한국경영과학회지
제42권 제1호
2017년 2월
https://doi.org/10.7737/JKORMS.2017.42.1.043

주문집약문제에 대한 $\frac{1}{3}$-근사해법*

명 영 수†
단국대학교 경영학부

A $\frac{1}{3}$-Approximation Algorithm for the Order Consolidation Problem

Young-Soo Myung
Department of Business Administration, Dankook University

■ Abstract ■

We consider the order consolidation problem that arises in production systems where customer orders are processed in batches. The problem involves maximizing the number of batches while satisfying the following conditions: (i) the total quantity processed in each batch must be above a prescribed level; (ii) the quantity in an order can be split and processed in more than one batches; (iii) each batch can include up to two different orders but can do so only when the two orders are compatible pair. This problem is known to be NP-hard and max-SNP hard. In this study, we develop an approximation algorithm with factor 1/3.

Keywords : Order Consolidation, Approximation Algorithm

논문접수일 : 2016년 11월 25일 논문게재확정일 : 2016년 12월 21일
논문수정일(1차 : 2016년 12월 14일, 2차 : 2016년 12월 17일)
* 이 연구는 2015년도 단국대학교 대학연구비의 지원으로 연구되었음.
† 교신저자 myung@dankook.ac.kr

1. 서 론

Hwang and Chang[4]은 일괄처리 생산을 위해서 요구량과 생산조건이 다른 주문들을 적절하게 통합하는 주문집약문제(order consolidation problem : OCP)를 제시하였다. 일괄처리 생산방식(batch processing)이란 시스템에서 주문 받은 물량을 일정한 규모의 배치(batch) 단위로 묶어서 생산하는 것이다. Hwang and Chang[4]이 제시한 주문집약문제는 다음과 같이 정의된다. 하나의 주문에 포함된 주문량을 두 개 이상의 배치에 나누어서 생산이 가능하고, 서로 다른 주문은 두 개까지는 하나의 배치에 포함시킬 수 있으나 요구사항이 유사한 주문끼리만 동일한 배치에 포함될 수 있다. 또한 배치가 구성되기 위해서는 배치에 포함되는 주문량의 합이 일정규모 이상이 되어야 한다. 주문집약문제는 주어진 주문들에 대해 조건을 만족하는 최대 개수의 배치를 구성하는 문제이다.

주문집약문제는 철강, 화학 등의 소재를 생산하는 시스템에서 흔히 발생하는 문제로 알려져 있다[6, 3]. 이러한 시스템에서의 일괄처리 생산은 주문집약문제와 유사한 배치통합문제(batch consolidation problem)에서도 다루어졌다[1, 3, 5]. 배치통합문제는 주어진 주문을 생산하는데 사용된 배치의 수를 최소화하는 문제이다. 배치통합문제와 유사한 문제들에 대한 기존 연구결과는 명영수[1]에 기술되어 있다.

Hwang and Chang[5]은 주문집약문제가 NP-hard 이며, 동시에 max-SNP hard임을 보였다. 앞서 언급한대로 요구사항이 유사한 주문끼리만 동일한 배치에 포함될 수 있는데, 두 주문이 함께 배치에 묶일 수 있는지는 양립성그래프(compatibility graph)를 이용하여 표현한다. 양립성그래프에서는 마디(node)에 주문이 대응되고 두 주문이 같은 배치에 포함될 수 있으면 해당하는 마디 사이에 호(edge)가 존재하게 된다. 양립성그래프가 임의그래프(arbitrary graph)인 경우에는 주문집약문제는 NP-hard 이면서 동시에 max-SNP hard이지만, Hwang and Chang[4]은 양립성그래프가 나무(tree) 구조를 갖는 경우에는 주문집약문제를 다항시간(polynomial time) 내에 풀 수 있음을 보였다. 박종호 등[2]은 양립성그래프가 임의그래프인 경우에 Hwang과 Chang의 해법을 활용한 휴리스틱 해법을 제시하였다.

본 논문의 목적은 양립성그래프가 임의그래프인 주문집약문제에 대한 근사해법(approximation algorithm)을 개발하는 것이다. 최적화문제의 근사해법이란 어떤 형태의 입력 자료에도 구해진 해가 최적해의 일정비율 이내인 해법이다. 주문집약문제처럼 최대화문제인 경우에는 근사해법이 최대배치의 수에 ρ 이상인 해를 보장하는 경우 근사비율이 ρ인 근사해법, 또는 ρ-근사해법이라고 부른다. 본 논문에서는 주문집약문제에 대한 $\frac{1}{3}$-근사해법을 제시하기로 한다. 또한, 잘 알려진 First-Fit 해법이 주문집약문제에서는 의미 있는 결과를 도출하지 못함도 보이기로 한다.

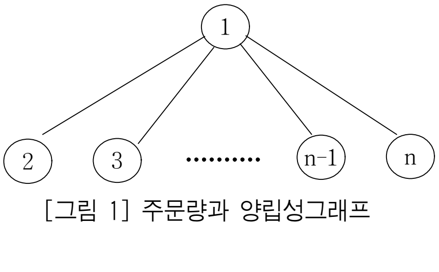
2. 근사해법

주문집약문제의 데이터를 표현하기 위하여 다음과 같은 기호를 정의하기로 한다. n개의 주문의 집합을 $V=\{1, \cdots, n\}$로, 주문 $i \in V$의 주문량을 $w(i)$로 표시한다. 배치가 구성되기 위해서는 배치에 포함되는 주문량의 합이 1 이상이 되어야 하는 것으로 가정한다. 또한 $w(i)>1$인 주문량도 허용한다. 주문 i와 j가 동일한 배치에 배정 가능한지 여부를 나타내기 위하여 주문의 집합 V를 마디의 집합으로 갖는 무향그래프(undirected graph) $G=(V, E)$를 활용하기로 한다. 이러한 그래프를 양립성그래프라고 부르고 호의 집합 $E=\{1, \cdots, m\}$은 다음과 같이 정의한다. 두 주문 i와 j에 해당하는 마디 $i \in V$와 $j \in V$ 사이에 호 $e=\{i, j\}$가 존재하면 두 주문은 양립 가능한, 즉 동일한 배치에 배정 가능한 주문임을 나타낸다. 하나의 배치에 포함될 수 있는 서로 다른 주문은 두 개까지이다. 구성된 배치 중 하나의 주문만이 포함된 배치는 단일배치(homogeneous batch), 두 개의 주문으로 구성된 배치는 혼성배치(heterogeneous batch)로 부르기로 한다.

주문집약문제와 같은 NP-hard 문제의 접근법 중 하나는 어떤 유형의 입력 자료에 대해서도 최적해의 일정비율(worst-case ratio) 이내인 해를 제공할 수 있는 근사해법을 개발하는 것이다. 본 연구에서는 주문집약문제에 대한 근사해법을 개발하기로 한다. 주문집약문제는 포장문제(packing problem)의 유형으로 분류할 수 있는데 이러한 문제에서 흔히 사용되는 근사해법에 First-Fit 해법이 있다. First-Fit 해법을 주문집약문제에 적용하면 주문을 순서대로 아직 1까지 채워지지 않은 배치에 주문량의 일부 또는 전부를 배정하여 1까지 채우고, 그러한 배치가 없는 경우에는 새로운 배치에 배정하는 방법이다. 그러나 이러한 First-Fit 해법이 주문집약문제에서는 일정한 값의 근사비율을 제공하지 못함을 보이기로 한다.

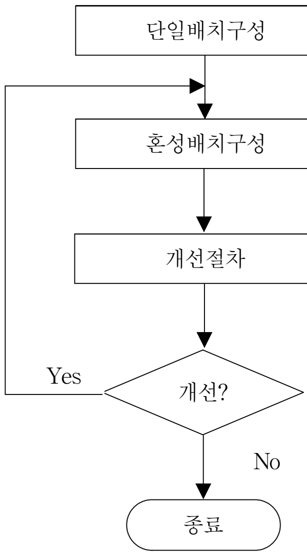
주문량이 $1-\epsilon$로 동일하고, 양립성그래프가([그림 1])과 같이 주어지는 n개의 주문이 있다고 가정하자. 주문번호별로 First-Fit 해법이 실행되는 경우에 주문 1의 전체 주문량과 주문 2의 주문량 중 ϵ만큼이 포함된 1개의 배치만이 구성될 수 있다. 그러나 최대 배치의 수는 $n-1$개이므로 근사비율은 n이 커짐에 따라 임의의 큰 값으로 발산하게 된다. 유사한 문제인 배치통합문제에서 First-Fit 해법이 1.5-근사해법이었던 것[1]을 고려하면 대조적인 결과라고 할 수 있다. 따라서 주문집약문제의 경우에는 의미 있는 근사해법을 얻기 위해서는 새로운 방법을 고려해야만 한다.

주문	1	2	…	…	…	n
주문량	1-ε	1-ε	…	…	…	1-ε



[그림 1] 주문량과 양립성그래프

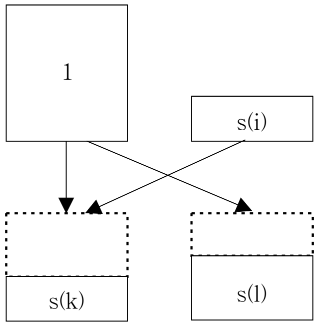
본 논문에서는 근사비율이 $\frac{1}{3}$인 근사해법을 제시하기로 한다. 우리의 해법은 초기절차(initial construction procedure)와 개선절차(improvement procedure)의 두 부분으로 구성되어 있다. 초기절차는 간단한 방법을 통해서 가능한 많은 단일 및 혼성배치를 만드는 절차인데, 단일배치 구성절차와 혼성배치 구성절차 두 부분으로 이루어져 있다. 개선절차는 초기절차가 완료된 뒤에 추가적인 배치구성을 위한 방법이다. 우리의 해법은 이 3가지 절차가 [그림 2]의 순서도처럼 실행된다. 단일배치 구성절차에서는 주문별로 주문량이 1 이상인 주문이 있는 경우에 단일배치를 구성한다. 혼성배치 구성절차에서는 양립성그래프의 호를 이루는 두 주문에 대해서 주문량의 합이 1 이상이면 혼성배치를 구성하게 된다. 각 주문 $i \in V$에 대해서 아직 완성된 배치에 배정되지 못하고 남은 주문량을 $s(i)$로 나타내기로 하자. 시작 때는 $s(i)=w(i),\ i \in V$이다. 단일배치 구성절차에서는 각 i에 대해서 $\lfloor w(i) \rfloor$만큼의 단일배치를 구성한다. 그러면 $s(i)=w(i)-\lfloor w(i) \rfloor,\ i \in V$가 되고, 모든 $s(i)$는 $0<s(i)<1$의 값을 갖게 된다. 혼성배치는 각 호 $e=\{i,j\} \in E$에 대해서 $s(i)+s(j) \geq 1$이면 혼성배치를 구성한다. 이 때 혼성배치에 배정되는 두 주문의 양은 주문번호가 앞선 주문을 먼저 배치한다. 즉 $i<j$이면, 배정 후 $s(i) \leftarrow 0,\ s(j) \leftarrow s(i)+s(j)-1$로 변화하게 된다.



[그림 2] 해법의 순서도

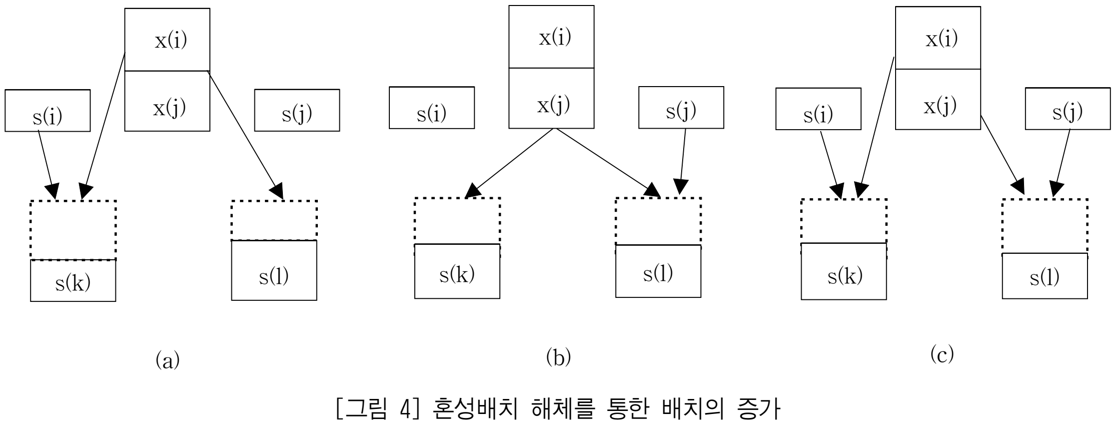
위의 초기절차가 완료되면 각 주문 $i \in V$에 대해서 $0 < s(i) < 1$이고, 두 주문 i와 j가 양립 가능하면 $s(i)+s(j) < 1$이 된다. 즉 이미 구성된 배치를 해체하지 않는 한 배치를 추가할 수 있는 방법은 없다. 그러나 이 초기절차만으로는 의미 있는 근사비율의 해를 얻을 수는 없다. 앞에서 예시한 [그림 1]의 주문들에 대해서 초기절차는 First-Fit 해법과 동일한 해를 산출한다. 즉 단일배치 구성절차에서는 전혀 배치가 구성되지 않고, 혼성배치 구성절차의 첫 단계에서 주문 1과 2에 대해서 하나의 배치를 만들게 된다. 이 후에 $s(1)=0$, $s(2)=1-2\epsilon$, 그리고 나머지 주문들은 $s(i)=1-\epsilon$이고, 이 후 더 이상의 배치는 만들어지지 않게 된다.

우리의 해법에서는 좀 더 나은 근사해를 도출하기 위하여 개선절차를 추가하였다. 개선절차는 기존에 만들어진 배치를 하나 해체하는 대신에 두 개 이상의 배치를 구성하기 위한 수단이다. 어떤 경우에 이러한 개선이 가능한지 살펴보기로 하자. 우선 해체할 배치가 단일배치인 경우를 먼저 생각해 보자. 주문 i를 포함하는 단일배치가 존재하고, 주문 i와 양립 가능한 주문 k와 주문 l이 존재한다고 가정하자. 만약에 단일배치를 해체하고, 주문 i와 k의 혼성배치와 주문 i와 l의 혼성배치를 각각 1개씩 만들 수 있다면 전체 배치의 수는 증가하게 된다. 이러한 관계가 [그림 3]에 표현되어 있다. [그림 3]의 개선이 가능하려면 $s(i)+s(k)+s(l) \geq 1$이면 가능하다.



[그림 3] 단일배치 해체를 통한 배치의 증가

이제 해체할 배치가 주문 i와 j로 구성된 혼성배치인 경우를 생각해 보자. 그리고 주문 i또는 j와 양립 가능한 주문 k와 l이 존재한다고 가정하자. 이 주문들의 양립 관계에 따라 3가지 서로 다른 형태의 개선이 가능하다. 즉 해체된 혼성배치에 포함된 i와 j중 한 종류만 사용하여 두 개의 새로운 혼성배치를 생성하는 경우와 i와 j 모두 사용하여 새로운 혼성배치 두 개를 생성하는 경우인데 이러한 관계는 [그림 4]에 표현되어 있다. [그림 4]처럼 주문 i와 j의 혼성배치에 포함된 i와 j의 주문량을 각각, $x(i)$와 $x(j)$로 표시하기로 하자. (a)와 (b) 유형의 개선이 이루어지려면, 각각, $x(i)+s(i)+s(k)+s(l) \geq 2$, $x(j)+s(j)+s(k)+s(l) \geq 2$를 만족해야 한다. (c) 유형의 개선이 이루어지려면, $x(i)+s(i)+s(k) \geq 1$과 $x(j)+s(j)+s(l) \geq 1$를 동시에 만족해야 한다. 이러한 개선절차는 Procedure Improve에 정리되어 있다.



[그림 4] 혼성배치 해체를 통한 배치의 증가

Procedure Improve

Input : $s(i)$, $i \in V$, and the current set of batches, L

Output : updated $s(i)$ and L

(Step 0) Set $CANDID = L$.

(Step 1) **if** $CANDID = \varnothing$ **then** return
else select a batch in $CANDID$;
if the selected bin is a homogeneous one with order i **then** go to Step 2
else (a heterogeneous one with orders i and j) go to Step 3;

(Step 2) **if** there exist a pair of orders $k, l \in V$ such that $\{i, k\}, \{i, l\} \in E$ and $s(i)+s(k)+s(l) \geq 1$ **then**
construct two heterogeneous batches, one with $x(i)=1-s(k)$ and $x(k)=s(k)$ and the other with $x(i)=1-s(l)$ and $x(l)=s(l)$;
set $s(i) \leftarrow s(i)+s(k)+s(l)-1$, $s(k) \leftarrow 0$, and $s(l) \leftarrow 0$;
replace the selected homogeneous batch by the two new heterogeneous batches in L;
return
else remove all homogeneous batches having order i from $CANDID$;
go to Step 1

(Step 3) **if** there exist a pair of orders $k, l \in V$ uch that $\{i, k\}, \{i, l\} \in E$ and $x(i)+s(i)+s(k)+s(l) \geq 2$ **then**
remove the selected heterogeneous batch from $CANDID$;
construct two heterogeneous batches, one with $x(i)=1-s(k)$ and $x(k)=s(k)$ and the other with $x(i)=1-s(l)$ and $x(l)=s(l)$;
set $s(i) \leftarrow x(i)+s(i)+s(k)+s(l)-2$, $s(k) \leftarrow 0$, and $s(l) \leftarrow 0$;
replace the selected heterogeneous batch by the two new heterogeneous batches in L;
return

(Step 4) **if** there exist a pair of orders $k, l \in V$ such that $\{j, k\}, \{j, l\} \in E$ and $x(j)+s(j)+s(k)+s(l) \geq 2$ **then**
remove the selected heterogeneous batch from $CANDID$;
construct two heterogeneous batches, one with $x(j)=1-s(k)$ and $x(k)=s(k)$ and the other with $x(j)=1-s(l)$ and $x(l)=s(l)$;
set $s(j) \leftarrow x(j)+s(j)+s(k)+s(l)-2$, $s(k) \leftarrow 0$, and $s(l) \leftarrow 0$;
replace the selected heterogeneous batch by the two new heterogeneous batches in L;
return

(Step 5) **if** there exist a pair of orders $k, l \in V$ such that $\{i, k\}, \{j, l\} \in E$ and $x(i)+s(i)+s(k) \geq 1$ and $x(j)+s(j)+s(l) \geq 1$ **then**
remove the selected heterogeneous batch from $CANDID$;
construct two heterogeneous batches, one with $x(i)=1-s(k)$ and $x(k)=s(k)$ and the other with $x(j)=1-s(l)$ and $x(l)=s(l)$;
set $s(i) \leftarrow x(i)+s(i)+s(k)-1$, $s(j) \leftarrow x(j)+s(j)+s(l)-1$, $s(k) \leftarrow 0$, and $s(l) \leftarrow 0$;
replace the selected heterogeneous batch by the two new heterogeneous batches in L;
return

(Step 6) remove the selected heterogeneous batch from $CANDID$;
go to Step 1.

개선절차 Improve의 Step 2는 [그림 3], Step 3, 4, 5는 각각 [그림 4]의 (a), (b), (c)의 과정을 수행하는 단계이다. 개선절차를 통해 배치의 추가가 이루어지지 않는 경우에는 input으로 주어진 변수 s와 배치 L에 변화가 전혀 없게 되고 전체 해법은 종결하게 된다. 개선절차에서 배치의 증가가 이루어지는 경우, Step 2, 3, 4, 5에서 변수 s가 재조정 된다. 재조정과정에서 알 수 있듯이 s의 값은 항상 1 미만의 값을 갖게 되어 단일배치가 추가될 가능성은 없다. 이처럼 개선절차를 통해서 단일배치는 늘어나지 않고, 혼성배치만 늘어나게 된다. Step 2에서 주문 i를 포함하는 단일배치의 해체를 통한 개선이 불가능하면 동일한 주문을 포함하는 단일배치는 더 이상 고려할 필요가 없기 때문에 모두 $CANDID$에서 제거한다. 개설절차에서 증가가 이루어진 경우에는 변수 s의 값이 변화하고(증가하는 경우도 발생), 새로운 혼성배치도 추가되므로 혼성배치 구성을 위한 절차와 개선절차를 반복해서 수행하는 것이 바람직하다. 이러한 모든 점을 감안해서 우리의 해법은 [그림 2]의 순서도와 같은 방식으로 개선절차를 반복적으로 수행한다. 우리의 해법은 다음과 같이 표현할 수 있다.

Algorithm OCP

Input : $w(i),\ i\in V$ and the compatibility graph $G=(V, E)$

Output : A set of batches, L

(Step 0) Set $L=\varnothing$ and $s(i)=w(i),\ i\in V$.

(Step 1) For each $i\in V$, construct $\lfloor s(i)\rfloor$ homogeneous batches, add them to L, and set $s(i)\leftarrow s(i)-\lfloor s(i)\rfloor$.

(Step 2) For each $e=\{i, j\}\in E$ such that $s(i)+s(j)\geq 1$, construct a heterogeneous batch with $x(i)=s(i)$ and $x(j)=1-s(i)$, add the batch to L, and set $s(j)\leftarrow s(i)+s(j)-1$ and $s(i)\leftarrow 0$.

(Step 3) Call **Improve**. If any improve, then go to Step 2. Otherwise, stop.

3. 근사비율의 증명

이제 제2장에서 제시한 근사해법은 근사비율이 $\frac{1}{3}$이고, 계산시간은 $O(n^3)$임을 보이기로 한다.

정리 1 : OCP는 근사비율 $\frac{1}{3}$인 근사해법이다.

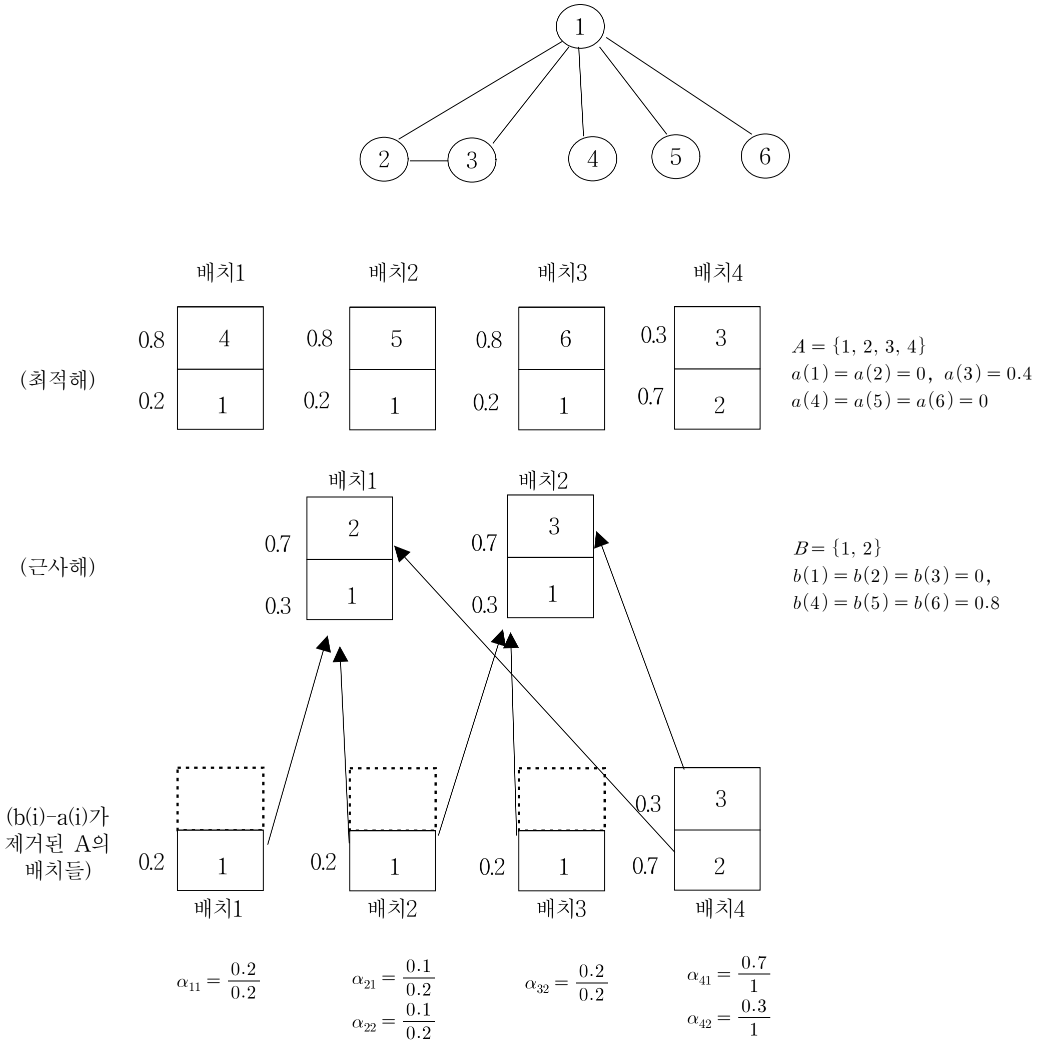
(증명) 임의의 주문집약문제에 대해서 최적으로 구성된 배치의 집합을 A, 그리고 OCP를 적용하여 얻어진 배치의 집합을 B라고 하자. 그러면 $|A|\leq 3|B|$임을 증명하면 된다. 각 주문 $i\in V$에 대해서 완성된 배치에 배정되지 못하고 남은 주문량이 A에서는 $a(i)$, B에서는 $b(i)$라고 하자. 당연히 $0\leq a(i),\ b(i)<1$이고, 각 $e=\{i, j\}\in E$에 대해서 $a(i)+a(j)<1$, $b(i)+b(j)<1$을 만족할 것이다. 또한, 어떤 $i\in V$에 대해서 $b(i)>a(i)$라면 주문 i의 주문량 중 $b(i)-a(i)$만큼은 A의 배치 중 하나에 포함되어 있을 것이다. 이제 각 $i\in V$에 대해서 $b(i)>a(i)$이면 $b(i)-a(i)$만큼 A의 배치로부터 순차적으로 제거하기로 한다. 이 과정에서, A의 배치 중 배치에 포함된 모든 주문량, 즉 1만큼, 제거되는 배치는 없다. 이러한 사실은 A의 배치가 주문 i를 포함하는 단일배치인 경우는 $b(i)<1$라는 사실로부터, A의 배치가 주문 i와 j를 포함하는 혼성배치인 경우에는 $b(i)+b(j)<1$라는 사실로부터 쉽게 알 수 있다. 그리고 이러한 제거 과정을 거친 뒤에는 각 주문별로 B의 배치들에 포함되어 있는 양이 A의 배치들에 남아 있는 양보다 많게 된다. 여기서부터 특별한 언급이 없는 경우에 A에 속한 배치의 주문량은 제거되고 남은 양만을 의미 하는 것으로 전제한다.

이제 A의 배치들에 남아 있는 주문량들을 B의 배치에 포함된 동일한 주문들의 주문량에 순차적으로 대응시켰다고 가정하자. A의 p번째 배치에 남아 있는 주문량 중 B의 q번째 배치에 대응된 양의 비율을 표시하기 위하여 α_{pq}를 다음과 같이 정의한다. A의 p번째 배치에 남아 있는 전체 주문량은 t이고, 이 중 주문 $i\in V$의 주문량이 c만큼 포함되어 있으며, B의 q번째 배치의 주문 i에 대응된 양을 $c'(\leq c)$이라고 하면

$\alpha_{pq} = \frac{c'}{t}$ 으로 정의한다. 이해를 돕기 위해서 [그림 5]에 α_{pq}의 계산 예를 제시하였다. 그림의 화살표는 주문이 대응된 관계를 나타낸다. α_{41}의 경우를 예를 들면 다음과 같이 설명할 수 있음. b(i)-a(i)를 제거한 A의 4번째(즉 p가 4) 배치에 남아있는 주문량과 B의 1번째 배치(즉 q가 1)에 대응된 양의 비율을 표시하는 것이 α_{41}이다. A의 4번째 배치의 주문 2의 주문량 0.7이 B의 1번째 배치에 속한 주문 2의 주문량 0.7에 대응되었다고 가정하면 $t=1$이고, $c=c'=0.7$이고, $\alpha_{41} = \frac{0.7}{1} = 0.7$이 된다.

따라서 모든 $p \in A$, $q \in B$에 대해서 $0 \le \alpha_{pq} \le 1$이고, 모든 $p \in A$에 대해서 $\sum_{q \in B} \alpha_{pq} = 1$이 성립된다. 만약에

주문	1	2	3	4	5	6
주문량	0.6	0.7	0.7	0.8	0.8	0.8



[그림 5] α_{pq}의 설명을 위한 예제

$|A| > 3|B|$ 라면, $\sum_{p\in A}\sum_{q\in B}\alpha_{pq} = |A| > 3|B|$ 이므로, 적어도 하나의 $q\in B$에 대해서 $\sum_{p\in A}\alpha_{pq} > 3$이 성립하여야 한다. 우리는 그러한 q가 존재할 수 없음을 보임으로써 $|A| \le 3|B|$ 임을 증명하려 한다. 이제 q가 단일배치인 경우와 혼성배치인 두 가지 경우로 나누어 $\sum_{p\in A}\alpha_{pq} > 3$은 불가능함을 보이기로 하자.

A의 배치들 중 q와 대응된 (즉 $\alpha_{pq} > 0$인) 배치들로 이루어진 A의 부분집합을 $A(q)$라고 하자. 그리고 $A(q)$의 배치들은 남아 있는 주문량이 적은 순서로 배열되어 있다고 가정하자.

(1) q가 주문 $i\in V$를 포함하는 단일배치인 경우.

$A(q)$에 속한 배치들은 주문 i만 포함하거나 주문 i및 i와 양립 가능한 주문 2종류를 포함하는 배치이다. $\sum_{p\in A}\alpha_{pq} > 3$이면, $A(q)$에 속한 처음 세 개의 배치들의 평균 주문량은 $\frac{1}{3}$미만이 된다. $A(q)$의 첫 번째 배치와 두 번째 배치의 주문량을 각각, $t(1)$, $t(2)$라고 하면, $t(1)+t(2) < \frac{2}{3}$이므로, 두 배치에서 제거된 주문량의 합은 1을 초과한다. 그런데 제거된 주문은 주문 i이거나 i와 양립 가능한 주문이다. $b(i) < 1$이므로 제거된 주문 중 주문 i를 제외한 주문 i와 양립 가능한 주문은 적어도 하나 이상 존재한다. 양립 가능 주문이 주문 j 하나라면 $b(i)+b(j) > 1$이므로 혼성배치 구성절차에서 혼성배치를 구성할 수 있었을 것이다. 양립 가능 주문이 주문 k와 l, 둘이였다면, $b(i)+b(k)+b(l) > 1$이므로 개선절차의 단일배치 해체과정(Improve의 Step 2)에서 새로운 혼성배치를 구성할 수 있었을 것이다. 따라서 $\sum_{p\in A}\alpha_{pq} > 3$인 단일배치 q는 존재하지 않는다.

(2) q가 주문 $i\in V$와 주문 $j\in V$를 각각 $x(i)$와 $x(j)$ 만큼 포함하는 혼성배치인 경우.

$A(q)$에 속한 배치들은 주문 i 또는 주문 j만을 포함하거나, 주문 i와 j, 주문 i와 양립 가능한 주문, 주문 j와 양립 가능한 주문의 두 종류를 포함하는 배치들이다. α_{pq}에서 q에 포함된 주문 i에 대응된 부분을 $\alpha_{pq}(i)$, 주문 j에 대응된 부분을 $\alpha_{pq}(j)$로 나누어 표기하기로 하자. 즉, $\alpha_{pq} = \alpha_{pq}(i)+\alpha_{pq}(j)$이다. [그림 5]의 예제에서 B의 1번째 배치는 주문 1과 주문 4의 혼성배치인데, α_{21}은 주문 1에만 대응되므로 $\alpha_{21}(1) = 0.5$, $\alpha_{21}(4) = 0$가 된다. $\sum_{p\in A}\alpha_{pq} > 3$이라고 가정하면, 다음의 두 가지 경우로 나누어 볼 수 있다.

(a) $\sum_{p\in A}\alpha_{pq}(i) > 2$ 또는 $\sum_{p\in A}\alpha_{pq}(j) > 2$인 경우 : 전자의 경우와 후자의 경우는 동일한 상황이므로 전자의 경우만 증명하기로 한다. $\sum_{p\in A}\alpha_{pq}(i) > 2$라고 가정하자. $A(q)$에서 주문 i를 포함하는 첫 번째 배치와 두 번째 배치의 주문량을 각각, $t(1)$, $t(2)$라고 하면, $t(1)$과 $t(2)$의 평균 중량은 $\frac{x(i)}{2}$ 미만이므로 $x(i) > t(1)+t(2)$이다. 따라서 두 배치에서 제거된 주문량의 합은 1을 초과하게 되고, 제거된 주문은 주문 i이거나 i와 양립 가능한 주문이다. $b(i) < 1$이므로 제거된 주문 중 주문 i를 제외한 주문 i와 양립 가능한 주문은 적어도 하나 이상 존재한다. 양립 가능 주문이 주문 k 하나라면 $x(i)+b(i)+b(k) > 2$이므로 $b(i)+b(k) > 1$이 성립한다. 그렇다면, 혼성배치구성절차에서 주문 i와 k로 이루어진 혼성배치를 구성할 수 있었을 것이다. 양립 가능 주문이 주문 k와 l 둘이였다면, $x(i)+b(i)+b(k)+b(l) > 2$이므로 개선절차의 혼성배치 해체과정(Improve의 Step 3)에서 새로운 혼성배치를 구성할 수 있었을 것이다.

(b) $1 < \sum_{p\in A}\alpha_{pq}(i) < 2$이고, $1 < \sum_{p\in A}\alpha_{pq}(j) < 2$인 경우 : $A(q)$에서 주문 i를 포함하는 첫 번째 배치의 주문량을 $t(1)$이라 하고, 주문 j를 포함하는 배치 중 앞에서 선택된 배치를 제외한 첫 번째 배치의 주문량을 $t(2)$라고 하자. 그러면 $x(i) > t(1)$이고 $x(j) > t(2)$이다. 주문량이 $t(1)$인 배치에서 제거된 주문의 종류는 주문 i이거나 i와 양립 가능한 주문이다. 또한 주문량이 $t(2)$인 배치에서 제거된 주문의 종류는 주문 j이거나 j와 양립 가능한 주문이다. 주문 i와 양립 가능한 주문

을 주문 k로, 주문 j와 양립 가능한 주문을 주문 l이라고 하면, $x(i)>t(1)$과 $x(j)>t(2)$로부터 $x(i)+b(i)+b(k)>1$과 $x(j)+b(j)+b(l)>1$이 성립함을 알 수 있다. 이 경우 개선절차의 혼성배치 해체과정 (Improve의 Step 5)에서 새로운 혼성배치를 구성할 수 있었을 것이다.

(a)와 (b)로부터 $\sum_{p\in A}\alpha_{pq}>3$인 혼성배치 q는 존재하지 않음을 알 수 있다.

(1)과 (2)에서 $\sum_{p\in A}\alpha_{pq}>3$인 어떤 $q\in B$도 존재하지 않으므로 $|A|\leq 3|B|$가 성립된다. □

이제 제2장에서 제시한 근사해법의 계산시간이 $O(n^3)$임을 보이기로 한다.

정리 2 : 알고리즘 OCP의 계산시간은 n개의 주문에 대해서 $O(n^3)$이다.

(증명) OCP의 Step 0와 Step 1은 $O(n)$시간 내에 완료된다. Step 2는 1회 시행에 $O(|E|)$, 즉 $O(n^2)$시간이 걸린다. Procedure Improve에서는 해체를 통해 개선이 가능한 배치들을 탐색하고 그러한 배치가 존재하면 새로운 배치를 구성한다. 탐색의 수는 CANDID에 속한 단일배치와 혼성배치의 수에 좌우된다. 단일배치의 경우 동일한 주문을 포함하는 배치는 한번만 탐색하므로, Improve 1회 실행에 탐색하는 단일배치의 수는 최대 n개다. 혼성배치의 수는 Improve 반복시행에 따라 변화하게 된다. 최초에 Step 2에서 만들어 질 수 있는 혼성배치의 수는 n개 이하이다. 따라서 Step 3에서 첫 번째 Improve를 실행할 때 탐색하는 혼성배치의 수는 최대 n개다. 그리고 Step 2와 3이 반복 시행되면 혼성배치의 수가 증가하게 된다. Step 2와 3은 Improve에서 배치가 하나 이상 증가될 때 다시 실행된다. 이제 Improve를 통해 증가될 수 있는 배치의 최대의 수를 고려해 보자. OCP의 Step 1에서 만들어진 배치의 수를 x라 하면 $x\geq\sum_{i\in V}\lfloor w(i)\rfloor$이다. 그리고 최적해에서 구해질 수 있는 배치의 수는 $\sum_{i\in V}\lceil w(i)\rceil$이하이다. $\sum_{i\in V}\lceil w(i)\rceil-\sum_{i\in V}\lfloor w(i)\rfloor\leq n$이므로 Improve의 반복시행은 n번 이하이고, Step 2와 3도 n번 이하로 반복된다. Improve를 통한 혼성배치의 증가는 2개 이하이고, Improve에서 개선이 일어난 뒤에 수행하는 Step 2에서 추가될 수 있는 혼성배치도 2개 이하이다. 후자의 사실은 Improve의 Step 2, 3, 4에서 s값이 증가할 수 있는 주문은 하나이고, Improve의 Step 5에서는 둘 이하라는 사실로부터 알 수 있다. 따라서 Improve를 실행할 때 탐색하는 혼성배치의 수도 $O(n)$개 이하이다.

단일 또는 혼성배치의 해체를 통해 개선이 가능한지를 점검하는 Improve의 Step 2-5도 $O(n)$시간 내에 완료된다. 각 Step에서 k와 l의 존재를 찾을 때, 모든 가능한 한 쌍의 주문을 점검하면, $O(n^2)$의 시간이 걸리는 것으로 생각될 수도 있다. 그러나 각 Step의 양립관계조건을 만족하는 주문들 중 s값이 큰 순으로 두 주문을 선택하면 된다. 왜냐하면, 이렇게 선택된 두 주문이 요건을 만족하는 k와 l이 되지 못하면 어떠한 다른 주문의 쌍도 요건을 만족 못하기 때문이다. 또한, Improve의 Step 2-5에서 해체를 위한 배치를 찾은 후 새로운 배치를 구성하는데 드는 시간도 모두 $O(n)$시간 내에 완료된다. 따라서 Improve의 실행, 즉 Step 3은 1회 시행에 $O(n^2)$시간이 걸린다. Step 2와 3이 n번 이하로 반복되므로, 알고리즘 OCP의 계산시간은 $O(n^3)$이내가 된다. □

4. 결 론

주문집약문제는 하나의 주문에 포함된 주문량을 두 개 이상의 배치에 나누어 생산할 수 있고, 하나의 배치에 포함시킬 수 있는 서로 다른 주문의 수가 2개로 제한되며, 함께 배치에 포함할 수 있는 주문과 포함할 수 없는 주문이 존재할 때, 일정 용량을 만족하는 배치의 수를 가능한 많이 구성하는 문제이다. 주문집약문제는 철강, 화학 등의 소재를 생산하는 시스템에서 발생하는 문제로 알려져 있다. 주문집약문제는 NP-hard이며, 동시에 max-SNP hard의 문제이다. 이러한 문제들에 대한 적절한 접근법 중 하나는 근사

해법을 개발하는 것이다. 근사해법이란 어떤 형태의 입력 자료에도 구해진 해가 최적해의 일정비율 이내의 값을 갖게 되는 해법이다. 본 논문에서는 주문집약문제의 $\frac{1}{3}$-근사해법을 개발하였다. 즉, 본 논문에서 제시된 해법을 사용하면 최대배치의 $\frac{1}{3}$ 이상인 배치를 항상 구성할 수 있다. 주문집약문제와 유사한 배치통합문제는 물론 상자포장문제(bin-packing problem) 등에 유용하게 사용된 First-Fit 해법이 주문집약문제에서는 유한한 값의 근사비율을 갖지 못함을 고려할 때 본 논문의 근사해법은 의미 있는 결과라고 평가할 수 있다. 한편으로는 주문집약문제의 첫 번째 근사해법의 개발이므로 향후 더 좋은 근사비율을 가진 해법의 연구의 시발점이 될 수도 있다고 생각된다.

참 고 문 헌

[1] 명영수, “일괄처리를 위한 배치통합문제의 근사해법”, 「한국경영과학회지」, 제38권, 제1호(2013), pp.61-67.

[2] 박종호, 임경국, 최봉하, “주문집약문제에 대한 휴리스틱 기법”, 「대한산업공학회지」, 제34권, 제4호(2008), pp.408-413.

[3] Chang, J., S.Y. Chang, S.-P. Hong, Y.-H. Min, and M.-J. Park, “Approximation of a batch consolidation problem,” *Networks*, Vol.58 (2011), pp.12-19.

[4] Hwang, H.C. and S.Y. Chang, “Order consolidation for match processing,” *Journal of Combinatorial Optimization*, Vol.9(2005), pp.121-138.

[5] Lee, K., S.Y. Chang, and Y. Hong, “Continuous slab caster scheduling and interval graphs,” *Production Planning & Control*, Vol.13, No.5(2004), pp.495-501.

[6] Tang, L., J. Liu, A. Ring, and Z. Yang, “A review of planning and scheduling system and methods for integrated steel production,” *European Journal of Operational Research*, Vol.133(2001), pp.1-20.