖는 SKU를 입고고 장소의 제일 앞에 배치하는 것이다.

상품 배치 전략은 크게 지정배치(dedicated storage), 임의배치(randomized storage), 클래스 별 배치(class-based storage) 등 세 가지로 분류된다(Koster et al., 2007; Merkurjev, 2009; Roodbergen, 2001). 지정배치 전략에서는 각 상품은 미리 정해진 고정 장소에 배치된다. 임의배치 전략에서 상품들은 임의로 선택된 장소에 배치된다. 클래스 별 배치 전략에서는 SKU를 몇 개의 클래스로 분류하고, 각 클래스에 대해 고정된 장소를 지정한 후, 각 클래스 장소 내에서 임의배치 전략을 적용한다. White and Kinney(1982)의 지적에 따르면, 클래스 별 배

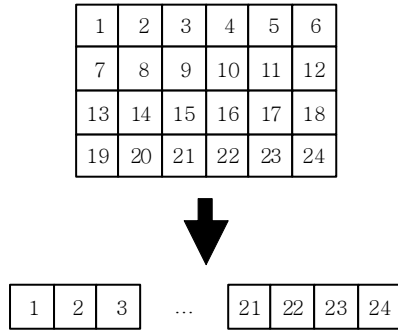
치 전략에 비하여 임의 배치 전략이 저장 공간을 덜 필요로 한다. 반면에, 임의배치 전략에 비하여 클래스 별 배치 전략은 주문 상품반출을 위한 이동 시간을 감소시킨다.

상품들을 브랜드 별로 묶어서 배치하는 본 연구의 문제는 클래스 별 배치 전략을 채택한 창고에 관한 문제이지만, 클래스 별 배치 장소가 고정되지 않았다는 점에서 기존 연구들과는 또 다른 특징을 갖는다. 클래스의 동적 배치에 관련된 문제는 컴퓨터 프로그래밍에서 선형 리스트 유지를 위한 순차 메모리 할당(Sequential allocation) 문제와 유사하다(Knuth, 1997). 순차 메모리 할당 문제는 고정된 메모리 공간 내에 동적으로 크기가 변하는 여러 개의 메모리 블록을 유지하는 문제이다. 각 메모리 블록의 크기가 변할 때마다 각 메모리 블록들은 전체 공간 내에서 재배치되어야 한다. 그러나 이 순차 메모리 할당 문제에서도 메모리 이동의 횟수를 최소화하는 알고리즘은 개발하지 못했으며, 게다가, 메모리 블록 내 바이트(byte)의 순서를 유지해야 하는 점은 동일 브랜드 내 상자들의 순서를 유지할 필요는 없는 본 연구의 문제와는 또 다른 점이다.

3. 상품 배치 방법론

3.1 문제 정의

문제를 명확히 정의하기 위하여 상품 상자가 놓이는 창고 내의 각 장소를 셀(Cell)이라고 부르자. 상자를 보관하는 선반은 다층의 격자형 선반이므로, 원칙적으로 셀의 배치 형태는 2차원 배열의 형태이다. 그러나 2차원 공간 상에서 새로운 상자의 추가에 따라 동일 브랜드의 상자 영역을 전후 및 좌우로 확장할 경우 그 확장 결과가 단순한 직사

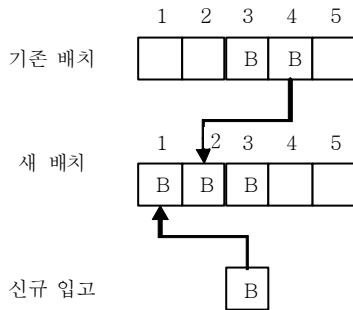


<그림 1> 셀의 배치 형태

각형 형태가 아닌 아주 복잡한 형태가 되어 관리를 오히려 복잡하게 할 수 있다. 또한 창고관리자가 셀의 배치 형태를 1차원 배열로 파악하여 동일 브랜드 셀의 시작점부터 끝점까지의 두 가지 값을 기억하여 상자들을 기억하고 관리하는 것이 간편하다. 따라서 본 문제에서는 셀의 배치 형태를 일렬로 늘어선 1차원의 배열로 파악하고, 동일 브랜드의 상자끼리 묶는 것도 이 1차원 배열 상에서 묶이도록 한다<그림 1>. 그러면, 본 연구문제의 목표는 새로 상품 상자를 입고할 때 1차원의 각 셀에 동일 브랜드의 상품 상자끼리 묶이도록 상품 상자들을 배치하되, 그때 상품 상자의 총 이동 횟수를 최소화 하는 것이 된다.

3.2 최적 모형화

최적 모형화를 위하여 이진 변수 x_{ij} 를 도입하자. x_{ij} 는 j번째 셀부터 시작하여 i번째 브랜드의 모든 상자들(기존 상자 및 신규 입고 상자)을 일렬로 배치할지 여부를 나타낸다. 예를 들어, 브랜드 1의 상자들이 총 3개이고 이것을 1번째 셀에 배치한다면 $x_{11} = 1$ 이 되고, 1, 2, 3번 셀이 3개의 브랜드 1 상자들로 채워지게 된다. 이때 $x_{12} = x_{13} = 0$ 이며, x_{ij} 변수 자체에는 상자의 개수 정보가 나타



<그림 2> 계산 방법 예

나지 않음에 유의하기 바란다.

목적함수를 구하기 위하여 비용 c_{ij} 를 도입하자. 비용 c_{ij} 는 i 번째 브랜드의 상자들을 j 번째 셀부터 배치하고자 할 경우 이동하여야 하는 i 번째 브랜드 상자들의 개수를 나타낸다. 예를 들어, 브랜드 1의 상자가 기존에 2개가 있었고 3, 4번 셀에 위치하고 있었다고 하자<그림 2>. 그리고 브랜드 1의 상자가 새로 1 상자가 추가 입고되었다고 하자. 이

경우 브랜드 1의 총 3개의 상자를 1번 셀부터 연속해서 배치하려면 기존 4번 셀의 상자를 2번 셀로 옮기고, 신규 입고 상자는 1번 셀에 배치하면 되므로 c_{11} 은 2(= 이동 1개 + 신규 1개)가 된다.

지금까지 정의한 변수 x_{ij} 와 비용 c_{ij} 를 이용하여 우리의 최적화 문제를 정수계획법 문제로 모형화하면 다음과 같다.

m : 브랜드의 개수

n : 셀의 개수

s_i : 브랜드 i 상품의 상자의 총 개수(기존 상자 및 신규 입고 상자) ($i=1, \dots, m$)

x_{ij} : 브랜드 i 상품을 j 번째 셀부터 s_i 개를 연속하여 배치하는지 여부(이진 변수) ($i=1, \dots, m, j=1, \dots, n-s_i+1$)

c_{ij} : 브랜드 i 상품을 j 번째 셀부터 연속하여 s_i 개를 배치할 경우 이동해야 하는 모든 상자

<표 1> 정수계획법에 의한 상용 최적화 패키지의 문제 해결 시간

셀 수	브랜드 수	상자 수	입고 상자 수	입고 브랜드 수	수행시간(초)	최적해(비용)
500	4	450	2	1	0	2
1,000	9	900	9	1	2	15
1,500	13	1,350	10	2	4	15
2,000	17	1,800	16	2	7	22
2,500	21	2,250	17	3	10	26
3,000	25	2,700	26	3	15	32
3,500	30	3,150	26	4	47	34
4,000	34	3,600	31	5	72	43
4,500	39	4,050	39	5	62	58
5,000	43	4,500	37	6	237	42
5,500	47	4,950	45	6	N/A	N/A
6,000	52	5,400	51	7	N/A	N/A
6,500	56	5,850	55	7	N/A	N/A
7,000	60	6,300	60	8	N/A	N/A

의 개수($i = 1, \dots, m, j = 1, \dots, n - s_i + 1$)

$$\text{Minimize } \sum_{i=1}^m \sum_{j=1}^{n-s_i+1} c_{ij} x_{ij} \quad (1)$$

subject to

$$\sum_{j=1}^{n-s_i+1} x_{ij} = 1 \quad \text{for } i = 1, \dots, m \quad (2)$$

$$\sum_{i=1}^m \sum_{k=\max(1, j-s_i+1)}^{\min(j, n-s_i+1)} x_{ik} \leq 1 \quad \text{for } j = 1, \dots, n \quad (3)$$

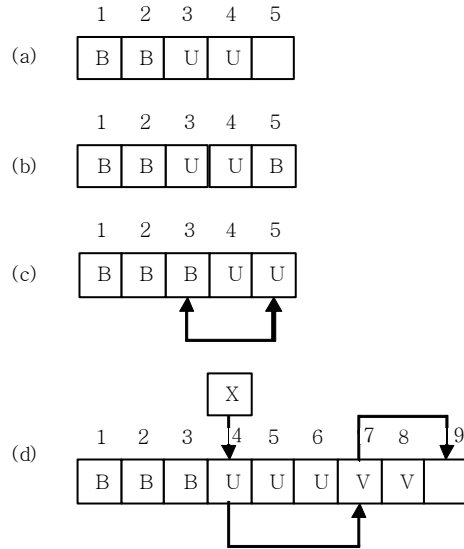
$$x_{ij} = 0 \text{ or } 1 \quad \text{for } i = 1, \dots, m, j = 1, \dots, n - s_i + 1 \quad (4)$$

위 모형에서 식 (2)는 모든 브랜드 i 는 특정 셀에 반드시 한 번만 배치되어야 함을 나타내는 제약식이고, 식 (3)은 각 셀에는 한 개 이내의 상자만 배치되어야 함을 나타내는 제약식이다.

위 모형을 실제의 한 인터넷 쇼핑몰의 창고 문제에 대해 적용하였을 때, 브랜드의 개수는 약 60개, 셀의 개수는 약 7,000개로 변수의 개수는 약 420,000개가 되었다. 그런데, LINDO, CPLEX 등 상용 최적화 패키지를 이용하여 문제 해결 시간을 검토해보았을 때, 실제 의류 인터넷 쇼핑몰이 갖고 있는 컴퓨팅 환경인 2GHz CPU를 장착한 윈도우 XP PC에서 브랜드의 개수 43개, 셀의 수 5,000개 이내일 경우 약 4분 이내에 풀지만, 그 이상이 되면 풀지 못하는 것을 확인할 수 있었다(<표 1> 참조). 따라서, 실제의 의류 인터넷 쇼핑몰 컴퓨팅 환경 하에서는 브랜드의 개수 약 60개, 셀의 개수 약 7,000개의 문제는 정수계획법에 의한 해결이 불가능하다고 이야기할 수 있다.

3.3 문제 완화에 의한 모형화

주어진 컴퓨팅 환경 하에서 기대 가능한 시간 내에 이 문제를 풀기 위해서는 이 문제를 다른 각



<그림 3> 문제 완화 예제

도에서 모형화 할 필요가 있다. 이를 위하여 상품 상자의 배치 방법을 빈 셀(상자가 배치되지 않은 셀)의 관점에서 생각해 보자.

입고할 여러 개의 상품 상자 중 상품 상자 한 개를 고르자. 이 상품 상자를 X라고 하고, X상자의 브랜드를 B라고 하자. 그리고 X를 각 빈 셀에 배치할 경우의 비용을 생각해 보자. 예를 들어, B 브랜드의 상자들이 배치된 셀들의 바로 옆의 셀에 X를 배치할 경우 같은 브랜드의 상자끼리 묶기 위한 특별한 재배치 비용은 없다.

그러나 예를 들어, B 브랜드의 상자들이 배치된 옆에 빈 셀이 없이 다른 브랜드 U의 상자들이 배치되어 있으며, 그 U상자 옆에 비로소 빈 셀이 있다고 하자. 이 경우 배치 형태는 <그림 3(a)>와 같다. 만약, 상자 X를 <그림 3(a)>의 5번 빈 셀에 배치하게 되면 배치 후 상자들의 브랜드 분포는 <그림 3(b)>와 같다. 이 경우 B 브랜드의 상자끼리 묶기 위해서는 3번 셀의 U 브랜드 상자와의 5번 셀의 X상

자를 교환하면 되며<그림 3(c)>, 이때 재배치 비용은 1이 된다. 즉, 상자 X를 5번 빈 셀에 배치하는 비용은 1이 된다.

좀 더 복잡하게, 다른 2개의 브랜드 U, V의 상자들이 있는 <그림 3(d)>의 배치 형태를 보자. 이때, 브랜드 B의 상자끼리 묶기 위하여 7번 V상자를 9번으로 옮기고, 4번 U상자를 7번으로 옮기며, 상자 X를 4번에 넣으면 된다. 따라서 이 경우 상자 X를 9번 셀에 배치하는 비용은 2가 된다. 이때, 상자 X는 궁극적으로는 4번 셀에 배치되는 점에 유의하기 바란다.

이것을 일반화하여 이야기하면, B 브랜드의 상자와 특정 빈 셀 사이에 있는 상자들의 브랜드의 개수가 d개라면 재배치 비용은 d가 된다. 이상의 논의를 바탕으로 상자 배치 문제를 모형화하면 다음과 같다.

- p : 입고되는 상자의 개수
- q : 빈 셀의 개수(단, $q \geq p$)
- y_{ij} : 상자 i 를 j 번째 빈 셀에 배치하는지 여부 (이진 변수) ($i = 1, \dots, p, j = 1, \dots, q$)
- d_{ij} : 상자 i 를 j 번째 빈 셀에 배치할 경우 재배치 비용($i = 1, \dots, p, j = 1, \dots, q$)

$$\text{Minimize } \sum_{i=1}^p \sum_{j=1}^q d_{ij} y_{ij} \quad (5)$$

subject to

$$\sum_{j=1}^q y_{ij} = 1 \quad \text{for } i = 1, \dots, p \quad (6)$$

$$\sum_{i=1}^p y_{ij} \leq 1 \quad \text{for } j = 1, \dots, q \quad (7)$$

$$y_{ij} = 0 \text{ or } 1 \quad \text{for } i = 1, \dots, p, j = 1, \dots, q \quad (8)$$

식 (6)은 신규 입고되는 상자가 어느 빈 셀에 반드시 배치되어야 함을 나타내는 제약식이고, 식

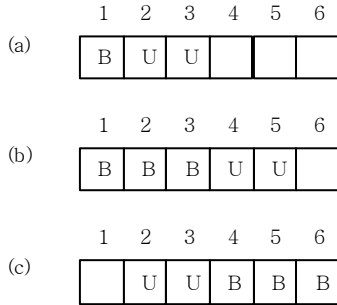
(7)은 빈 셀에는 1개 이내의 상자가 배치되어야 함을 나타내는 제약식이며, 두 제약식의 형태로부터 위 최적화 모형은 할당문제(Assignment Problem)가 됨을 알 수 있다.

할당문제는 수송문제(Transportation problem)의 특수한 형태로 식 (8)의 이진 변수 제약 없이 선형계획법(Linear programming)에 의하여 풀어도 정수해를 보장하는 것이 알려져 있고, 또한 수송문제와 할당문제를 풀기 위한 빠른 알고리즘들이 고안되어 있다. 따라서 위 최적화 모형은 식 (1)~식 (4)의 정수계획법 모형보다 훨씬 빠른 시간 내에 풀 수 있음을 알 수 있다. 다만, 이 모형이 원래 문제에 대해 최적화를 보장하는가가 문제가 되며, 이에 대해서는 다음에서 논의하기로 하겠다.

3.4 최적성 및 복잡성

복잡성 측면에서 할당모형 문제는 최적해를 보장하는 앞의 정수계획법 문제보다 풀기 쉬운 것이 자명하다. 또한, 실제 창고에서 빈 셀의 개수가 전체 셀의 20~30% 이내로 유지되는 점을 고려하면 정수계획법 문제의 셀의 수가 7,000개, 변수 개수가 420,000개에 이르는 데 반하여, 할당모형 문제의 빈 셀의 수는 약 1,400~2,100개, 변수의 개수는 신규 입고 상자의 개수가 60개 일때 약 84,000~126,000개 정도로 현저히 문제의 크기도 줄어드는 것을 알 수 있다. 실제로 의류 인터넷 쇼핑몰 컴퓨팅 환경 하에서 브랜드의 개수 약 60개, 셀의 개수 약 7,000개의 문제를 할당모형 문제로 모형화하여 풀어보았을 때 약 10초 이내에 해를 찾아주는 것을 확인할 수 있었다.

그러나, 위 할당문제 모형은 최적해를 보장하지는 않는다. 예를 들어, 입고되는 상자를 1개씩 생각하지 않고, 브랜드 별로 여러 상자를 동시에 생각해보자. 예를 들어, B 브랜드의 상자 2개를 입고하



<그림 4> 최적해 검토 예제

며, 기존에 B 브랜드의 상자가 1개 있다고 하자. 그리고, 입고 전 상자 배치가 <그림 4(a)>와 같다고 하자. 즉, B 브랜드의 상자 옆에는 빈 셀이 없고, U 브랜드의 상자를 건너서 빈 셀이 3개 있다고 하자. 위 할당모형을 이용하여 문제를 풀 경우 B 브랜드의 상자 1개당 재배치 비용이 1로서 총 재배치 비용은 2가 되며, 그때의 최종 배치는 <그림 4(b)>와 같이 된다. 그러나, 기존 1개의 B 브랜드 상자를 4번으로 옮기고, 2개의 신규 입고 상자를 5번과 6번에 배치한다고 하면, 재배치 비용은 1이 되며, 그때의 최종 배치는 <그림 4(c)>와 같이 된다.

4. 실험 및 결과

완화된 할당문제 모형의 해의 타당성을 검토하

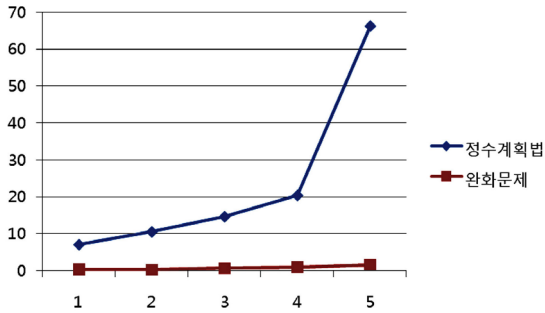
기 위하여 상용 패키지에 의해 푼 원래의 정수계획법 모형의 해와 수송문제 알고리즘에 의해 푼 완전화된 할당문제 모형의 해를 비교 실험하였다. 실제적인 비교를 위해서는 셀의 수 7,000개, 브랜드 수 60개인 문제를 풀어서 비교해야 하지만 원래의 정수계획법 모형의 경우 상용 패키지가 풀 수 없으므로, 상용 패키지가 풀 수 있는 한계 내에서 셀 수가 2,000, 2,500, 3,000, 3,500 및 4,000개, 브랜드 수가 17, 21, 25, 30 및 34개인 각각의 경우에 대해서 임의의 문제를 20개씩 생성하여 그 속도 및 최적해를 비교하였다. 실험에서 사용한 셀의 수 및 브랜드의 수는 실제 문제의 셀 수 7,000개 및 브랜드 수 60개에 비례하여 결정하였다.

<표 2>를 보면 먼저 속도 측면에서 문제 크기 별로 20개 문제를 푼 평균 시간이 정수계획법 모형의 경우 7.1, 10.55, 14.65, 20.4, 66.15초 등으로 기하급수적으로 증가하는 모습을 보이는데 반하여, 완전화된 할당문제 모형의 경우 0.35, 0.25, 0.65, 0.95, 1.6초 등으로 정수계획법 모형에 비해 현저히 작으면서, 문제의 크기가 커질 때 상대적으로 적게 증가하는 것을 볼 수가 있다<그림 5>.

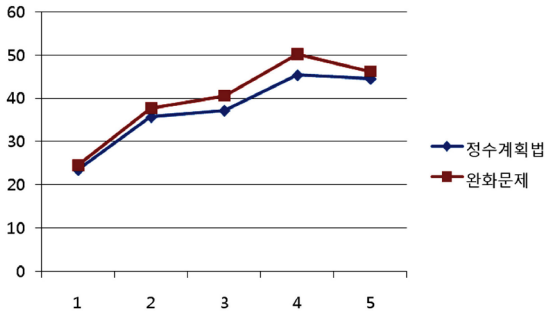
최적해의 측면에서 20개 문제의 평균 비용이 정수계획법 모형의 23.5, 35.7, 37.2, 45.4, 44.55 등의 최저값에 비하여 완전화된 할당문제 모형의 경우 24.5, 37.7, 40.6, 50.2, 46.2 등으로 각각 104.30%, 105.27%,

<표 2> 실험 결과(정수계획법과 완화문제의 실행시간 및 비용 비교)

셀 수	브랜드 수	상자 수	입고 상자 수	입고 브랜드 수	정수 계획법(초)	정수 계획법(비용 A)	완화문제(초)	완화문제(비용 B)	비율(비용 B/비용 A)(%)
2,000	17	1,800	16.4	2	7.1	23.5	0.35	24.5	104.30
2,500	21	2,250	20.75	3	10.55	35.7	0.25	37.7	105.27
3,000	25	2,700	24.65	3	14.65	37.2	0.65	40.6	109.20
3,500	30	3,150	29.8	4	20.4	45.4	0.95	50.2	110.07
4,000	34	3,600	31	5	66.15	44.55	1.6	46.2	103.40
평균	25.4	2,700	24.52	3.4	23.77	37.27	0.76	39.84	106.45



<그림 5> 문제 해결 시간 비교



<그림 6> 평균 비용 비교

109.20%, 110.07%, 103.40%로 증가한 것을 볼 수 있다<그림 6>. 이것은 평균 약 6.45%의 비율로 총 비용이 증가한 것을 나타내며, 완화된 할당문제 모형이 정수계획법 모형에 비해 현저히 빠른 시간 내에 해를 구하며, 특히 정수계획법 문제가 셀 수 7,000개짜리 실제 문제를 풀지 못하는 것을 고려하면 상당히 만족스러운 것으로 평가되었다.

정수계획법은 NP-완전(NP-complete) 문제이나 완화문제는 선형계획법 문제로서 Karmarkar 알고리즘 등에 의해 다항시간(Polynomial time) 내에 해를 도출할 수 있다. 이런 관점에서 문제의 크기가 커질수록 완화문제가 정수계획법 모형에 비해 훨씬 짧은 시간 내에 해를 도출해 낼 수 있음을 알 수 있으며, 이러한 사실은 <표 2>를 통해서도 간접적으로나마 확인할 수 있다. 또한 실험결과 셀

이 4,000개 일때 정수계획법이 완화문제에 비해 약 65초의 시간이 더 걸리지만, 움직여야 하는 상자의 개수가 약 2개 정도 늘어난 것에 대해 쇼핑물의 입고 담당자는 상자 2개를 움직이는 데 수 십초이면 충분하며, 이러한 값의 차이는 셀의 개수가 늘어남에 따라 더욱 커질 것이라는 점에서 완화문제에 의한 해를 만족스럽게 받아들였다.

5. 결론

본 연구에서는 의류 인터넷 쇼핑몰에서 주문 상품의 반출을 용이하게 하기 위한 상품의 배치를 정하는 문제를 해결하고자 하였다. 상품 반출 빈도, 상품 크기나 무게 등을 기준으로 하는 SKU의 위치를 정하는 기존 연구들과는 달리, 브랜드 별로 상품을 묶어 유지하는 것이 기준이 된다는 점에서 본 연구의 문제는 기존 문제와 다른 새로운 유형의 문제라고 할 수 있다.

본 연구의 문제는 정수계획법에 의한 최적 모형화가 가능하기는 하나, 현실적으로 문제의 크기가 너무 커서 상용 정수계획법 패키지로는 만족할 만한 시간 내에 해결하기가 곤란하였다. 따라서, 문제를 빈 셀의 관점에서 접근함으로써 문제를 완화하여 할당 문제 모형으로 변환할 수 있었다. 이렇게 얻어진 할당 문제 모형은 그 특성상 선형계획법 패키지에 의해서 빠른 시간 내에 풀릴 수 있으며, 또한 문제의 크기가 빈 셀의 수에 비례하므로 창고 내 전체 셀의 크기가 커져도 원래의 최적화 모형과 달리 문제의 크기가 심각히 커지지 않는다는 장점이 있다. 한편, 할당 문제로 완화된 모형은 최적해를 보장하지 못하는 문제가 있으나, 실험 결과 그 해가 충분히 만족스러운 것으로 판단되었다. 현재 이 방법론을 구현한 의류창고 입출고 관리 시스템이 ㈜아이에프네트워크의 패션플러스 쇼핑몰에 구

측되어 사용되고 있다.

참고문헌

- Brynzer, H. and M. I. Johansson, "Storage location assignment : Using the product structure to reduce order picking times", *Int. Journal of Production Economics*, Vol.46-47(1996), 595~603.
- Burkard, Rainer E., "Selected topics on assignment problems", *Discrete Applied Mathematics*, Vol.123(2002), 257~302.
- Caron, F., Marchet G., and Perego A., "Layout design in manual picking systems : a simulation approach", *Integrated Manufacturing Systems*, Vol.11, No.2(2000), 94~100.
- Elsayed, E. A. and O. I. Unal, "Order batching algorithms and travel-time estimation for automated storage/retrieval systems", *Int. Journal of Production Research*, Vol.27, No.7 (1989), 1097~1114.
- Goetschalckx, M. and H. D. Ratliff, "An efficient algorithm to cluster order picking items in a wide aisle", *Engineering Costs and Production Economics*, Vol.13, No.4(1988), 263~271.
- Goetschalckx, M. and R. Wei, "Bibliography on Order Picking Systems, Vol.1 : 1985~1992", 1994. Available on line at : www2.isye.gatech.edu/~mgoetsch/bibpick.ps, accessed May, 2010.
- Graves, Stephen C., Warren H. Hausman, and LeRoy B. Schwarz, "Storage-Retrieval Interleaving in Automatic Warehousing Systems", *Management Science*, Vol.23, No.9(1977), 935~945.
- Hausman, Warren H., LeRoy B. Schwarz, and Stephen C. Graves, "Optimal Storage Assignment in Automatic Warehousing Systems", *Management Science*, Vol.22, No.6(1976), 629~638.
- Heragu, S. S., L. Du, R. J. Mantel, and P. C. Schuur, "Mathematical model for warehouse Design and Product Allocation", *International Journal of Production Research*, Vol.43, No.2(2005), 327~338.
- Heskett, J. L., "Cube-per-order index-A key to warehouse stock location", *Transportation and Distribution Management*, Vol.3, No.1 (1963), 27~31.
- Jewkes, Elizabeth, Chulung Lee, and Ray Vickson, "Product Location, Allocation and Server Home based Location for an Order Picking Line with Multiple Servers", *Computers and Operations Research*, Vol.31(2004), 623~636.
- Jane, Chin Chia, "Storage location assignment in a distribution center", *International Journal of Physical Distribution and Logistics Management*, Vol.30, No.1(2000), 55~71.
- Koster, René de, Tho Le-Duc, and Kees Jan Roodbergen, "Design and Control of Warehouse Order Picking : A Literature Review", *European Journal of Operations Research*, Vol.182(2007), 481~501.
- Knuth, D. E., *The Art of Computer Programming* 3rd Edition, Volume 1 Fundamental Algorithms, Addison-Wesley, New York, NY, 1997.
- Le-Duc, T. and R.(M.) B. M. De Koster, "Travel Distance Estimation and Storage Zone Optimization in a 2-block Class-based Storage Strategy Warehouse", *International Journal of Production Research*, Vol.43, No.17(2005), 3561~3581.
- Little, T. D. C. and D. Venkatesh, "Popularity-Based Assignment of Movies to Storage Devices in a Video-on-Demand System", *MCL Technical Report*, 05-01-1994, 1994.

- Malmborg, C. J. and B. Krishnakumar, "Optimal Storage Assignment Policies for Multiaddress Warehousing Systems", *IEEE Transactions on Systems, Man, and Cybernetics*, Vol.19, No.1(1989), 197~204.
- Merkuryev, Y., A. Burinskiene, and G. Merkur-yeva, "Warehouse Order Picking Process", In : Merkuryev, Yuri, Galina Merkuryeva, Mi-quel Àngel Piera, and Antoni Guasch(Eds.) *Simulation-Based Case Studies in Logistics : Education and Applied Research*, Springer, London, 2009, 147~165.
- Muralidharan, B., R. J. Linn, and R. Pandit, "Shuffling heuristics for the storage location assignment in an AS/RS", *Int. Journal of Production Research*, Vol.33, No.6(1995), 1661~1672.
- Petersen, C. G., "An evaluation of order picking routing policies", *International Journal of Operations and Production Management*, Vol.17, No.11(1997), 1098~1111.
- Petersen, C. G. and Schmenner R. W., An evaluation of routing and volume-based storage policies in an order picking operation, *Decision Sciences*, Vol.30, No.2(1999), 481~501.
- Roodbergen, K. J., Layout and routing methods for warehouses. Erasmus Research Institute of Management, Erasmus University, Rotterdam, 2001.
- White, J. A. and H. D. Kinney, "Storage and Warehousing", In *Handbook of Industrial Engineering*, Chapter 10.4, G. Salvendy(ed.), John Wiley and Sons, New York, NY, 1982.

Abstract

An Efficient Heuristic for Storage Location Assignment and Reallocation for Products of Different Brands at Internet Shopping Malls for Clothing

Yong Uk Song* · Byung-Hyuk Ahn**

An Internet shopping mall for clothing operates a warehouse for packing and shipping products to fulfill its orders. All the products in the warehouse are put into the boxes of same brands and the boxes are stored in a row on shelves equipped in the warehouse. To make picking and managing easy, boxes of the same brands are located side by side on the shelves. When new products arrive to the warehouse for storage, the products of a brand are put into boxes and those boxes are located adjacent to the boxes of the same brand. If there is not enough space for the new coming boxes, however, some boxes of other brands should be moved away and then the new coming boxes are located adjacent in the resultant vacant spaces.

We want to minimize the movement of the existing boxes of other brands to another places on the shelves during the warehousing of new coming boxes, while all the boxes of the same brand are kept side by side on the shelves. Firstly, we define the adjacency of boxes by looking the shelves as an one dimensional series of spaces to store boxes, i.e. cells, tagging the series of cells by a series of numbers starting from one, and considering any two boxes stored in the cells to be adjacent to each other if their cell numbers are continuous from one number to the other number. After that, we tried to formulate the problem into an integer programming model to obtain an optimal solution. An integer programming formulation and Branch-and-Bound technique for this problem may not be tractable because it would take too long time to solve the problem considering the number of the cells or boxes in the warehouse and the computing power of the Internet shopping mall.

As an alternative approach, we designed a fast heuristic method for this reallocation problem by focusing on just the unused spaces-empty cells-on the shelves, which results in an assignment problem model. In this approach, the new coming boxes are assigned to each empty cells and then those boxes are reorganized so that the boxes of a brand are adjacent to each other. The objective of this new approach is to minimize the movement of the boxes during the reorganization process

* School of Business Administration, Yonsei University Wonju Campus

** Dept. of MIS, Gyeongsang National University

while keeping the boxes of a brand adjacent to each other. The approach, however, does not ensure the optimality of the solution in terms of the original problem, that is, the problem to minimize the movement of existing boxes while keeping boxes of the same brands adjacent to each other. Even though this heuristic method may produce a suboptimal solution, we could obtain a satisfactory solution within a satisfactory time, which are acceptable by real world experts. In order to justify the quality of the solution by the heuristic approach, we generate 100 problems randomly, in which the number of cells spans from 2,000 to 4,000, solve the problems by both of our heuristic approach and the original integer programming approach using a commercial optimization software package, and then compare the heuristic solutions with their corresponding optimal solutions in terms of solution time and the number of movement of boxes. We also implement our heuristic approach into a storage location assignment system for the Internet shopping mall.

Key Word : Internet Shopping Mall, Order Picking, Reallocation, Storage Location Assignment, Warehouse

저자 소개



송용욱

현재 연세대학교 원주캠퍼스 경영학부 부교수로 재직 중이다. 서울대학교 국제경제학과를 졸업하고, 한국과학기술원(KAIST) 경영학과 및 산업경영학과에서 석사 및 박사학위를 취득하였다. Management Science, Annals of Operations Research, Expert Systems with Applications 등에 논문을 게재한 바 있다. 주요 관심분야는 전자상거래, 정보시스템 개발, 경영분야 문제의 전문가시스템 응용, 전문가시스템 및 수리계획법과 전자상거래의 통합 등이다.



안병혁

경상대학교 경영정보학과 조교수로서 현재 미국 Kent State University에 교환교수로 파견 연구 중이다. 한국외국어대학 러시아어과를 졸업하고 서울대학교 대학원과 미국 Michigan State University에서 경영학 석사를, 미국 Kent State University에서 경영학 박사를 취득하였다. Decision Sciences, IIE Transactions 등에 논문을 게재하였으며 관심 분야로는 정보시스템 개발, 신경망 응용, 전역 최적화 등이다.