

순서 의존 준비시간 및 수송시간을 고려한 배관 없는 공정에 대한 스케줄링

복진광 · 박선원[†]

한국과학기술원 화학공학과
(1998년 8월 24일 접수, 1999년 4월 14일 채택)

Scheduling of Pipeless Processes Considering Sequence Dependent Setup Time and Transferring Time

Jin-Kwang Bok and Sunwon Park[†]

Dept. of Chemical Engineering, Korea Advanced Institute of Science and Technology
(Received 24 August 1998; accepted 14 April 1999)

요 약

본 논문에서는 생산순서에 의존한 준비시간을 갖는 다목적 무배관 공정에 대한 스케줄링 문제를 다루었다. 여러 종류의 생산물을 생산하는 다품종 혹은 다목적 공정에서의 준비시간은 조업을 맡는 장치 뿐 아니라 그 장치를 바로 전에 거친 생산물에 영향을 받는다. 효율적인 시간표현법을 위해 연속시간표현법을 이용하였고 이는 두 좌표계를 통해 나타내진다. 한 좌표는 각 생산물의 조업을 나타내는 stage block으로 구성되어 있고 또 다른 좌표는 각 장치가 조업에 참여하는 것을 time slot으로 나타내고 있다. 순서의존 준비시간을 고려하기 위해 이진변수를 새롭게 도입하기 보다는 논리적인 제약식을 이용하여 모델의 효율성을 높였다. 기존의 접근이 다품종 혹은 순서적인 다목적 공정에 한계가 있었던 반면, 제안되는 모델은 일반적인 다목적 공정에 대해서도 효과적으로 적용될 수 있다. 제안되는 모델의 효율성은 예제를 통해 보이고 있다.

Abstract – A practical scheduling problem for multipurpose pipeless processes where sequence dependent setup time should be considered is addressed. Setup times for production at units are dependent on the immediately previous production in multiple production processes as well as on the unit. A two-coordinate is presented for efficient continuous time representation where each product has deterministic processing stage blocks with duration for unit allocation and each processing unit has corresponding time slot. Instead of introducing transition 0-1 variables of which the number becomes enormous in multipurpose cases, we enforce the relation between the productions at units with logic-based constraints. While the existing models are restricted to multiproduct or sequential multipurpose processes in their applications, the proposed model is effectively applied to general multipurpose processes. The performances of the model and the solution method are illustrated through examples.

Key words: Pipeless Plant, Batch Scheduling, Continuous Time, Sequence Dependent Setup Time

1. 서 론

'90년대 들어서 화학산업은 시장 및 여러 가지 규제의 변화에 따라 새로운 국면을 맞이하고 있다. 엄격해진 환경규제, 품질조건의 까다로움, 업체들의 경쟁으로 인한 이윤감소 등은 현존하는 많은 화학산업으로 하여금 새로운 돌파구를 찾게 한다. 이는 작게는 기존의 공장시설을 부분 개조를 통한 효율증대를 꾀하지만 아예 공정자체를 새롭게 모색을 요구하는 경우도 발생하고 있다. 이러한 시점에서 등장하게 된 새로운 개념의 공정이 바로 배관 없는 공정이다. 배관 없는 공정은 말 그대로 복잡한 배관 구조의 특징을 갖고 있는 화학공장에서 배관을 없애거나 그 수를 상당히 줄임으로써 마치 유연 생산 시스템(Flexible Manufacturing System, FMS)의 형태를 갖게 한다. 따라서 각 단위장치 간에는 물질 수송을 위해, 배관을 대신한 새로운 운

송수단인 AGV(Automated Guided Vehicle)가 그 역할을 하게 된다. Fig. 1은 배관 없는 공정이 취하는 전형적인 배치도를 나타내고 있다. 필요한 단위조작은 해당되는 station을 거침으로써 이루어지게 되고 각 station간의 이동은 레일 위를 움직이는 AGV를 통해 이루어진다. 현재까지는 주로 윤활유, 페인트, 잉크 등의 비교적 간단한 공정을 갖고 있는 화학물질 생산에 이용되고 있으며, 국내에는 아직 그 기술이 개발되지 못한 상태이다. Niwa[1, 2]는 문헌으로 처음으로 배관 없는 공정을 소개하였고 투자비용, 유연 생산, 운용, 효율면에서 기술하고 있다. 특히, 실제 조업 중인 상태의 공정을 토대로 내린 결론은, 초기 투자 비용이 거대한 반면 조업비용이 현저히 줄어든다는 점을 강조하고 있다. 이는 다품종 소량생산의 공정 경우, 각 생산물이 바뀔 때마다 기존 배관공정에서는 장치 뿐 아니라 배관 세척으로 인한 비용이 큰 반면, 배관 없는 공정에서는 해당 유동장치만을 세척하기 때문에 조업비용의 감소 및 생산물의 오염도 감소에 기인한다. 더불어, 설계단계에서 각 장치들의 자동화가 용이하기 때문에 필요한 인력이 줄

[†]E-mail : swpark@hanbit.kaist.ac.kr

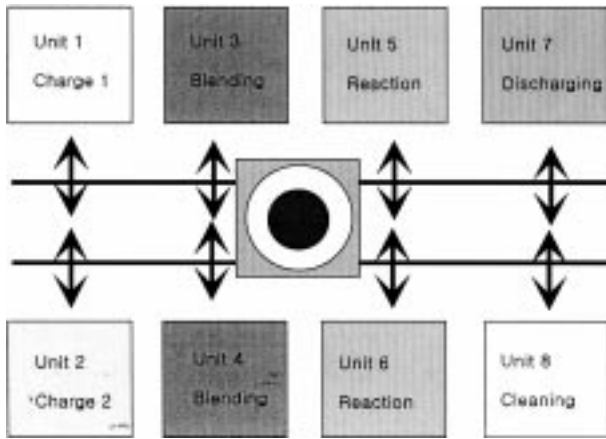


Fig. 1. Layout of a pipeless plant.

어 들게 되고, 조업 상의 실수로 인한 품질저하라든지 사고의 빈도를 크게 줄일 수 있다. 환경에 대한 규제가 날로 심각해져 가는 상황에서는 더욱 이러한 공정의 도입이 필수적이라 할 수 있겠다.

배관 없는 공정이 다품종 소량생산에 적합한 형태이기 때문에 생겨나는 문제는 바로 각 생산에 대한 스케줄링이라고 할 수 있다. Pantelides 등[3]은 배관 없는 공정의 물질수지식의 고려를 통해 체계적인 스케줄링을 위한 수학적 모델식을 제안하였다. 제안된 모델은 스케줄링에 따른 원료물질의 양 및 생산물의 양의 결정을 통해 총 조업 비용을 최소화한 엄격한 모델인 반면 이산시간표현법(discrete time representation)으로 구현했다는 모델링 상의 취약성이 있다. 이산시간표현법에 따른 문제점은 2절에서 자세하게 다루도록 하겠다. Reaff 등[4]은 배관 없는 공정의 스케줄링을 설계와 장소배치단계(layout)에서 미리 고려해야 하는 중요성을 거론하면서 모델식을 제안하였다. 이 모델식을 바탕으로 Gonzalez와 Reaff[5]는 얻어진 MILP의 결과를 이산사건 모사기(discrete event simulator)를 이용해 구현하였다. 또한 그들[6]은 또 다른 결과성능 평가를 위해서 dispatch rule을 제안해서 이를 MILP에 의한 결과와 비교하였다. 국내에 소개된 연구는 Lee 등[7]이 중간 저장조를 고려한 새로운 모델을 제시하였고, 유 등[8]은 생산계획과 공정 구성을 동시에 고려한 무배관 공정의 최적 설계 문제를 다루었다. Lee 등[7]이 제시한 모델은 모델링이 쉬운 다품종 공정에만 적용 가능하다는 한계가 있고, 유 등[8]의 모델은 변수의 수를 많이 줄이기는 했지만 여전히 이산시간표현법으로 인한 변수 개수의 문제를 갖고 있다. 이산시간표현법의 단점을 보완하고자 제안된 개념이 연속시간표현법(continuous time representation)이다. 특히 최근에는 관련된 시간 영역을 연속시간표현법을 통해 큰 규모의 문제를 풀고자 하는 노력이 진행 중이다[9, 10]. 특히 Bok과 Park[10]은 연속시간표현법을 배관 없는 공정에 효과적으로 적용하였다.

다품종 혹은 다목적 공정에 있어서 반드시 고려되어야 할 사항 중 하나가 장치에서 생산을 위한 준비시간이다. 준비시간이란 각 장치에서 조업해야 할 생산물이 바뀌기 때문에 새로운 생산물의 조업을 위해 필요한 시간이다. 보통의 경우 문제의 간결성을 위해 준비시간을 장치에만 의존한다고 가정하는 경우가 많다. 하지만 많은 회분식 공정의 경우 준비시간은 현재 조업에 해당하는 장치 뿐 아니라 그 장치에서 직전에 수행한 생산물도 영향을 미치게 된다. 예를 들어 색깔이 비슷한 생산물을 앞서 처리했다면 필요한 준비시간은 줄어드는 반면, 그 차이가 많이 나는 생산물을 처리한 경우는 새로운 조업을 위해 더 많은 시간의 준비가 필요하게 될 것이다. 이러한 이유로 생산 순서를 고려한 준비시간을 포함한 일정계획에 대한 연구결과가 많이 발표되어 왔다. 현재까지 발표된 모델들은 다품종 공정에 대한 모델

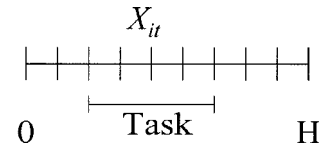


Fig. 2. Discrete time representation.

이거나 순서적인 다목적 공정(sequential multipurpose process)에 대한 것이 전부였다. 준비시간을 포함하는 모델에 대한 연구는 3절에서 살펴 보기로 한다.

본 논문에서는 순서에 의존한 준비시간을 고려한 배관 없는 공정에 대한 효과적인 일정계획모델을 제시하고자 한다. 준비시간을 나타내기 위해 새로운 0-1변수를 도입하기보다는 논리적인 제한식을 이용하여 모델의 효율성을 꾀하였고, 시간표현도 연속시간표현법을 이용함으로써 큰 규모의 문제에 적용 가능하도록 하였다.

2. 연속시간표현법

일정계획 문제가 수학적 모델식을 통한 접근법으로 가장 관심을 끄는 이유가 시간표현 때문이다[9]. 시간표현을 나타내는 방법에는 가장 범용적이면서도 모델링에 있어서 용이성을 갖는 이산시간표현법과 이를 보완하고자 하는 연속시간표현법이 있다. 이산시간표현법은 Fig. 2에서 나타나는 바와 같이 주어진 시간 범위(horizon)를 균일한 간격을 갖도록 나누어서 각 단위별로 사건이 일어남을 나타내는 이진변수를 도입함으로써 나타낼 수 있다. 이때 범위를 몇 개로 나눌 것인가는 주어진 문제의 시간단위를 나타내 줄 수 있는 최대공약수로 결정되게 된다. 가령 가공시간이 생산물 별로 {20분, 30분, 25분}, 준비시간이 {5분, 10분, 5분}이면 간격이 5분을 갖도록 전체 시간 범위를 나누어 주게 되는 것이다. 따라서 문제의 종류에 따라서는 그 간격이 상당히 줄어들 수 있고, 시간 범위를 나누는 단위의 수가 늘어남에 따라 각 단위에서 사건 발생을 표현하는 이진변수의 숫자가 엄청나게 늘어날 수 있다는 단점이 있다.

Fig. 3은 연속시간표현법을 보여 주고 있다[9, 10]. 연속시간표현법을

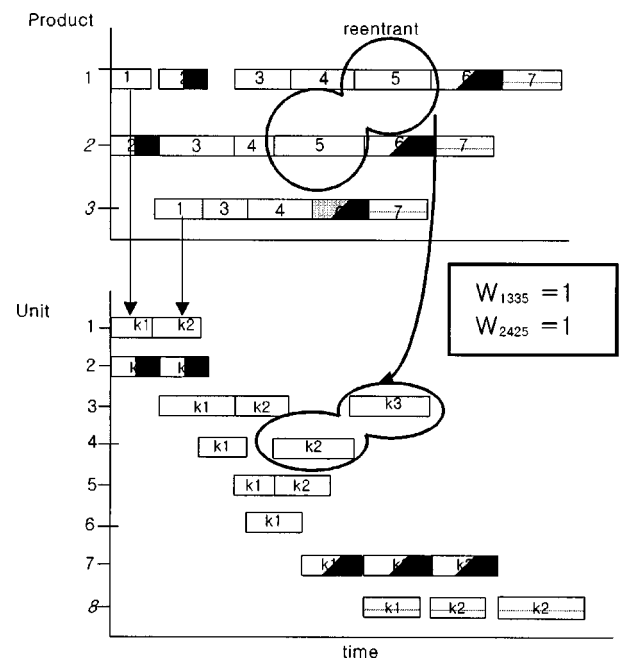


Fig. 3. Continuous time representation.

서는 사건의 시작과 끝만을 결정하는 이진변수만이 필요하게 된다.

Fig. 3에서 보여 주듯이, 각 생산물이 거치게 되는 하나의 단위 공정은 product축의 하나의 stage로 표현될 수 있다. 그러면 각 생산물마다, 완성될 때까지 거치게 되는 단위 공정의 개수만큼 stage의 수를 갖게 될 것이다. 동일한 장치를 다시 거치게 되는 경우도 product축에서는 하나의 다른 stage로 표현된다. 예를 들어, 생산물 1의 경우, 1단, 2단, 3단, 4단, 다시 3단, 5단, 6단을 거처서 완성이 된다. 하지만 두 차례 거치게 되는 3단은 서로 다른 두 개의 stage를 통해 구분 가능해진다. 생산물 2의 경우도 마찬가지로 이해할 수 있다. 장치의 입장에서 본 것은 unit축으로 표현한다. 각 장치에서는 처리하게 되는 하나의 단위공정을 unit축의 하나의 time slot으로 표현한다. 각각의 장치는 처리하는 단위공정의 수, 즉 거치게 되는 생산물의 수만큼의 time slot이 필요하게 된다. 스케줄링을 한다는 것은 product축의 stage를 처리할 수 있도록 unit축의 time slot으로 할당하는 것을 의미한다. 그림에서 생산물 1의 첫 번째 stage는 time slot k1에, 생산물 3의 첫 번째 stage는 time slot k2에 할당되는 것은, 생산물 1과 3이 장치 1을 통해 가공이 이루어진다는 것을 의미한다. 여기에 두 축간을 연결하는 이진변수의 도입을 통해 각 사건들의 시작과 끝을 결정해 줄 수 있다. 생산물 i 의 stage l 이 장치 j 의 time slot k 에 할당될 때, 이진변수 W_{ijkl} 은 1의 값을 지니게 된다. 생산물 i 의 stage l 의 시작시간과 종결시간은 다음과 같은 식으로 나타낼 수 있을 것이다.

$$TE_{il} = TSI_{il} + \sum_j \sum_k W_{ijkl} P_{ij} \quad \forall i, l \quad (1)$$

생산물의 stage가 장치의 time slot에 할당 될 때($W_{ijkl}=1$), 종결시간은 시작시간에 작업시간을 더하게 된다. 장치 j 의 time slot k 에서의 종결시간도 비슷하게 얻어진다.

$$TE_{jk} = TS_{jk} + \sum_i \sum_l W_{ijkl} P_{ij} \quad \forall i, k \quad (2)$$

생산물 i 의 stage l 과 연결되는 장치 j 의 time slot k 의 시작시간은 같아야 하는데 이는 다음의 식을 통해 이루어진다.

$$TS_{jk} - TSI_{il} \geq -U(1 - W_{ijkl}) \quad \forall i, j, k, l \quad (3)$$

$$TS_{jk} - TSI_{il} \leq U(1 - W_{ijkl}) \quad \forall i, j, k, l \quad (4)$$

생산물의 stage가 장치의 time slot에 할당될 때($W_{ijkl}=1$), 두 시작시간은 같아지지만 그 이외의 경우에 대해서는 완화(relaxed)된다.

각 생산물의 stage에 대해서는 장치의 time slot이 한 개 연결되어야 하는 조건은 다음의 식으로 표현된다.

$$\sum_j \sum_k W_{ijkl} = 1 \quad \forall i, l \quad (5)$$

하지만, stage에 해당하는 장치의 개수가 많을 때(multiple-unit process)는 한 stage에서 하나의 장치만 작업에 참여하므로 time slot들 중에는 생산물과 연결되는 것이 있고 그렇지 못한 것이 존재할 것이다. 그러므로 장치의 time slot에 따라서는 다음과 같은 식으로 나타내어진다.

$$\sum_i \sum_l W_{ijkl} \leq 1 \quad \forall j, k \quad (6)$$

Fig. 3에서의 화살표는 최적화 모델을 통해 생산물 축의 stage가 장치 축의 time slot에 연결되었을 때의 한 예를 나타내는 것이다. Pinto와 Grossmann[9]은 하나의 stage에 해당될 수 있는 장치의 숫자가 여러 개인 다중 장치문제(multiple-unit problem)를 다루었지만, 각 생산물이 동일한 생산경로를 가지면서 모든 stage를 거쳐야 한다는 가정을 두었다. Bok과 Park[10]은 이러한 제한성을 없애면서 일반적인 다목적 생산공정에 적용 가능하도록 모델을 수정하였다. Fig. 3에서의 5

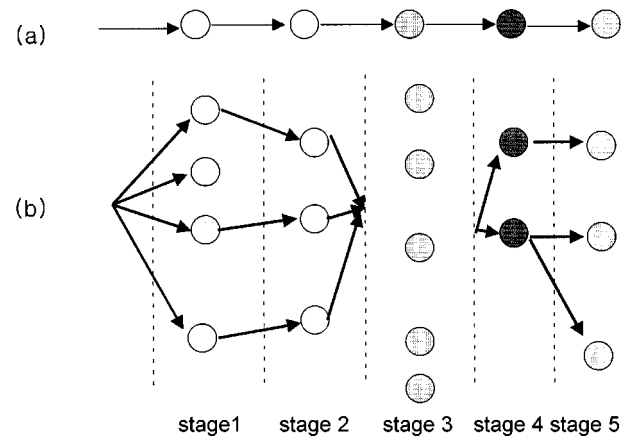


Fig. 4. (a) Single-unit assignment model, (b) multiple-unit assignment model.

번 stage는 3번 stage를 다시 거치게 되는 재진입 형태(re-entrant flow)이지만 stage의 번호를 달리 함으로써 표현이 가능하고 결과적으로는 곡선의 화살표처럼 연결되는 결과를 얻어낼 수 있다.

3. 준비시간의 고려

회분식 공정에 있어서 준비시간을 고려한 문제들을 분류해 보면, 크게 단일장치 할당모델(single-unit assignment model)과 다중장치 할당모델(multiple-unit assignment model)로 분류할 수 있다[11]. Fig. 4에서 보여주듯이, 전자의 경우는 각 stage마다 역할을 수행하는 장치의 개수가 하나인 경우이고 후자의 경우는 두 개 이상인 경우를 포함하는 문제이다.

우선 단일장치 할당모델에 대한 준비시간 고려 모델을 살펴 보자. Sahinidis와 Grossmann[12]은 time slot k 에 선행 생산물 i 와 후행 생산물 i' 의 할당을 나타내는 이진변수 $Z_{i'k}$ 를 도입하였다. 그래서 각 생산물 i 의 time slot k 에 할당을 나타내는 이진변수 X_{ik} 와의 관계를 다음과 같이 나타내었다.

$$Z_{i'k} \geq X_{ik} + X_{i'k+1} - 1 \quad \forall i, i', k \quad (7)$$

즉 생산물 i 가 time slot k 에 할당되고 생산물 i' 가 그 바로 다음 time slot인 $k+1$ 에 할당되었을 때, $Z_{i'k}$ 가 1의 값을 지니게 되는 것이다. 동일한 의미로 다음과 같이 나타낼 수도 있다.

$$\sum_i Z_{i'k} = X_{i'k+1} \quad \forall i', k \quad (8)$$

$$\sum_{i'} Z_{i'k} = X_{ik} \quad \forall i, k \quad (9)$$

이진변수 $Z_{i'k}$ 의 도입으로 순서 의존 준비시간은 쉽게 고려될 수 있다. 장치 j 에서 생산물 i 의 뒤를 따르는 생산물 i' 의 준비시간을 $S_{i'j}$ 이라고 하면 장치 j 의 time slot k 에서의 종결시간은 다음과 같이 나타내어 질 것이다.

$$TE_{jk} = TS_{jk} + \sum_i \sum_{i'} Z_{i'k} (P_{ij} + S_{i'j}) \quad \forall j, k \quad (10)$$

Jung 등[13]은 완료시간 알고리즘(completion time algorithm)을 이용하여 순서 의존 준비시간을 고려하였다. 장치 j 의 time slot k 에서의 완료시간을 C_{jk} 이라고 하고 생산물 i 의 time slot k 로의 연결을 의미하는 이진변수를 X_{ik} 라고 하면 완료시간의 제약식을 통해 준비시간을 포함시킬 수 있다.

$$C_{jk} \geq C_{j,k-1} + \sum_i P_{ij} X_{ik} + \sum_{i'} S_{i'j} X_{i,k-1} X_{i'k} \quad \forall j, k \quad (11)$$

하지만 이진변수들의 곱이 포함되어 있기 때문에 모델식이 비선형성을 갖게 되어 여러 난점을 가져올 수 있다. 이에 Byun 등[14]은 time slot k 에서 직전에 거친 생산물과 현재 거치는 생산물의 관계를 나타내는 이진변수 $Z_{ii'k}$ 를 도입함으로써 비선형성을 없앴다. 즉, 이진변수가 다음의 조건을 만족하면,

$$Z_{ii'k} \geq X_{ik} + X_{i',k+1} - 1 \quad \forall i, i', k \quad (12)$$

다음과 같이 비선형성을 없애면서 준비시간을 고려할 수 있다.

$$C_{jk} \geq C_{j,k-1} + \sum_i P_{ij} X_{ik} + \sum_{i'} S_{ii'} Z_{ii'k} \quad \forall j, k \quad (13)$$

다중장치 할당문제는 Sahinidis와 Grossmann[12]의 모델을 Pinto[11]가 확장하여 발표하였다. 장치의 time slot에서의 생산순서를 고려하기 위하여 Z_{ijk} 를 도입하였다. 즉, 장치 j 의 time slot k 에서 생산물 i 가 time slot $k+1$ 에서 생산물 i' 가 할당되면 위의 변수는 1의 값을 갖게 된다. 생산물 i 가 장치 j 의 time slot k 에 할당되면 1의 값을 갖는 이진변수 X_{ijk} 를 이용하면 다음과 같은 관계식으로 표현할 수 있다.

$$\sum_i Z_{ii'jk} = X_{i'ik+1} \quad \forall i', k \quad (14)$$

$$\sum_i Z_{ii'jk} = X_{ijk} \quad \forall i, k \quad (15)$$

그러면 종료시간과 시작시간의 관계식을 통해 준비시간을 고려할 수 있다.

$$TE_{jk} = TS_{jk} + \sum_{i'} \sum_i Z_{ii'jk} (P_{ij} + S_{ii'}) \quad \forall j, k \quad (16)$$

Gooding 등[15]도 다중장치 할당문제에서 준비시간을 고려한 모델을 발표하였지만 이산시간표현법의 비효율성으로 본 논문에서는 그에 대한 설명은 생략하기로 한다.

4. 모델식

본 논문에서 제시하는 모델은 다목적 공정의 다중장치 할당문제를 위한 것이다. Bok과 Park[10]이 제안한 모델을 확장하여 준비시간을 고려하도록 하였다.

목적식은 Makespan인 MS를 최소화하기 위한 것으로 다음과 같이 나타낼 수 있다.

$$\text{Minimize MS} \quad (17)$$

$$MS \geq TE_{il} \quad \forall i \quad (18)$$

식 (18)은 MS가 가장 순서가 늦은 생산물에 대한 최종 stage에서의 종료시간과 같다는 것을 의미한다.

다중장치 할당문제에서는 동일한 해가 반복되는 binary degeneracy가 최적해를 찾는데 시간을 지연시키게 된다[9]. 이런 이유로 목적식에 다음 항을 추가하게 된다[9, 10].

$$\varepsilon \sum_{i \in I_j} \sum_{l \in (L_i \cap L_j)} \sum_{j \in (J_i \cap J_l)} \sum_{k \in K_j} W_{ijkl}$$

여기서 ε 은 아주 작은 수를 의미한다.

그런 경우의 목적식은 다음과 같이 표현될 것이다.

$$\text{Minimize MS} + \varepsilon \sum_{i \in I_j} \sum_{l \in (L_i \cap L_j)} \sum_{j \in (J_i \cap J_l)} \sum_{k \in K_j} W_{ijkl} \quad (17')$$

모든 생산물의 stage는 장치의 time slot 중 하나에 반드시 연결되어야 한다. 그러므로,

$$\sum_{j \in (J_i \cap J_l)} \sum_{k \in K_j} W_{ijkl} = 1 \quad \forall i, l \in (L_i \cap L_j) \quad (19)$$

본 모델이 다루고자 하는 문제는 다중장치 할당문제이기 때문에 어떤 장치의 time slot은 생산물의 stage에 연결되지 못하는 경우도 발생할 수 있다.

$$\sum_{i \in I_j} \sum_{l \in (L_i \cap L_j)} W_{ijkl} \leq 1 \quad \forall j \in (J_i \cap J_l), k \in K_j \quad (20)$$

만약 각 장치의 time slot의 수를 미리 정확하게 계산할 수 있다면 부등식은 등식으로 바뀔 것이다. 또한, 동일 장치에 있어 time slot의 연결은 선행 time slot이 연결된 후에 가능하다고 본다면 다음의 두 식을 통해 표현이 가능하다.

$$\sum_{i \in I_j} \sum_{l \in (L_i \cap L_j)} W_{ijkl} + y_{jk} = 1 \quad \forall j \in (J_i \cap J_l), k \in K_j \quad (21)$$

$$y_{ik} \leq y_{j,k+1} \quad \forall j \in (J_i \cap J_l), k \in K_j \quad (22)$$

종료시간은 시작시간에 가공시간을 추가함으로써 표현될 수 있다.

$$TE_{il} = TS_{il} + \sum_{j \in (J_i \cap J_l)} \sum_{k \in K_j} (W_{ijkl} P_{ijl} + TR_{ijl}) \quad \forall j, l \in (L_i \cap L_j) \quad (23)$$

Bok과 Park[10]이 설명했듯이 stage에 대한 정보를 추가함으로써 재진입이 있는 공정까지 다룰 수 있는 모델로 나타낼 수 있다. 서로 다른 종류의 생산물이 순서대로 동일 장치를 지나게 되면 준비시간을 고려해 주어야 한다. 생산물 i 의 stage l 이 장치 j 의 time slot k 에 연결되고, 또 다른 생산물 i' 의 stage l' 가 장치 j 의 time slot $k+1$ 에 연결된다고 하면 생산물 i 의 stage l 의 시작시간에서 준비시간 $S_{ii'l'}$ 를 추가하게 된다. 그럼 식 (23)과 같이 나타낼 수 있을 것이다.

$$TS_{i'l'} \geq TE_{il} + S_{ii'l'} - U(2 - W_{ijkl} - W_{i'j,k+1,l'}) \quad \forall i, i', l \in (L_i \cap L_j), l' \in (L_{i'} \cap L_j), j \in (J_i \cap J_l \cap J_{i'} \cap J_{l'}) \quad (24)$$

식 (24)를 통해 장치에서 생산물의 순서를 고려한 준비시간이 포함될 수 있다는 점을 주목해야 한다. 이는 [11, 12, 14]의 모델처럼 새로운 변수를 도입하는 것이 아니고 논리적인 제약식을 통해 나타내고 있다. 만약 다목적 공정에서 새로운 변수를 도입하게 되면 $Z_{ii'jkl'}$ 의 변수가 필요하게 되고 이는 계산상의 복잡도를 크게 늘릴 것이다. 간단한 계산만으로도 (생산물의 수) × (장치의 수) × (time slot의 수) × (stage의 수) 만큼의 0-1 변수 혹은 연속변수가 증가하기 때문이다.

여기서 U 는 큰 값의 상수로 식 (24) 보다 tight하기 위해서 다음의 값으로 한다.

$$U = \sum_{i \in I_j} \sum_{j \in J_i} \max_{l \in (L_i \cap L_j)} \{P_{ijl}\} \quad (25)$$

동일한 생산물에 있어서, 순서가 늦은 stage의 시작시간은 선행 stage의 종료시간 보다 늦어야 한다.

$$TE_{il} \leq TS_{il^*} \quad \forall i, l \in L_i, l' \in L_i \quad (26)$$

장치들의 time slot의 개수를 정확히 예측해서 모델을 세우게 되면 y_{jk} 가 필요 없게 되고, 동일장치의 선행 time slot이 먼저 연결되어야 한다는 식 (22)가 없어지게 되므로 다음의 식으로 이를 대신한다.

$$\sum_{i \in I_j} \sum_{l^* \in (L_i \cap L_j)} W_{i^*j,k+1,l^*} \leq \sum_{i \in I_j} \sum_{l \in (L_i \cap L_j)} W_{ijkl} \quad \forall j \in (J_i \cap J_l), k \in K_j \quad (27)$$

변수의 영역에 대한 제한 조건은 다음과 같다.

$$W_{ijkl} \in \{0, 1\} \quad (28)$$

$$y_{jk} \geq 0 \quad (29)$$

$$TS_{il}, TE_{il} \geq 0 \quad (30)$$

최적화 모델은 GAMS[16]를 통해 구현되었고, MILP solver로는 OSL[17]을 이용하였다. 본 모델은 SUN SPARC10의 컴퓨터를 이용

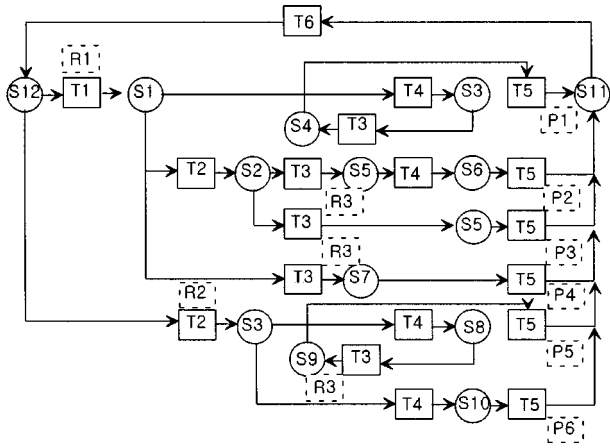


Fig. 5. State-task-network for example 1.

Table 1. Processing times for example 1

Unit	Product					
	1	2	3	4	5	6
1	5	16	4	3	-	-
2	-	14	6	-	4	-
3	5	16	-	-	-	-
4	-	-	4.5	7.5	-	-
5	7.5	-	-	-	-	6.5
6	-	5	-	-	6	-
7	5	5	5	5	5	5
8	5	5	5	5	5	5

Table 2. Setup times for example 1

Sequence	Setup times at each unit							
	1	2	3	4	5	6	7	8
2-1	0.3	0.2	0.4	0.7	0.6	0.6	0.6	0.4
3-1	0.4	0.3	0.4	0.2	0.6	0.4	0.6	0.4
4-1	0.3	0.3	0.5	0.3	0.4	0.3	0.4	0.5
5-1	0.6	0.4	0.6	0.3	0.3	0.6	0.3	0.6
6-1	0.5	0.6	0.6	0.4	0.4	0.7	0.4	0.6
1-2	0.3	0.4	0.4	0.5	0.4	0.2	0.4	0.4
3-2	0.4	0.3	0.3	0.2	0.5	0.6	0.5	0.3
4-2	0.4	0.6	0.6	0.3	0.6	0.6	0.6	0.6
5-2	0.5	0.7	0.7	0.3	0.6	0.4	0.6	0.7
6-2	0.6	0.2	0.2	0.4	0.4	0.3	0.4	0.2
1-3	0.6	0.5	0.6	0.6	0.3	0.4	0.7	0.6
2-3	0.4	0.6	0.6	0.4	0.3	0.4	0.2	0.4
4-3	0.3	0.6	0.4	0.3	0.5	0.5	0.3	0.3
5-3	0.6	0.4	0.3	0.6	0.2	0.6	0.3	0.6
6-3	0.7	0.6	0.6	0.7	0.3	0.6	0.4	0.7
1-4	0.2	0.7	0.7	0.2	0.3	0.4	0.5	0.2
2-4	0.3	0.2	0.2	0.6	0.4	0.3	0.2	0.6
3-4	0.3	0.3	0.3	0.6	0.4	0.6	0.3	0.6
5-4	0.4	0.3	0.3	0.4	0.5	0.7	0.3	0.4
6-4	0.4	0.5	0.4	0.3	0.6	0.2	0.4	0.3
1-5	0.5	0.2	0.5	0.7	0.7	0.3	0.6	0.3
2-5	0.6	0.3	0.2	0.2	0.2	0.3	0.4	0.3
3-5	0.6	0.3	0.3	0.3	0.3	0.5	0.3	0.5
4-5	0.4	0.4	0.3	0.3	0.3	0.2	0.6	0.2
6-5	0.6	0.4	0.4	0.4	0.4	0.3	0.7	0.3
1-6	0.7	0.5	0.4	0.5	0.5	0.3	0.2	0.3
2-6	0.2	0.6	0.5	0.2	0.2	0.4	0.6	0.4
3-6	0.3	0.6	0.6	0.3	0.3	0.4	0.6	0.4
4-6	0.3	0.4	0.6	0.3	0.3	0.5	0.4	0.5
5-6	0.5	0.6	0.4	0.4	0.4	0.6	0.3	0.6

하여 예제들에 적용되었다.

5. 예 제

5-1. 예제-1

본 예제는 Fig. 5에서 주어지듯이 각 생산물의 생산경로가 복잡한 배관 없는 공정이다. 하나의 생산물을 처리한 moving vessel은 다음 생산물을 처리하기 위해 세척공정(T6)을 거쳐야 한다. 6개의 생산물은 {T1, T4, T3, T5, T6}, {T1, T2, T3, T4, T5, T6}, {T1, T2, T3, T5, T6}, {T1, T3, T5, T6}, {T2, T4, T3, T5, T6}, 그리고 {T2, T4, T5, T6}의 작업(task)을 거치게 되고, T1에서 T6의 6개의 작업에 참여하는 장치는 각각 {j1}, {j2}, {j3, j4}, {j5, j6}, {j7}, 그리고 {j8}이다. 가공시간과 준비시간 및 이동시간에 대한 데이터는 Table 1, 2 그리고 3에 각각 주어져 있고, 이에 대한 결과는 Table 4와 Fig. 6에 나타내었다.

Table 3. Transferring times for example 1

Product-stage	Transferring time at each unit							
	1	2	3	4	5	6	7	8
1-1	0.05							
2-1	0.05							
3-1	0.05							
4-1	0.05							
5-1	0.05							
6-1	0.05							
1-2		0.05						
2-2		0.05						
3-2		0.05						
4-2		0.05						
5-2		0.05						
6-2		0.05						
1-3			0.05	0.05				
2-3			0.05	0.05				
3-3			0.05	0.05				
4-3			0.05	0.05				
5-3			0.05	0.05				
6-3			0.05	0.05				
1-4					0.05	0.05		
2-4					0.05	0.05		
3-4					0.05	0.05		
4-4					0.05	0.05		
5-4					0.05	0.05		
6-4					0.05	0.05		
1-5			0.07	0.08				
2-5			0.07	0.08				
3-5			0.07	0.08				
4-5			0.07	0.08				
5-5			0.07	0.08				
6-5			0.07	0.08				
1-6								0.05
2-6								0.05
3-6								0.05
4-6								0.05
5-6								0.05
6-6								0.05
1-7								0.01
2-7								0.01
3-7								0.01
4-7								0.01
5-7								0.01
6-7								0.01

Table 4. Computational statistics of example 1

0-1 variables	Constraints	Continuous variables	Makespan	CPU[s]	Iterations
155	652	251	78.53	314.54	18850

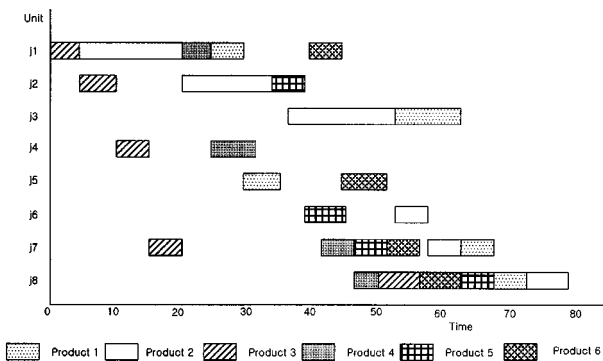


Fig. 6. Gantt chart for example 1.

5-2. 예제-2

본 예제에서는 5단계를 거쳐 5가지 종류의 생산물을 만들어 내는 무배관 공정에 대한 예제이다. 다중 장치 할당 예제[9]에서 주어진 문제 형태를 무배관 형태로 문제를 구성하였다. 공정시간은 [9]의 데이터를 사용하였고, 여기에 준비시간과 이동시간을 추가하였다. 각 단계별로 할당되는 장치의 집합은 $\{j1, j2, \dots, j6\}$, $\{j7, j8, j9\}$, $\{j10, j11, \dots, j19\}$, $\{j20, j21, j22\}$, 그리고 $\{j23, j24, j25\}$ 로 주어져 있다. 제한조건으로는 생산물 2의 경우 1, 2, 3단계를 거쳐 4단계에서는 j22, 5단계에서는 j25에서만 조립이 가능하고, 나머지 생산물은 1, 2, 3단계를 거쳐 4, 5단계에서 이들의 장치를 제외한 장치들에서만 조립이 이루어져야 한다는 것이다. Table 5는 공정시간 데이터, Table 6은 준비시간, Table 7은 이동시간 데이터를 나타내고 있다. 예제에 대한

Table 5. Processing times for example 2

Unit	Processing time for each production				
	1	2	3	4	5
1	18.1	23.0	18.1	20.0	17.0
2	18.1	23.0	18.1	20.0	17.0
3	18.1	23.0	18.1	20.0	17.0
4	18.1	23.0	18.1	20.0	17.0
5	18.1	23.0	18.1	20.0	17.0
6	18.1	23.0	18.1	20.0	17.0
7	14.0	14.0	14.0	11.0	14.0
8	5.0	5.0	5.0	5.0	5.0
9	5.0	5.0	5.0	5.0	5.0
10	12.0	12.0	24.0	12.0	12.0
11	12.0	12.0	24.0	12.0	12.0
12	12.0	12.0	24.0	12.0	12.0
13	12.0	12.0	24.0	12.0	12.0
14	12.0	12.0	24.0	12.0	12.0
15	12.0	12.0	24.0	12.0	12.0
16	12.0	12.0	24.0	12.0	12.0
17	12.0	12.0	24.0	12.0	12.0
18	-	12.0	-	-	-
19	-	12.0	-	-	-
20	9.5	-	9.3	7.9	12.5
21	9.5	-	9.3	7.9	12.5
22	-	100.0	-	-	-
23	24	-	24	24	24
24	24	-	24	24	24
25	-	48.0	24	24	24

최적 결과는 Table 8과 Fig. 7에서 보여주고 있다.

6. 결 론

본 논문에서는 순서 의존 준비시간을 포함하는 배관 없는 공정에

Table 6. Setup times for example 2

Unit	Production sequences																			
	2-1	3-1	4-1	5-1	1-2	3-2	4-2	5-2	1-3	2-3	4-3	5-3	1-4	2-4	3-4	5-4	1-5	2-5	3-5	4-5
1	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3	1	2
2	2	3	2	3	2	3	2	3	2	3	2	3	2	3	2	3	2	3	2	3
3	4	3	2	1	4	3	2	1	4	3	2	1	4	3	2	1	4	3	2	1
4	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
5	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1
6	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
7	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3	1	2
8	3	2	3	2	3	2	3	2	3	2	3	2	3	2	3	2	3	2	3	2
9	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3	1	2
10	2	3	2	3	2	3	2	3	2	3	2	3	2	3	2	3	2	3	2	3
11	4	3	2	1	4	3	2	1	4	3	2	1	4	3	2	1	4	3	2	1
12	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
13	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1
14	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
15	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3	1	2
16	3	2	3	2	3	2	3	2	3	2	3	2	3	2	3	2	3	2	3	2
17	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3	1	2
18	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
19	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
20	-	2	2	2	-	-	-	-	2	-	2	2	2	-	2	2	2	-	2	2
21	-	1	2	1	-	-	-	-	2	-	2	1	2	-	2	1	2	-	2	1
22	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
23	-	2	3	1	-	-	-	-	3	-	2	3	1	-	3	1	2	-	1	2
24	-	2	3	2	-	-	-	-	3	-	3	2	3	-	3	2	3	-	3	2
25	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

Table 7. Transferring times for example 2

Production-stage	Processing time at each unit							
	j1-j6	j7-j9	j10-j17	j18-j19	j20-j21	j22	j23-j24	j25
1-1	0.1							
2-1	0.1							
3-1	0.2							
4-1	0.3							
5-1	0.2							
1-2		0.1						
2-2		0.3						
3-2		0.2						
4-2		0.3						
5-2		0.3						
1-3			0.2					
2-3			0.4					
3-3			0.1					
4-3			0.2					
5-3			0.2					
1-4					0.4		0.1	
2-4				0.3		0.3		
3-4					0.4		0.2	
4-4					0.2		0.3	
5-4					0.2		0.4	
1-5								0.3
2-5								
3-5								
4-5								
5-5								

Table 8. Computational statistics of example 2

0-1 variables	Constraints	Continuous variables	Makespan	CPU[s]	Iterations
206	928	353	189.2	8.3	2540

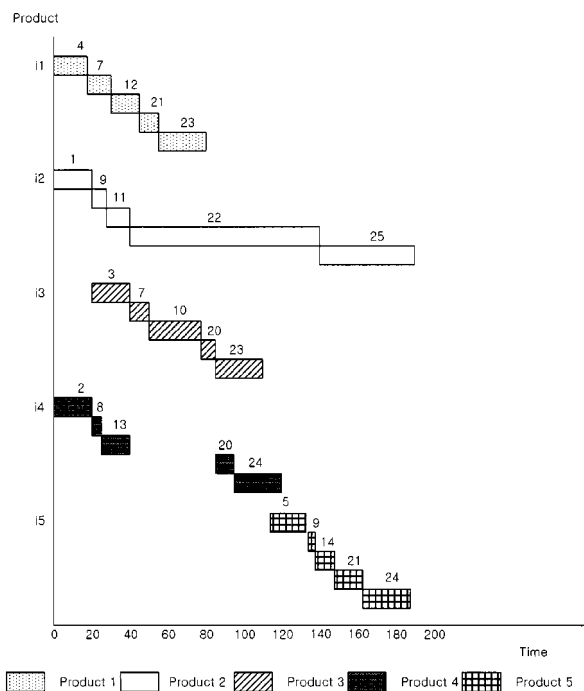


Fig. 7. Gantt chart for example 2.

대한 일정계획 모델을 제시하였다. 배관 없는 공정에서는 여러 가지의 생산물을 순서적으로 처리해야 되기 때문에 각 장치에서 순서에

의존한 준비시간을 고려하는 것이 중요하다. 효율적인 시간표현을 위해 연속시간표현법을 이용하였고, 이는 MILP 모델로 나타내었다. 기존의 연구들이 대부분 준비시간을 고려하기 위해 이진변수를 새롭게 도입하였지만 본 모델에서는 논리적인 제한식을 이용해 이를 포함할 수 있었다. 특히, 각 생산물이 다양한 경로를 갖게 되는 다목적 회분식 공정에 적용시킴으로써 그 효율성을 보였다.

감 사

본 연구는 한국과학재단 지정 우수 연구센터인 포항공대의 지능자동화 연구센터를 통한 과학재단의 부분적인 재정지원에 의해 수행되었으므로 이에 감사를 드립니다.

사용기호

- i, i^* : product
- j : unit
- k : time slot for equipment
- k_j : last time slot defined for unit j
- l, l^* : processing stage
- l_i : last stage for product i
- I : set of products
- I_j : set of products which can be processed in unit j
- J : set of units
- J_i : set of units which can process product i
- J_l : set of units which belongs to stage l
- K : set of time slots
- K_j : set of time slots for unit j
- L : set of stages
- L_i : set of stages involved in product i
- L_j : set of stages corresponding to unit j
- C_{jk} : completion time of time slot k at unit j
- M : number of units
- MS : makespan
- N : number of products
- P_{ijl} : processing time of stage l of product i in unit j
- $S_{ii'lj}$: changeover time for production stage l of product i immediately after i' in unit j
- TE_{il} : start time for product i in stage l
- TS_{il} : end time for product i in stage l
- TR_{ijl} : transferring time of product i to unit j which belongs to stage l
- U : upper bound on start times
- W_{ijkl} : binary variable that assigns stage l of product i to time slot k of unit j
- X_{ik} : binary variable that assigns product i to time slot k
- y_{jk} : slack variable for Eq. (2)
- $Z_{ii'k}$: binary variable that product i' follows product i at time slot k
- ϵ : very small number

참고문헌

- Niwa, T.: *Chem. Eng.*, **June**, 103(1993).
- Niwa, T.: Proc. 5th Intl. Symp. on PSE, Kyungju, Korea, 497(1994).
- Pantelides, C. C., Realf, M. J. and Shah, N.: *Chem. Eng. Res. Des.*, **73**, 431(1995).

4. Realff, M. J., Shah, N. and Pantelides, C. C.: *Comput. Chem. Eng.*, **20**, 869(1996).
5. Gonzalez, R. and Realff, M. J.: *Comput. Chem. Eng.*, **22**, 841(1998).
6. Gonzalez, R. and Realff, M. J.: *Comput. Chem. Eng.*, **22**, 857(1998).
7. Lee, H.-K., Jung, J. H. and Lee, I.-B.: *Chemical Industry and Technology*, **13**, 272(1995).
8. Yoo, D. J., Lee, H.-K., Jung, J. H. and Lee, I.-B.: *HWAHAK KONG-HAK*, **34**, 742(1996).
9. Pinto, J. M. and Grossmann, I. E.: *Ind. Eng. Chem. Res.*, **34**, 3037 (1995).
10. Bok, J.-K. and Park, S.: *Ind. Eng. Chem. Res.*, **37**, 3652(1998).
11. Pinto, J. M.: Ph. D. Dissertation, Carnegie Mellon University, PA, USA(1995).
12. Sahinidis, N. V. and Grossmann, I. E.: *Computers Chem. Engng.*, **15**, 85 (1991).
13. Jung, J. H.: Ph. D. Dissertation, POSTECH, Pohang, Korea(1993).
14. Byun, S. H., Lee, H. P., Kim, K. S. and Yoo, K. O.: *HWAHAK KONG-HAK*, **35**, 877(1997).
15. Gooding, W. B. and Pekny, J. F.: *Computers Chem. Engng.*, **18**, 909 (1994).
16. Brooke, A., Kendrix, D. and Meeraus, A.: In GAMS: Users Manual, Scientific Press, Redwood City, CA(1992).
17. IBM: OSL(Optimization Subroutine Library) Guide and reference, release 2; Kingston, NY(1991).