Journal of the Korean Institute of Industrial Engineers
Vol. 30, No. 4, pp. 301-305, December 2004.

판재류의 주문별 출고순서 결정

명영수†

단국대학교 경상학부

Optimal Sequencing of Orders for Picking in a Plate Warehouse

Young-Soo Myung

Dept. of Business Administration, Dankook University, Cheonan, 330-714

Abstract: In a warehouse, plates listed in the different orders are placed mixed. Nevertheless, when plates are going out, a group of plates listed in the same order should be sequentially loaded on a transporter. Since the access to the plates is only from the top of the stack, it happens often that we have to move some plates on a temporary place in order to access a plate below them. This paper deals with a problem of sequencing orders for transportation to minimize the number of plates temporarily moved.

Keywords: optimal sequencing, plate warehouse, combinatorial optimization

1. 서 론

본 논문은 창고에서 판재를 출고할 때 발생하는 의사결정을 다루고 있다. 이 창고와 관련된 의사결정 환경은 다음과 같다 판재는 물리적인 특성상 창고에 보관할 때 상하로 쌓아 놓게 되는데, 이렇게 쌓아 놓는 것을 적치한다고 표현하기로 한다. 창고의 바닥은 주차장의 주차구역처럼 일정한 면적으로 구분되어 있는데, 이렇게 구분된 구역을 베드라고 부른다. 한 베드에 적치할 수 있는 판재의 개수는 제한되어 있고, 동일한 베드에는 출고일정이 같은 판재만 적치한다 동일한 주문에 속한 판재의 출고일정은 같다 판재는 창고에 한 장씩 입고되고, 입고된 판재는 적치할 여유가 있는 베드의 맨 위쪽에 놓이게 된다. 한 주문에 속한 판재의 개수가 베드의 허용개수보다 적은 경우에는 베드의 효율적 활용을 위해서 서로 다른 주문에 속한 판재를 함께 적치하게 된다. 이 때, 동일한 주문에 속한 판재들이 차례대로 창고에 입고되지 못하므로, 베드에는 동일한 주문에 속한 판재 사이에 이들과는 다른 주문에 속한 판재가 위치하는 경우가 흔하게 발생한다.

창고에서는 출고지시에 따라 판재를 베드에서 운송차량으로 옮겨 싣는 출고작업을 하게 된다. 출고일정이 같은 판재의 운송순서에는 제약이 없으나, 동일한 주문에 함께 포함된 판재들은 출고 때에 다른 주문에 속한 판재와 섞여서 운송되어서는 안 된다. 판재의 특성상 출고를 위한 작업은 가장 위쪽에 위치한 판재부터 이루어져야 하기 때문에, 아래쪽에 위치한 판재를 먼저 출고하기 위해서는 출고하려는 판재의 위에 놓여 있는 판재들을 일단 다른 장소로 임시로 적치하여야 된다. 따라서 출고일정이 동일한 판재를 출고할 때 어떤 주문의 판재들을 먼저 출고하느냐에 따라 임시적치의 수가 달라져서, 창고에서는 임시적치의 수를 최소화하는 주문별 출고순서를 정하는 의사결정이 필요하게 된다. 본 논문에서 다룰 주문별 출고순서 결정문제의 목적은 이러한 최적 의사결정을 찾는 것이다. 임시로 적치하는 판재들은 겹쳐서 쌓아 놓지 않기 때문에 임시로 적치된 판재를 출고할 때는 추가로 임시적치가 발생하지는 않는다. 따라서 임시적치는 판재별로 한 번을 초과하지는 않는다.

주문별 출고순서 결정문제에 대한 이해를 돕기 위해서 베드가 한 개인 단순한 경우의 예를 이용하여 설명하기로 하자. <Figure 1>에서와 같이 한 개의 베드에 판재가 적치되어 있다고 가정하자. 그림에서 사각형은 판재를 의미하며, 사각형 안의 두 숫자 중 앞에 있는 숫자는 판재가 속한 주문의 번호를, 뒤에 있는 숫자는 판재의 번호를 표현한다. 판재의 번호가 클수록 아래쪽에 위치하므로 번호가 큰 판재가 먼저 창고에 입

이 연구는 2004학년도 단국대학교 대학연구비의 지원으로 연구되었음.
† 연락저자 : 명영수 교수, 330-714 충남 천안시 안서동 산 29 단국대학교 경상학부, Fax : 041-550-3390, E-mail : myung@dankook.ac.kr
2003년 8월 접수; 2004년 3월 수정본 접수; 2004년 6월 게재 확정.

고되었을 것이다. 만약에 주문별 출고순서가 주문번호2-3-1의 순서라고 가정하여 보자. 우선 2번 주문에 속한 판재를 출고하기 위해서 1, 2, 3번의 판재를 실은 후에 8번 판재를 싣기 위해서 8번 판재의 위쪽에 위치한 3번 주문에 속한 4, 5, 6번 판재와 1번 주문에 속한 7번 판재를 임시로 적치하여야 한다. 그 다음에 3번 주문에 속한 판재를 출고할 때는 임시로 적치하고 있는 4, 5, 6번 판재를 싣고 베드에 유일하게 남아 있는 1번 판재를 싣게 된다. 마지막으로 임시로 적치되어 있는 1번 주문에 속한 3번 판재를 출고하게 된다. 따라서 이 경우에 임시적치는 총 4번이 필요하다. 또 다른 예로서, 만약에 출고가 주문 1-2-3의 순서로 이루지면 필요한 임시적치의 수는 6번이 된다. 왜냐하면 판재 1, 2, 3, 4, 5, 6번이 임시로 적치되기 때문이다.

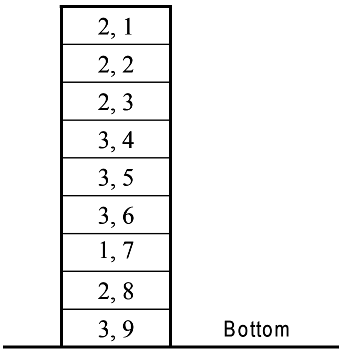

Figure 1. Plates on a bed.

동일한 주문에 속한 판재를 두 개 이상의 서로 다른 베드에 나누어 적치하는 것을 분적이라고 부른다. 하나의 주문에 속한 판재의 개수가 하나의 베드에 적치할 수 있는 판재의 개수보다 상대적으로 적을 때에는 같은 주문에 속한 판재는 같은 베드에 적치하는 정책을 사용하게 된다. 즉, 분적을 허용하지 않는다. 이러한 정책은 같은 주문에 속한 판재들이 여러 베드에 산재하여 위치하면 크레인의 이동 등 출고에 필요한 작업 시간이 길어지기 때문이다. 한편으로 하나의 주문에 포함되는 판재의 개수가 베드에 적치할 수 있는 판재의 수보다 상대적으로 많거나, 그렇지 않더라도 베드의 활용을 극대화하기 위해서는 분적을 허용하는 전략을 사용할 수도 있다.

출고순서 결정문제는 다음과 같이 수학적인 개념을 이용하여 표현할 수 있다. 출고순서를 정해야 할 주문들의 번호로 이루어진 집합을 $N=\{1, 2, \cdots, n\}$이라 하자. 그러면 출고순서는 N의 선형순서(linear ordering 또는 permutation)에 해당한다. 선형순서는 N에서 N으로의 일대일 함수 $\sigma: N \to N$로 표현할 수 있다. 임의의 선형순서 σ에 맞춰 출고할 때 발생하는 임시적치의 수를 $c(\sigma)$로 표시하면, 출고순서 결정문제는 $c(\sigma)$를 최소로 하는 선형순서 σ를 결정하는 문제가 된다. 따라서 출고순서 결정문제의 난이도는 $c(\sigma)$를 쉽게 표현할 수 있는지, 또한 $n!$에 해당하는 선형순서 중에서 $c(\sigma)$를 최소로 하는 선형순서를 쉽게 찾을 수 있는지에 달려 있다.

출고순서 결정문제의 대표적인 응용 예는 제철소의 후판창고에서 찾을 수 있다. 제철소의 판재는 중량이 무거워서 크레인으로 움직이게 되므로 가능한 임시적치를 최소화하는 것이 창고의 중요한 관심 사항이다. 조선소에서는 판재의 배달장소, 판재를 가공할 기계, 판재가 사용될 선박의 블록 등을 고려하여 하나의 주문에 포함시킬 판재들을 결정한다. 조선소에서는 배달된 판재의 선별작업을 용이하게 하기 위하여 서로 다른 주문에 속한 판재가 섞이지 않고 배달되기를 원한다. 출고는 선박을 이용하므로 동일한 선박으로 운송되는 주문들의 출고순서에는 제약이 없고, 주문별 출고순서의 결정은 후판창고에서 하게 된다. 따라서 창고에서는 임시적치가 최소가 되도록 출고순서를 정하는 것이 중요한 과제이다. 후판창고의 의사결정모형이 좀더 현실적이기 위해서는 입고되는 후판이 적치될 베드를 선택하는 결정도 포함하여야 하지만, 전체 의사결정을 동시에 최적화하는 모형은 너무 복잡해진다. 따라서 의사결정을 베드의 선택과 출고순서 결정의 두 단계로 나누어서 접근하는 것이 현실적이다.

이처럼 출고순서 결정문제는 판재류를 취급하는 기업의 창고에서 흔히 발생하는 중요한 의사결정문제이나, 이에 대한 연구는 이전의 연구에서 찾아볼 수가 없다. 본 논문에서는 분적을 허용하지 않는 경우와 허용하는 경우에 대해서 판재류의 주문별 출고순서 결정문제의 특성을 분석한다. 분적이 허용되지 않는 경우에 대해서는 최단경로문제(Shortest Path Problem)의 해법을 이용하여 다항시간 내에 해결할 수 있음을 보이고 분적을 허용하는 경우에는 이 문제가 NP-hard임을 증명하기로 한다.

2. 분적이 허용되지 않는 경우

분적이 허용되지 않는 경우와 허용되는 경우의 결정적인 차이는 전자의 경우에는 각 베드별로 독립적으로 최적의 출고순서를 결정함으로써 전체의 최적출고순서를 결정할 수 있다는 점이다. 따라서 분적을 허용하지 않는 경우의 출고순서 결정문제는 판재가 한 개의 베드에만 적치되어 있는 특수한 경우로 단순화시킬 수 있다. 이 장에서는 이러한 특수한 경우의 출고순서 결정문제가 동등한 성질을 갖는 최단경로문제로 표현할 수 있음을 보이기로 한다.

앞에서 소개한 <Figure 1>의 예제를 활용하여 베드가 한 개인 출고순서 결정문제의 특성을 설명하기로 하자. 앞으로 n개의 주문에 속한 m개의 판재가 베드에 적치되어 있는 경우에 판재의 번호와 주문의 번호는 다음과 같이 부여되었다고 가정한다. 판재의 경우는 가장 위쪽의 판재가 1번이고 바닥에 위치한 판재일수록 큰 번호를 갖는다. 주문의 경우는 주문에 속한 판재 중에서 번호가 가장 큰 — 가장 먼저 적치되어서 바닥에 가장 가깝게 있는 — 판재의 번호가 클수록 큰 주문의 번호가 부여되었다. <Figure 1>의 번호는 이러한 가정에 부합된다. 이

처럼 주문의 번호를 부여한 이유는 특정 출고순서에 의하여 발생하는 임시적치의 수는 각각의 주문에 속한 판재 중에서 가장 번호가 큰 판재의 위치에 의해서 결정되기 때문이다.

이제 출고순서와 출고순서에 따라서 발생하는 임시적치의 수와의 관계를 분석해 보자. 특정 출고순서에서 임의의 주문이 자신의 주문번호보다 큰 번호를 갖는 주문보다 늦게 출고되면, 해당주문에 포함된 모든 판재는 임시로 적치하여야 된다. 예로서 2번 주문이 3번 주문보다 늦게 출고되는 경우에는 2번 주문에 속한 모든 판재는 임시적치가 필요하다. 이러한 관찰을 일반화하여 표현하기 위해서 다음과 같은 용어를 정의하자. 특정 출고순서에서 임의의 주문이 자신의 번호보다 큰 번호를 갖는 주문보다 뒤에 위치하면, 해당주문은 주어진 출고순서에서 역순으로 배정되었다고 표현하기로 한다. 즉, 출고순서 2-3-1에서 1번 주문은 역순으로 배정된 것이며, 출고순서 3-1-2에서는 1, 2번 주문 모두 역순으로 배정된 것이다.

(관찰 1) 특정 출고순서에서 역순으로 배정된 주문에 속한 모든 판재는 임시적치가 필요하다.

임의의 출고순서에 대해서 역순의 주문을 제외한 나머지 주문들의 출고순서를 해당 출고순서에 대한 순행순서라고 부르기로 하자. <Figure 1>의 예에서 2-1-3과 2-3-1의 출고순서에서 주문 1은 두 경우에 모두 역순이다. 두 가지 출고순서에서 역순인 1을 제외하면 남은 순서는 2-3으로 동일하게 된다. 따라서 출고순서 2-1-3과 2-3-1에 대응되는 순행순서는 2-3으로 동일하다. (관찰 1)에서 나타난 바와 같이 두 가지의 출고순서 중 어떤 순서로 출고가 이루어져도 역순인 주문1에 속한 판재는 모두 임시적치가 필요하다. 또한 두 출고순서에 의해서 발생하는 임시적치의 수는 4번으로 동일한데, 이러한 사실은 다음과 같이 일반화할 수 있다.

(정리 1) 임의의 두 출고순서의 순행순서가 동일하면, 두 출고순서에 의해서 발생하는 임시적치의 수는 동일하다.

(증명) 두 출고순서의 순행순서가 동일하면 두 출고순서에 포함된 역순의 주문들은 위치는 다르나 번호는 동일하다. (관찰 1)에서 나타난 바와 같이 역순의 주문에 포함된 판재들은 위치에 상관없이 모두 임시로 적치되고, 자신의 출고순서에서는 임시로 적치된 위치에서 운송되므로 별도의 임시적치를 발생시키지 않는다. 또한, 두 출고순서에 따라 출고가 이루어지는 과정을 비교하여 보자. 순행순서에 속한 임의의 주문에 대한 출고가 시작되는 시점에 베드에 남아 있는 판재는, 두 출고순서에서 모두 동일하다. 따라서 두 출고순서에 의해서 발생하는 임시적치의 수는 동일하다. □

순행순서는 성격상 항상 오름차순의 형태가 되고, 번호가 가장 큰 주문은 역순이 될 수 없으므로 모든 순행순서에 항상 포함되며 맨 마지막에 위치하게 된다. 즉, 1, 2, 3번 3개의 주문에 대한 모든 가능한 출고순서 6가지에 대해서 생성될 수 있는 순행순서는 1-2-3, 1-3, 2-3, 3의 4가지뿐이다. 3개의 주문에 대한 출고순서와 이에 대응되는 순행순서는 <Table 1>에 나타난 것과 같다.

Table 4. Sequences of three orders and the corresponding forward sub-sequences

Sequence	forward subsequence
1-2-3	1-2-3
1-3-2	1-3
2-1-3 2-3-1	2-3
3-1-2 3-2-1	3

이처럼 출고순서와 순행순서는 다대일(multiple to one) 대응이므로 특정 순행순서에 복수의 출고순서가 대응될 수 있다. 그러나 특정 순행순서에 대응되는 출고순서는 쉽게 파악할 수 있고, (정리 1)에서 알 수 있듯이 동일한 순행순서에 대응되는 출고순서들이 발생시키는 임시적치의 수는 동일하다. 예로서 순행순서 1-3에 대응되는 출고순서를 어떻게 파악할 수 있는지 생각해 보자. 남아 있는 주문 2가 역순으로 들어가야 하고, 현재의 1,3을 역순으로 만들지 않기 위해서는 1-3-2가 유일하다. 즉, 2가 들어갈 수 있는 남은 두 가지 경우 중 1-2-3은 2가 역순이 안 되고, 2-1-3은 1이 역순이므로 이 경우에 대응되는 순행순서는 1-3이 아니라 2-3이 되기 때문이다.

주문의 수가 늘어날수록 순행순서의 경우의 수는 출고순서의 경우의 수에 비하여 현격히 줄어들게 된다. 하지만 순행순서의 경우의 수는 여전히 주문의 수에 대해서 기하급수적으로 늘어나므로, 모든 가능한 경우를 비교하여 임시적치를 최소로 하는 출고순서를 결정하는 방법은 실용적이지 못하다. 따라서 좀 더 효율적인 방법이 가능하다면 유용하게 쓰일 수 있을 것이다. 우리는 베드가 한 개인 경우의 출고순서 결정문제가 최단경로문제로 표현될 수 있음을 보임으로써 분적을 허용하지 않는 경우의 문제가 기존에 개발된 최단경로문제의 해법을 이용하여 쉽게 해결될 수 있음을 보이기로 한다. 특히 이 경우의 최단경로문제는 사이클(cycle)이 존재하지 않는 유방향 그래프(directed graph)에 정의되기 때문에 일반적인 그래프에서의 최단경로문제보다 더 효율적인 해법이 가능하다. 이러한 해법에 대해서는 (Ahuja *et al.*, 1993)을 참조하기 바란다.

우선 하나의 베드에 적치되어 있는 n개의 주문에 속한 판재의 자료에 대해서 다음과 같은 보조 그래프를 구성한다. 노드는 가상노드 (노드 0)와 각 주문별로 1에서 n까지 노드가 하나씩 대응되며 노드의 번호는 주문번호와 같다. 따라서 노드의 수는 n+1이 된다. 두 노드 i와 j에 대해서 $i < j$이면 아크 (i, j)가

존재한다. <Figure 1>처럼 3개의 주문에 속한 판재가 적치되어 있는 자료에 대응되는 보조 그래프는 <Figure 2>와 같다.

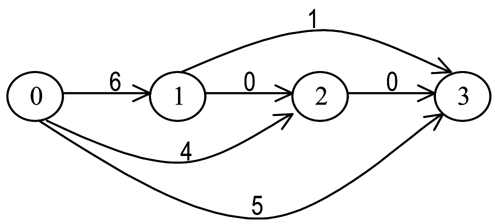

Figure 2. Auxiliarly graph with respect to (Figure 1).

이렇게 구성된 보조 그래프에서는 그래프에 포함된 노드 0에서 노드 n까지의 경로(path)와 순행순서가 1대1로 대응된다. <Figure 2>의 경우 노드 0에서 노드 3까지의 임의의 경로에서 노드 0를 뺀 나머지 노드의 순서와 <Figure 1>의 순행순서가 1대1로 대응된다. 즉, <Figure 2>의 노드 0에서 노드 3까지의 경로를 나열하면 0-1-2-3, 0-1-3, 0-2-3, 0-3인데, 여기서 노드 0을 뺀 나머지 순서와 <Figure 1>의 순행순서 4가지와 일치하게 된다. 이제 아크의 거리를 적절한 값으로 정하면, 노드 0에서 노드 n까지의 임의의 경로의 길이가, 이 경로와 대응되는 순행순서에 의해서 발생하는 임시적치의 수와 같아질 수 있음을 보인다. 만약에 이러한 시도가 가능하다면, 임시적치를 최소화하는 출고순서는, 노드 0에서 노드 3까지의 최단경로를 구함으로써 가능하게 된다. 이를 위해서 아크(i, j)의 거리 d_{ij}를 다음과 같이 정의한다.

(정의) 아크 (i, j)의 거리 d_{ij}: 주문 i에 속한 가장 번호가 큰 판재의 번호보다는 크고(i=0이면 0보다는 크고), 주문 j에 속한 가장 번호가 큰 판재의 번호보다는 작은 번호를 갖고 있는 판재 중에서, 주문 j에 속하지 않은 판재의 수.

정의에 의해서 아크 (i, j)의 거리는 주문 i에 속한 판재의 출고를 완료한 시점부터 주문 j에 속한 판재의 출고를 완료할 때까지 발생하는 임시적치의 수가 된다. <Figure 1>의 예를 이용하여 만들어진 보조 그래프에 속한 아크의 거리가 <Figure 2>에 나타나 있다. 그림에서 아크 (1,3)에 대한 거리는 정의상 7보다 크고 9보다 작은 번호의 판재 중에서 주문 3에 속하지 않은 판재의 개수이므로 1이 된다. 그리고 이 값은 1번 주문에 속한 판재의 출고가 완료된 후부터 3번 주문에 속한 판재의 출고가 완료될 때까지 발생하는 임시적치의 수를 의미하는데, 8번 판재만 해당되므로 1임을 알 수 있다. 이제까지 설명한 내용을 공식적으로 증명하기로 한다.

(정리 2) 주문이 n개인 출고순서 결정문제에서 임시적치를 최소로 하는 출고순서는, 보조 그래프의 노드 0에서 노드 n까지의 최단경로에서 0을 뺀 나머지 노드의 순서를 순행순서로 갖는다. 또한 최소 임시적치의 수는 최단경로의 길이와 같다.

(증명) 앞서 언급한 대로 순행순서는 성격상 항상 오름차순의 형태이며, 주문 n은 맨 마지막에 항상 포함된다. 보조 그래프에서 $i < j$인 노드 i와 j에 대해서만 아크 (i, j)가 존재하므로 노드 0에서 노드 n까지의 경로에 속한 노드의 번호는 오름차순의 형태를 취하고, 마지막에 노드 n이 위치한다. 따라서 경로에 속한 노드에서 0을 뺀 나머지 노드의 순서와 순행순서는 1대1로 대응된다. 이제 임의의 출고순서에 의해 발생하는 임시적치의 수는, 이 출고순서의 순행순서에 대응되는 노드 0에서 노드 n까지의 경로의 길이와 같음을 보이기로 한다. 역순의 주문에 포함된 판재들은 자신에 대한 출고가 이루어질 때, 임시 위치에 있으며 추가적인 임시적치를 발생시키지 않는다. 또한 어떤 출고순서에 따라서 출고가 이루어져도 주문 n에 대한 출고가 끝나는 순간에는 베드에는 판재가 더 이상 존재하지 않으므로 임시적치도 더 이상 발생하지 않는다. 따라서 총 임시적치의 수는 순행순서에 나타난 주문에 대한 출고가 이루어질 때 발생된 임시적치의 수만 합하면 되는데, 이는 대응되는 경로에 속한 아크의 거리의 합과 같다. □

<Figure 2>에서 최단경로는 0-2-3이므로 최선의 출고순서는 순행순서가 2-3인 출고순서가 된다. 따라서 출고순서 2-1-3 또는 2-3-1이 최적 출고순서가 된다. 두 순서에 의해서 발생하는 임시적치의 수는 동일하고 최단경로의 길이인 4가 된다.

3. 분적이 허용되는 경우

하나의 주문에 포함되는 판재의 개수가 베드에 적치할 수 있는 판재의 수보다 상대적으로 많거나, 그렇지 않더라도 베드의 활용을 극대화하기 위해서 같은 주문에 포함된 판재들을 서로 다른 베드에 나누어 적치할 수 있다. 즉, 분적을 허용할 수 있다. 이 장에서는 분적이 허용되는 경우에 출고순서 결정문제에 대해서 분석해 보기로 한다. 분적을 허용하지 않는 경우에는 각각의 베드에 속한 주문의 출고순서를 최적화하면 전체 베드에 속한 주문의 출고순서를 최적화할 수 있었다. 그러나 분적이 허용되는 경우에는 베드별로 출고순서를 최적화하는 방법으로 전체 베드의 최적순서를 결정하는 것은 불가능하기 때문에, 베드가 하나인 경우의 출고순서 결정문제에 대한 해법으로는 해결이 안 된다.

출고순서의 결정문제는 분적이 허용되지 않는 경우에는 최단경로문제의 해법을 이용하여 다항시간 안에 풀 수 있었다. 분적이 허용되는 경우에도 다항시간 안에 풀 수 있는 해법이 가능할지를 우선 살펴보기로 한다. 이를 위해서 분적이 허용되는 경우의 출고순서 결정문제의 복잡성을 규명하기로 하자. 결론을 먼저 이야기하면 이 문제는 NP-hard이다. 이러한 사실은 NP-hard 문제인 선형순서결정 문제(Linear Ordering Problem) (Gröchel *et al*., 1986; Garey and Johnson, 1976)가 출고순서 결정문제의 특수한 경우임을 보임으로써 규명할 수 있다. 선형순서 결정문제는 비용을 최소화하는 집합 $N=\{1, 2, \cdots, n\}$의 선형순서

$\sigma: N \rightarrow N$를 결정하는 문제이다. N에 속한 임의의 두 원소 i와 j에 대해서 i가 j를 앞서도록 순서를 결정하면 c_{ij}의 비용이, 반대로 j가 i를 앞서도록 결정하면 c_{ji}의 비용이 발생한다. 즉, 선형순서 결정문제는 목적함수 $c(\sigma) = \sum_{\sigma^{-1}(i) < \sigma^{-1}(j)} c_{ij}$를 최소화하는 선형순서 σ를 결정하는 문제이다.

(정리 3) 분적이 허용되는 경우의 출고순서 결정문제는 NP-hard이다.

(증명) 선형순서 결정문제가 분적이 허용되는 출고순서 결정문제의 특수한 경우로 다항시간 안에 변형될 수 있음을 보임으로써 증명하기로 한다. 선형순서 결정문제의 집합 N에 속한 원소별로 주문을 하나씩 대응시킨다. 그리고 집합 N에 속한 임의의 두 원소 i와 j의 쌍(pair)에 대해서 베드를 하나씩 대응시킨다. 따라서 총 $n(n-1)/2$개의 베드가 존재하는데, 각 베드에는 다음과 같이 판재가 적치되어 있다고 가정한다. 두 원소 i와 j의 쌍(pair)에 대응된 베드에는 가장 위쪽으로부터 주문 j에 속한 판재 c_{ij}개가 위치하고, 바로 아래쪽부터 주문 i에 속한 c_{ji}개의 판재가 위치하며, 맨 마지막에 주문 j에 속한 판재가 한 개 위치한다. 따라서 두 원소 i와 j의 쌍(pair)에 대응된 베드에서는 주문 i의 출고순서가 주문 j의 출고순서보다 빠르면 c_{ij}번의 임시적치가 일어나고, 주문 j의 출고순서가 주문 i의 출고순서보다 빠르면 c_{ji}번의 임시적치가 일어나게 된다. 그러면 임의의 출고순서에 의해서 전체 베드에서 발생하는 임시적치의 수는 이 출고순서와 동일한 선형순서에 의해서 발생하는 총 비용과 같다. □

(정리 3)은 분적이 허용되는 경우의 출고순서 결정문제를 다항시간 안에 풀기는 어렵다는 것을 암시한다. 게다가 선형순서 결정문제와는 달리 임의의 출고순서를 선택할 때 발생하는 임시적치의 수를 명시적인 함수의 형태로 표시하는 것조차도 어려워 보인다. 선형순서 결정문제에서는 목적함수를 두 원소의 선후관계에 대한 값의 합으로 표시할 수 있었으나, 임시적치를 표시하는 함수는 이러한 방법으로는 표시가 불가능하기 때문이다. 따라서 분적이 허용되는 경우의 출고순서 결정문제에 대해서는 구체적인 수리계획법의 모형으로 표현하기는 어려워 보인다. 따라서 기존의 조합최적화의 기법을 이용하여 최적해를 구하기는 쉽지 않을 것으로 판단된다.

5. 결론 및 향후 연구에 대한 논의

본 논문에서는 분적을 허용하지 않는 경우와 허용하는 경우 각각에 대해서, 판재류의 주문별 출고순서 결정문제에 대해서 분석하였다. 분적이 허용되지 않는 경우에 대해서는 최단경로 문제의 해법을 이용하여 다항시간 내에 해결할 수 있음을 보였고, 분적을 허용하는 경우에는 이 문제가 NP-hard임을 증명하였다. 출고순서 결정문제는 판재류를 취급하는 기업의 창고에서 흔히 발생하는 문제이므로 본 연구의 결과에 대한 활발한 활용이 기대된다.

본 논문에서는 판재가 이미 베드에 적치되어 있다는 가정에서 주문별 출고순서만을 결정하는 문제를 다루고 있으나, 현실적으로는 판재의 입고에 대한 정보가 주어져 있을 때 임시적치가 필요한 판재의 수를 최소로 하기 위해서 판재를 어떤 베드에 적치할 것인지도 동시에 결정하여야 한다. 따라서 앞으로 판재의 적치와 주문의 출고순서를 동시에 결정하는 문제에 대한 연구가 필요하다. 이러한 연구에도 본 논문의 출고순서 결정문제에 대한 연구결과가 도움이 될 것으로 생각된다. 또한, 현실에서는 입고되는 판재에 대한 정보는 있으나 판재의 입고순서는 미리 알기 어려운 경우도 많다. 이러한 경우를 대비해서는 온라인 형태의 해법 개발이 필요하다고 생각된다. 분적이 허용되는 경우의 출고순서문제에 대해서는 메타휴리스틱과 같은 방법의 적용도 고려해 볼만하다.

참고문헌

Ahuja, R. K., Magnanti, T. L. and Orlin, J. B. (1993), *Network Flows*, Prentice-Hall, New Jersey, U.S.A.

Garey, M. R. and Johnson, D. S. (1976), *Computers and Intractability*, W. H. Freeman and Company, San Francisco, U.S.A.

Gröchel, M., Jünger, M., and Reinelt, G. (1985), On the acyclic subgraph polytope, *Mathematical Programming* **33**, 28-42.