

AND/OR 그래프를 이용한 공정계획과 일정계획의 통합에 관한 연구

A Study on Integration of Process Planning and Scheduling Using AND/OR Graph

김기동* · 정한일** · 정대영*** · 박진우***

Kidong Kim* · Hanil Jeong** · Daeyoung Chung*** · Jinwoo Park***

Abstract

Traditionally, the Process Planning problems and the Scheduling problems have been considered as independent ones. However, we can take much advantages by solving the two problems simultaneously. In this paper, we deal with the enlarged problem that takes into account both the process planning and the scheduling problems. And we present a solution algorithm for the problem assuming that the given process plan data is represented by AND/OR graph. A mathematical model(mixed ILP model) whose objective is the minimization of the makespan, is formulated. We found that we can get the optimal solutions of the small-size problems within reasonable time limits, but not the large-size problems. So we devised an algorithm based on the decomposition strategy to solve the large-scale problems (realistic problems) within practical time limits.

Keywords: Process Planning, Scheduling, AND/OR graph

* 고등기술연구원

** 대전대학교 산업공학과

*** 서울대학교 산업공학과

1. 서론

오늘날 기업은 고품질, 납기 단축, 상품의 다양화 및 제품 수명 주기의 단축 등 시장 변화에 대응하기 위해 여러 가지 방법을 시도하고 있다. 이러한 접근법 중 하나는 CIM (Computer Integrated Manufacturing : 컴퓨터 통합 생산)과 같은 제조 철학에서 볼 수 있는 생산 시스템의 통합이라 할 수 있다. 일반적인 제조기업에서의 제품 제조 과정을 살펴보면, 설계 부서에서는 도면을 작성하고, 공정계획 부서에서는 이러한 설계 자료와 공장의 상황을 고려하여 작업 지시서를 작성하고, 생산 관리 부서에서는 시간적인 측면과 자사의 보유 자원을 고려하여 경제적인 생산을 위한 생산계획을 수립하고, 이에 의해 생산 지시를 내리며 작업장을 통제한다. 이러한 일련의 과정은 크게 두 개의 영역으로 구분할 수 있는데 하나는 설계, 공정계획, 가공이 속하는 생산기술 영역이고 다른 하나는 생산을 경제적으로 수행할 수 있도록 통제하는 생산관리의 영역이다. 상기의 두 영역은 전통적으로 분리되어 수행되어져 왔지만 최근에는, 효율적인 생산 시스템을 이루기 위해 이들에 대한 통합의 노력이 많이 시도되고 있다[6, 8, 18].

생산 기술 영역과 생산 관리 영역의 통합을 위한 다양한 방법들 중 한정된 생산 설비의 효율적인 운용을 위해서는, 부품별 공정계획의 선택과 가공 기계를 결정하는 공정계획 문제와 각 기계에서의 가공 순서를 결정하는 일정계획 문제를 함께 고려하는 것이 가장 효율적이라고 볼 수 있다. 이 두 기능은 모두 공장내의 자원 할당에 관련된 문제

를 다루고 있으므로 어느 한 모듈에서의 결과가 다른 모듈에 결정적인 영향을 미치고 있다. 그러나 실제로는 이 두 기능은 각각 독립적이고 순차적으로 행해지고 있는 것이 현실이다. 공정계획자가 현장 상황을 고려하지 않고 최적이라고 제시한 공정계획이 일정계획자의 입장에서는 실행 불가능한 공정 계획이 될 수도 있다[13]. 공정 선택과 공장 기계의 선택에서는 기술적인 요구 사항만을 우선으로 고려하며, 일정계획 단계에서는 공정계획은 이미 주어졌다고 가정하고 있다. 공정계획 수립자는 공장 상황을 고려하지 않고 있으며, 일정계획자는 대체 공정 사용의 기회를 제공받지 못한다. 결과적으로 부품의 인도기간이 길어지고, 가공비용의 증가를 초래한다[8, 17]. 공정계획이란 부품에 대한 기하학적인 형상, 원자재, 공차, 표면 거칠기, GT 코드 등의 부품 관련 정보와 공장내 기계들의 능력 및 현재의 기계에 걸린 부하에 대한 정보를 이용하여, 요구된 품질의 부품을 생산하기 위한 구체적인 생산 지시서를 작성하는 것이다. 공정계획은 그 복잡한 기능상 특징으로 인하여 이의 결과가 전체적인 시스템에 미치는 영향을 평가하기는 쉽지 않다. 초기의 연구에서는 주로 공정계획과 CAD와의 연계 관계에 대해서만 고려하였으며, 그 결과 현 시점에서 구입 가능한 대부분의 CAPP 시스템은 생산기술 지향적인 시스템이다[7]. 그러나 최근 유럽을 중심으로 생산 관리 영역과의 연계에 대한 연구가 활발해지고 있다. 특히 CAPP에서 수행한 공정 선택의 결과는 향후 일정계획의 성능에 지대한 영향을 미치게 되며 따라서 공정의 선택에 대한 연구는 생산 관리 영역과의 연계에 있어서 가장 중

요한 연구라 할 수 있다[13].

본 논문에서는 대체 공정(alternative routing)에 대한 정보가 주어졌을 때, 이를 표현하는 방법론과 함께 이를 이용하여 대체 공정 중 하나를 선택하고 해당 공정을 가공할 기계를 결정하고 일정계획을 수립하는 방안에 대해 다루어 보고자 한다. 이하 제 2절에서는 관련 연구현황, 제 3절에서는 통합 모형이 제시된다. 제 4절에서는 모형의 해법, 제 5절에서는 실험 결과가, 제 6절에서는 결론이 제시된다.

2. 관련 연구 현황

공정 계획의 표현법에 대한 중요성의 인식은 최근의 일로서, 지금 발표되고 있는 대부분의 모델이 현재 개발 중이거나 실험단계의 수준이다. 그 중 대표적인 것으로 크게 세 가지를 들 수 있는데, 그 중 하나는 STEP의 파트 49에서 정의되어 있는 공정계획 표현안이다. 여기에서는 산업의 종류에 무관하게 적용될 수 있는 가장 일반적인 표현법을 제안하고 있으며 그 기본적인 아이디어는 공정의 분할에 의거하는 것이다. 여기서 제시한 표현법은 개념적인 모델이라 할 수 있으며 이러한 공정계획을 이용하여 직접적인 가공 명령이 이루어지지는 않는다. 한편 STEP에서 공정 계획의 표현모델에 대한 논의가 이루어질 때와 거의 비슷한 시기에 NIST(National Institute of Standards and Technology)에서는 국제 표준과는 별도로 ALPS(A Language for Process Specification)라는 실험적인 공정계획 언어를 만들었다. STEP과는 달리 이 언어는 실행을 염두에 두고 만들었으며, 그래프 구

조를 이용하여 공정계획을 작성하게 되어있다. 공정계획 표현법중 나머지 하나는 Mettala [12]가 제시한 AND/OR 그래프이다. 이 방법은 DeMello와 Sanderson[4]이 조립 계획에 대해 사용했던 AND/OR 그래프에서 아이디어를 얻어서 제시한 방법으로 노드는 가공해야 할 공정을 의미하고, 아크는 선행관계를 의미한다. STEP에서 제정한 공정계획 표현안은 일반적인 표현을 위해서 노력했지만 계층구조 자체가 가지는 한계로 인해 표현할 수 없는 관계가 존재할 수 있으며, ALPS에서도 표현할 수 없는 관계가 존재한다. 한편 AND/OR 그래프에서는 분명한 선후행 관계를 표현할 수는 있지만 동시가공과 같은 특정한 관계가 있는 경우에는 표현할 수 없는 한계가 있다.

공정계획과 일정계획을 연계하여 계획을 수립하기 위한 구체적인 결정 사항은 다음과 같다.

- 1) 대체 공정 중 사용할 공정의 선택
- 2) 대체 기계 중 가공할 기계의 선택
- 3) 기계에서의 가공 순서의 결정

공정계획과 일정계획의 연계에 관한 문제는, 대체 공정과 각 공정을 가공할 수 있는 대체 기계가 존재할 때의 일정계획 문제와 동일하다. 대체 기계를 고려한 일정계획과 관련된 연구는 다수 있지만 대체 공정과 대체 기계를 모두 고려한 일정계획과 관련된 연구는 드물다[2]. Zijm[18]은 AND/OR 그래프로 표현된 대체 공정계획에서, 공장의 부하를 고려하지 않고 가장 빨리 가공을 완료할 수 있는 공정과 기계를 선택하는 문제를 다루었다. Brandimarte[2]는 잡샵(job shop)에서의 부하할당과 일정계획을 동시에 고려하고 있다. 대체 공정은 매크로 대체공정과 상세 대체공정

을 이용해서 표현하였고, 공정계획과 일정계획으로 문제를 분할하여 계층적으로 풀이하였다. Brandimarte[2]의 연구에서는 AND/OR 그래프로부터 옵션을 생성하는 과정을 단순화시켰으며, 부하할당 문제와 일정계획 문제로 문제를 분할하기는 하였지만 각각의 문제가 NP hard 인 관계로 문제의 풀이에 소요되는 시간이 길다.

3. 공정계획과 일정계획의 통합 모형

본 연구에서 다루는 공정계획/일정계획의 통합 시스템이 이용하는 중요한 데이터로는 기계, 주문, 부품의 공정계획이 있다. 현장내의 실시간 데이터의 취합이 가능하다고 할 때, 현장의 원활한 통제를 위하여 기계에 관련된 데이터로 최소한 갖추어져야 하는 내용은 예약된 부하를 표현하기 위해 {machine id, order#, part#, operation#, start time, stan-

dard time}의 내용을 포함해야 하며, 기계의 작동 상태를 파악하기 위해 {machine id, machine condition, 고장발생 시각, 예상수선 시간}의 내용이 포함되어야 한다. 고객의 주문에 관해서는 {order #, part #, due date, quantity}의 내용이 포함된다. 공정계획의 표현은 기본적으로 AND/OR 그래프를 이용한다. 다음의 그림 1에 본 연구에서 사용하는 AND/OR 그래프의 예를 도시했다. 본 연구에서 사용되는 노드의 종류는 7가지가 있다. AS_{ij}는 부품 i의 j번 AND SPLIT를 의미하며, AJ_{ij}는 부품 i의 j번 AND JOIN을 의미한다. OS_{ij}는 부품 i의 j번 OR SPLIT를 의미하며, OJ_{ij}는 부품 i의 j번 OR JOIN을 의미한다. S, T는 더미(dummy)노드로 그래프의 시작과 끝을 각각 의미한다. 그 외의 노드ij는 모두 1회 setup으로 가공 가능한, 부품 i의 j번 공정을 의미하며 각 노드에는 해당 공정을 수행할 수 있는 기계와 표준 가공 시간

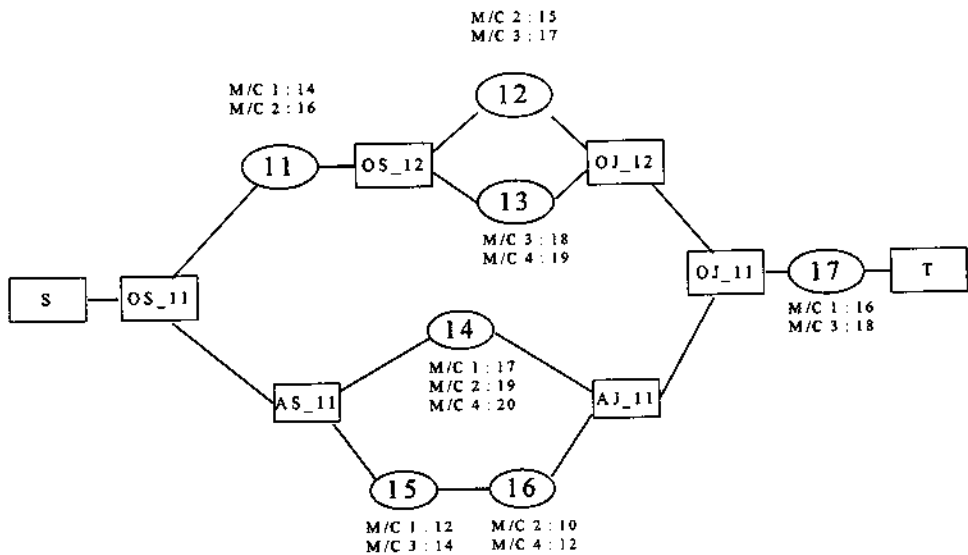


그림 1. 예제 부품 1의 AND/OR 그래프

이 명시되어 있다.

가공해야 할 부품의 대체 공정은 AND/OR 그래프로 표현되고, 이러한 부품을 여러 개 가공해야 할 경우 이들 각 부품에 대한 공정 선택, 기계 선택, 일정 계획을 통합해서 풀 수 있는 수리 모형을 다음에 제시했다. 다음의 [문제 1]은 공장내 기계들에 기존에 예약된 부하가 존재할 경우, MRP와 같은 상위 생산 계획에서 결정된 주문량을 최단시간에 가공할 수 있는 혼합 정수 계획법이다.

모형에 사용된 인덱스

i, l : 부품(job) 인덱스

j, m : 공정(operation, setup)과 OR 노드 인덱스

p, q : OR 노드에서의 option 인덱스

k : 기계 인덱스

모형에 사용된 상수 및 집합

M : 큰 정수 M

P_{ik} : 부품 i 의 j 공정을 k 기계에서 가공할 때의 가공 시간

OR_i : 부품 i 의 OR 노드 인덱스 집합

OP_i : 부품 i 의 공정 노드 인덱스 집합

SN_{jp} : 부품 i 의 j OR 노드 p 번째 옵션에 속한 OR 노드 인덱스 집합, p 번 옵션에 내포된 OR 노드에 다시 내포된 OR 노드 인덱스는 제외

SO_{jp} : 부품 i 의 j OR 노드 p 번째 옵션에 속한 공정 노드 인덱스 집합, p 번 옵션에 내포된 OR 노드에 다시 내포된 공정 노드 인덱스는 제외

PR_j : 부품 i 의 노드 중 j 노드 이후에 수행되어야 하는 공정 노드 인덱스 집합

MJ_k : k 기계를 사용 가능한 모든 공정 노드 인덱스 집합

LS_{lm} : k 기계에서 기존에 할당된 m 번째 공정의 시작 시간

LP_{lm} : k 기계에서 기존에 할당된 m 번째 공정의 가공 시간

결정변수

Z_{jp} : 부품 i 의 AND/OR 그래프에서 j 번 OR 노드의 p 번 옵션이 선택되면 1 아니면 0

Y_{ijk} : 부품 i 의 j 공정을 k 기계에서 수행하면 1 아니면 0

X_{ij} : 부품 i 의 j 공정의 가공 시작 시간 (j 가 OR 노드가 아닌 경우)

W_{jm} : 부품 i 의 j 공정이 부품 i 의 m 공정을 선행하면 1 아니면 0

V_{ijmk} : 부품 i 의 j 공정이 부품 i 의 m 공정을 기계 k 에서 선행하면 1 아니면 0

L_{ijmk} : 부품 i 의 j 공정이 기존에 할당된 m 공정을 선행하면 1 아니면 0

MS : 최종 가공 완료 시간

[문제 1]: 공정계획과 일정계획을 위한 0-1 혼합 정수 계획법

$$\text{Min } MS \quad (1)$$

subject to

$$\sum_p Z_{ijp} = 1 \text{ 모든 } i, \text{ 모든 } j, \text{ 단 } j \in OR_i,$$

$$j \notin \bigcup_{m,q} SN_{imq} \quad (2)$$

$$\sum_p Z_{ijp} = Z_{imq} \text{ 모든 } i, \text{ 모든 } j, \text{ 단 } j \in OR_i,$$

$$j \in SN_{imq} \quad (3)$$

$$\sum_k Y_{ijk} = 1 \text{ 모든 } i, \text{ 모든 } j, \text{ 단 } j \in OP_i,$$

$$j \notin \bigcup_{m,p} SO_{imp} \quad (4)$$

$$\sum_k Y_{ijk} = Z_{imp} \quad \text{모든 } i, \text{ 모든 } j, \text{ 단 } j \in OP_i, \\ j \in SO_{imp} \quad (5)$$

$$X_j + \sum_k P_{ijk} Y_{ijk} \leq X_{im} \quad \text{모든 } i, \text{ 모든 } j, \\ \text{모든 } m, \text{ 단 } m \in PR_{ij} \quad (6)$$

$$X_{ij} + \sum_k P_{ijk} Y_{ijk} \leq X_{im} + M(1 - W_{ijm}) \quad \text{모든 } i, \\ \text{모든 } j, \text{ 모든 } m, \text{ 단 } m \notin PR_{ij} \quad (7)$$

$$X_{im} + \sum_k P_{imk} Y_{imk} \leq X_{ij} + MW_{ijm} \quad \text{모든 } i, \\ \text{모든 } j, \text{ 모든 } m, \text{ 단 } m \notin PR_{ij} \quad (8)$$

$$X_{ij} + P_{ijk} \leq X_{im} + M(3 - Y_{ijk} - Y_{imk} - V_{ijlmk}) \\ \text{모든 } k, \text{ 모든 } i, \text{ 모든 } j, \text{ 모든 } l, \\ \text{모든 } m, \text{ 단 } j \in MJ_k, m \in MJ_k, i \neq l \quad (9)$$

$$X_{im} + P_{imk} \leq X_{ij} + M(2 - Y_{ijk} - Y_{imk} + V_{ijlmk}) \\ \text{모든 } k, \text{ 모든 } i, \text{ 모든 } j, \text{ 모든 } l, \\ \text{모든 } m, \text{ 단 } j \in MJ_k, m \in MJ_k, i \neq l \quad (10)$$

$$X_{ij} + P_{ijk} \leq LS_{km} + M(2 - Y_{ijk} - L_{ijmk}) \\ \text{모든 } k, \text{ 모든 } i, \text{ 모든 } j, \text{ 모든 } m, \\ \text{단 } j \in MJ_k \quad (11)$$

$$LS_{km} + LP_{km} \leq X_{ij} + M(1 - Y_{ijk} + L_{ijmk}) \\ \text{모든 } k, \text{ 모든 } i, \text{ 모든 } j, \text{ 모든 } m, \\ \text{단 } j \in MJ_k \quad (12)$$

$$X_{ij} \geq 0 \quad \text{모든 } i, j \text{는 START 노드} \\ \text{인덱스} \quad (13)$$

$$X_{ij} + \sum_k P_{ijk} Y_{ijk} \leq MS \\ \text{모든 } i, j \text{는 END 노드 인덱스} \quad (14)$$

$$L_{ijmk}, V_{ijlmk}, W_{ijm}, Y_{ijk}, Z_{ijm} \in (0, 1) \quad (15)$$

식 (2)와 (3)은 OR 노드에서의 공정 선택

에 관한 제약이고, 식 (4)와 (5)는 선택된 옵션에 속한 공정을 수행할 기계 선택에 관한 제약이다. 식 (6)은 그래프 상에서 선행 관계가 있는 작업들 간의 수행 순서에 관한 제약이며, 식 (7), (8)은 선행관계가 없는 공정들이 동시에 가공되는 것을 방지하는 제약이다. 식 (9)와 (10)은 한 기계에서 서로 다른 작업이 동시에 수행되는 것을 방지하고, 식 (11)과 (12)는 기계에 미리 할당이 예약된 공정과 동시에 가공되는 것을 제약하며, 이 식에서 현재의 공장 상황이 반영된다. 식 (13)은 변수에 대한 초기조건을, 식 (14)는 목적함수의 식 (1)과 함께 최종 가공 완료시각의 최소화에 관한 식이다. [문제 1]은 제약식 (2), (3), (4), (5)가 없는 경우 잡샵(job shop) 일정 계획 문제와 동일하게 되며 잡 일정계획 문제는 NP-hard로 알려져 있다. 따라서 [문제 1]도 NP-hard이다.

4. 그래프 이론을 이용한 해법

본 연구에서는 전체 공정계획과 일정계획을 각 부품별로 나누어서 수립한다. 모든 부품을 동시에 고려하지 않고 우선 순위가 높은 부품부터 하나씩 순차적으로 고려한다. 하나의 부품만을 고려할 경우 [문제 1]의 모든 제약식은 한 부품에 대해서만 형성되며, 제약식 (9), (10)이 사라진다. 각 부품별로 공정 선택과 일정선택을 할 경우, 주어진 AND/OR 그래프에 기존의 그래프 이론을 확장, 적용함으로써 공정과 기계를 선택하고 일정을 수립하는 방안을 제시한다.

부품을 개별적으로 다루는 경우, 공장의 상황과 부품의 공정 구성에 따라 세 가지 유형

으로 구분할 수 있다. 그 중 하나는 기존의 부하가 없는 상황이고, 두 번째는 기존의 부하가 있는데 주어진 부품의 공정정보에 AND SPLIT이 없는 경우, 마지막으로 기존의 부하가 있는데 주어진 부품의 공정정보에 AND SPLIT이 있는 경우이다. 알고리즘의 설명에서 이용되는 모든 시간 단위는 분이다.

4.1 기존 부하가 없는 상황

기존의 부하가 없는 상황에서는 주어진 AND/OR 그래프를 변형시킨 그래프를 이용하여 문제를 풀이한다. 이를 위해 다음을 정의한다.

“병렬 아크를 가진 AND/OR 그래프”라고 정의한다.

정의 2 변형(modified) AND/OR 그래프

병렬 아크를 가진 AND/OR 그래프에서 AND SPLIT에 속한 노드를 모두 직렬로 변환한 그래프를 “변형 AND/OR 그래프”라고 정의한다. AND SPLIT에 속한 공정 노드들 간의 순서는 그래프에 명시되어있는 선후행 관계를 지킬 수 있는 구조면 모두 가능하다.

위의 그림 1에 도시된 예제 부품 1의 그래프를 변형 AND/OR 그래프로 변환하면 다음의 그림 2와 같이 된다. 이와 같이 정의된 그래프를 이용하면 최단 경로를 구하는 알고리

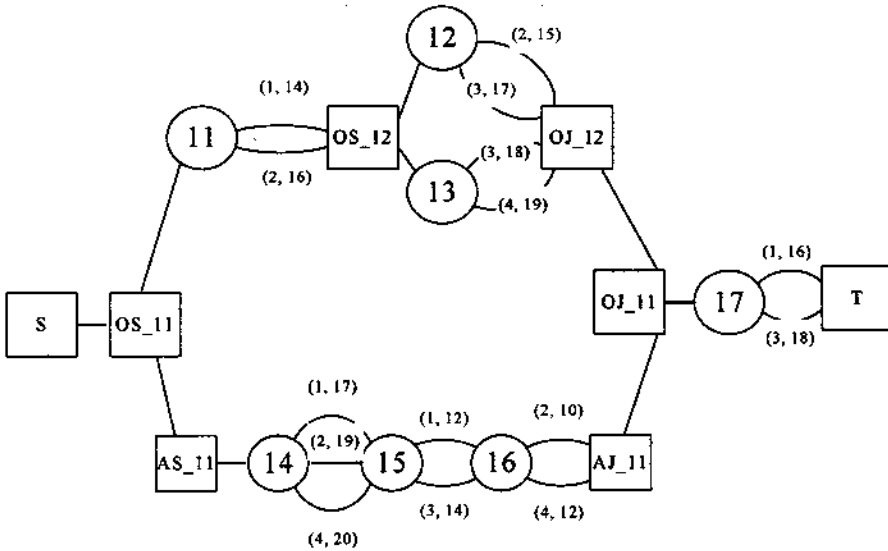


그림 2. 예제 부품 1의 변형 AND/OR 그래프

정의 1 병렬 아크(arc)를 가진 AND/OR 그래프

주어진 AND/OR 그래프에서 공정 노드 i에서의 기계와 가공시간에 대한 정보를 노드 i의 후행 노드까지의 아크로 바꾼 그래프를

즘에 의해 부하가 없는 상황에서의 최적해를 구할 수 있다.

그림 3에 최단 경로를 구하는 과정을 도시했다. 각 공정 노드의 아크에 (*)표시는 풀이 과정에서 결정된 최소값들이다.

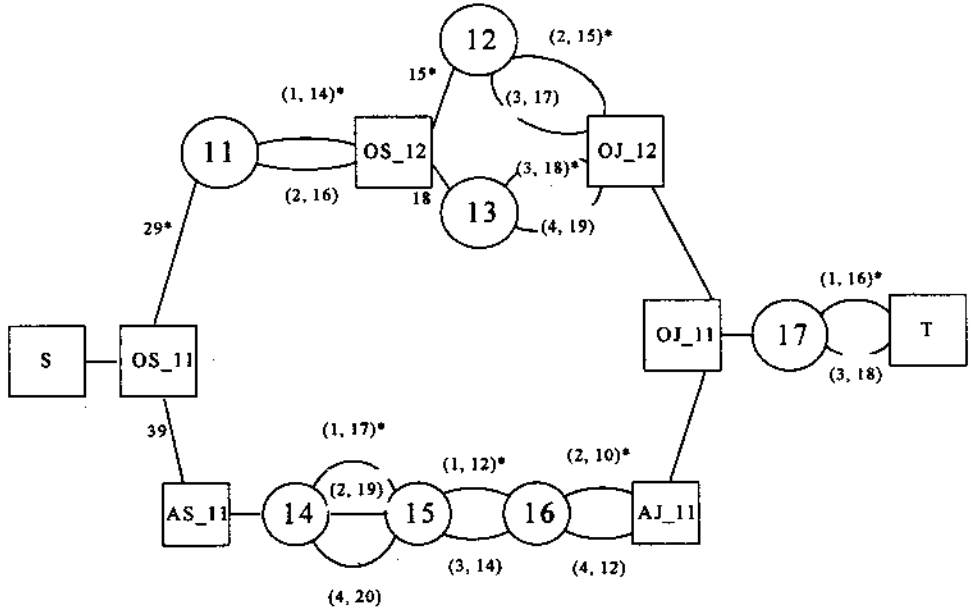


그림 3. 부품 1의 변형 AND/OR 그래프에서 최단 경로 탐색 과정

이 문제의 최적해는 공정 1은 1번 기계, 공정 3은 2번 기계, 공정 9는 1번 기계에서 수행한다. 이 때 총 가공 완료시간 45분이다.

4.2 기존 부하가 있고 부품에 AND SPLIT 이 없는 경우

현재 공장의 부하가 지금 처리해야 할 부품의 우선 순위보다 높은 것이 존재할 때는 공장의 기존 부하를 무시할 수가 없다. 이 때가 기존의 부하가 있는 경우가 된다. 이 경우에는 투입되는, 새로 가공해야 할 부품의 공정 구조에 따라 문제의 난이도가 달라진다. 공정을 나타내는 AND/OR 그래프에 AND SPLIT이 없는 경우에는 있는 경우보다 쉽게 문제를 해결할 수 있다.

Halpern과 Priess[5], Sancho[15, 16] 등은 시간 제약이 있는 유방향 그래프에서의 최단 경로에 대한 연구를 수행했다. 이들이 다루

고 있는 문제는 노드와 아크에 통과가 가능한 시간대가 따로 존재하고 있는 경우에 그래프에서의 최단 경로에 관한 문제이다. Halpern과 Priess[5]는 노드와 아크에 시간창이 존재하고 노드에서는 정박(parking)과 통과(bypass)를 허용하며 출발시간은 정해져 있지 않은 문제를 다루었고, Sancho[15]는 Halpern의 문제와 동일한 문제를 다루었는데 Halpern의 경우에는 해를 하나만 제공하는 반면, 여기서는 대체 해를 제공한다. 또한 해를 구하는 알고리즘의 복잡도(complexity) 측면에서는 Halpern의 경우가 해를 구하기 더 쉽다. Sancho[16]는 한 노드에서 그 원도우가 열리기 이전에 도착한 것에 대해서는 그 노드에서 대기하고, 노드의 통과를 허용하지 않는 경우에 동적 계획법(dynamic programming)으로 해를 구하는 방법에 대해 발표했다. 이러한 연구 결과를 바탕으로 기존 부하가 있

고 부품에 AND SPLIT이 없는 경우에 작동할 수 있는 해법을 동적 계획법으로 개발한다.

부품 i 의 공정을 표현한 AND/OR 그래프에 공장의 부하 상태를 참조하여 만든 그래프를 시간창을 가진 AND/OR 그래프를 정의한다.

정의 3 시간창(time window)을 가진 AND/OR 그래프

정의 1에서 정의한 병렬 아크(arc)를 가진 AND/OR 그래프의 노드 i 에서 노드 j 로 가는 k 의 아크에 기계 k 가 가용한 시간창을 기록한 그래프를 시간창을 가진 AND/OR 그래프로 정의한다. 시간창은 (a_{ijk}^t, b_{ijk}^t) 로 표현되는데 의미는 노드 i 에서 노드 j 로 가는 아크 k 의 통과 가능한 t 번째 시간창이 열리는 시간과 닫히는 시간이다.

4.1절의 예제를 풀이한 결과로 할당된 각 기계의 부하는 기계 1에 $\{0, 14\}$, $\{29, 45\}$ 이고 지금 시점에 기계 4가 고장이 발생해 20분 후에 고쳐진다고 가정하면, 부하와 고장에 의한 기계의 시간창은 다음 표 1과 같다.

표 1. 부품 1 투입 후 기계의 부하와 고장으로 인한 시간창

기계	예약된 기간과 고장 기간	시간창
1	$\{0, 14\}$, $\{29, 45\}$	$\{14, 29\}$, $\{45, \infty\}$
2	$\{14, 29\}$	$\{0, 14\}$, $\{29, \infty\}$
3	$\{ \}$	$\{0, \infty\}$
4	$\{0, 20\}$	$\{20, \infty\}$

다음 그림 4의 예제 부품 2의 AND/OR 그래프에 표 1의 시간창을 가진 AND/OR 그래

프를 그림 5에 나타냈다. 기존 부하가 있고 부품에 AND SPLIT이 있는 경우 다음의 정리 1이 성립된다.

정리 1

기존 부하가 있고 AND SPLIT이 없는 경우의 최소 가공 완료 시간을 구하는 문제는 아크에 시간창을 가진 AND/OR 그래프에서 노드에서의 파킹(parking)을 허용한 상황에서 S 노드에서 T 노드까지의 최단 경로를 구하는 문제이다.

증명

기존의 부하가 있고 AND SPLIT이 없는 경우는, 이 문제에 대한 AND/OR 그래프에서 각 아크를 통과할 수 있는 시간대가 지정되어 있는 경우의 최단 경로를 구하는 문제가 된다. 노드 i 에서 노드 j 로 아크 k 를 통한 이동은 공정 i 를 기계 k 를 이용해서 가공하는 경우가 된다. 기계 k 가 유용한 시간대에만 가공을 할 수 있으며, 이는 아크에 통과 가능한 시간대가 존재한다는 의미이다. 그래프에 AND SPLIT이 없으므로 노드 S에서 노드 T로의 최단 경로가 주어진 부품을 가공할 수 있는 최소의 가공 완료시간이다. ■

노드 i 에서 노드 j 까지 아크 k 를 통하여 이동하는 경우 이동시간은 P_{ijk} 로 주어진다. f_i 를 노드 S에서 노드 i 까지의 최단 경로라고 정의한다. 아크 k 의 t 번째 시간창이 (a_{ijk}^t, b_{ijk}^t) 로 주어졌을 때, 이 시간창을 지날 수 있는 최초의 시각은 f_i 와 a_{ijk}^t 중 큰 값으로 결정된다. 따라서 이 시간창을 이용하려면 b_{ijk}^t 와 최초 시각의 차이가 P_{ijk} 보다 크거나 같아야 가능하다. 즉 시간창 k 를 이용하기 위해서는 다음의 조건을 만족해야 한다.

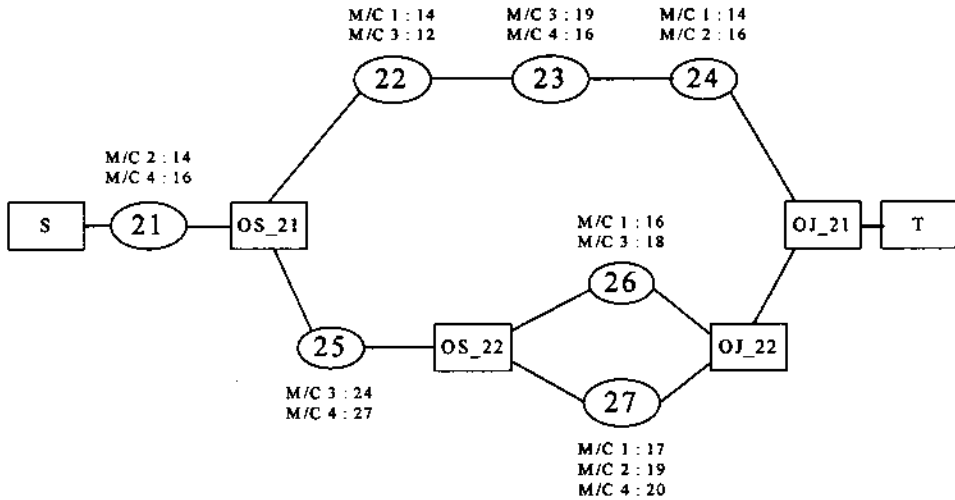


그림 4. 예제 부품 2의 AND/OR 그래프

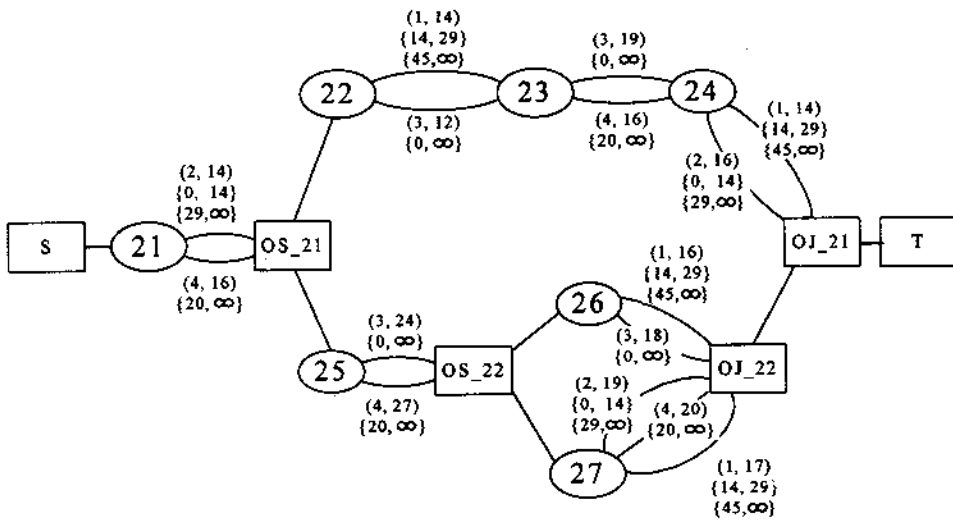


그림 5. 예제 부품 2의 시간창을 가진 AND/OR 그래프

$b_{ijk}^t - \max\{a_{ijk}^t, f_i\} \geq P_{ijk}$ (16) 동했을 경우 노드 j에 이르는 시간은 다음의 식 (17)로 주어진다.

시간창(a_{ijk}^t, b_{ijk}^t)을 이용해서 노드 j 로 이

$$\max\{a_{ijk}^t, f_i\} + P_{ijk} \quad (17)$$

노드 j의 선행 노드가 하나밖에 없는 경우에는 각 아크에서 위의 식 (17)의 값 중 최소 값을 주는 시간차를 찾고, 이 값이 최소가 되는 아크를 찾아서 이동하면 f_j 를 구할 수 있다. 한편 노드 j의 선행 노드가 둘 이상일 때는 식 (17)의 값을 최소로 하는 선행 노드에서 가는 경로를 선택한다. 따라서 다음의 회귀식(recursive equation) (18)을 이용하여 최단 경로를 구할 수 있다.

$$f_j = \min_{i,k,t} \{ \max\{a_{ijk}^t, f_i\} + P_{ijk} \mid i \text{는 } j \text{의 선행 노드, } b_{ijk}^t - \max\{a_{ijk}^t, f_i\} \geq P_{ijk} \} \quad (18)$$

$$f_s = 0$$

위의 식 (18)은 동적 계획법(dynamic programming)의 일종인 Dijkstra 류의 라벨 셋팅(label setting) 알고리즘으로 풀이할 수 있으며, 노드의 수를 n이라고 할 때 복잡도는 $O(n^2)$ 이다[1]. 위의 회귀식을 이용하여 예제 부품 2에 대해 풀이한 과정을 다음의 표 2에 나타내었다. 다음의 식 (19)에서 T_{ijk} 을 정의한다.

$$T_{ijk} = \min_t \{ \max\{a_{ijk}^t, f_i\} + P_{ijk} \mid i \text{는 } j \text{의 선행 노드, } b_{ijk}^t - \max\{a_{ijk}^t, f_i\} \geq P_{ijk} \} \quad (19)$$

표 2의 비교란에 표현되어 있는 \checkmark 의 의미는 하나만이 있는 경우는 풀이 과정에서 f_i 를 산출한 아크를, 두 개가 있는 경우는 최적해를 의미한다. 예제 2의 부품을 풀이한 결

표 2. 예제 부품 2에 대해 풀이한 과정

j	i(선행노드)	k(호)	T_{ijk}	비고	f_j
S2	null	null	0		0
21	S2	-	0	$\checkmark\checkmark$	0
OS_21	21	2	14	$\checkmark\checkmark$	14
		4	16		
22	OS_21	-	14	\checkmark	14
23	22	1	28		26
		3	26	\checkmark	
24	23	3	45		42
		4	42	\checkmark	
25	OS_21	-	14	$\checkmark\checkmark$	14
OS_22	25	3	38	$\checkmark\checkmark$	38
		4	47		
26	OS_22	-	38	$\checkmark\checkmark$	38
27	OS_22	-	38	\checkmark	38
OJ_22	26	1	61		56
		3	56	$\checkmark\checkmark$	
	27	1	62		
		2	57		
OJ_21	24	1	59		56
		2	58		
	OJ_22	-	56	$\checkmark\checkmark$	
T2	OJ_21	-	56	$\checkmark\checkmark$	56

과 선택된 공정은 1, 5, 6이다. 공정 1은 2번 기계에서 {0, 14} 동안 가공, 공정 5는 기계 3에서 {14, 38} 동안 가공, 공정 6은 기계 3에서 {38, 56} 동안 가공한다. 모든 시간 단위는 분이다.

4.3 기존 부하가 있고 부품에 AND SPLIT이 있는 경우

기존 부하가 있고 공정을 나타내는 AND/OR 그래프에 AND SPLIT이 있는 경우에는 AND

SPLIT이 없는 경우보다 풀이가 어렵다. 정의 3에 정의된 시간창을 가진 AND/OR 그래프를 이용하여 문제를 풀이한다. 예제 부품 3이 새로이 투입된다고 가정하자. 예제 부품 3의 AND/OR 그래프는 다음 그림 6과 같다. 지금 공장의 상황은 예제 부품 1, 예제 부품 2 그리고 기계 고장으로 인해 표 3과 같은 시간창을 갖게된다.

표 3. 부품 1, 부품 2 투입 후 기계의 부하와 고장으로 인한 시간창

기계	예약된 기간과 고장 기간	시간창
1	{0, 14}, {29, 45}	{14, 29}, {45, ∞}
2	{0, 29}	{29, ∞}
3	{14, 56}	{0, 14}, {56, ∞}
4	{0, 20}	{20, ∞}

곧 바로 적용하기에는 무리가 따른다. 그 이유는 노드를 따라가면서 f_i 값을 계산하다가 AND SPLIT i 를 만날 경우에는 AND JOIN i 를 만날 때까지의 경로를 찾는 것으로 문제가 풀리지 않는다. 기본적으로 AND SPLIT과 AND JOIN 사이의 노드는 그것이 OR SPLIT에서 OR JOIN에 내포되지 않았으면 모든 노드를 거쳐야 한다. 위의 예제 부품 3을 보면 AS_31과 AJ_31 사이의 노드 5, 6, 7 그리고 8은 모두 거쳐야 한다. 이 경우에는 다음의 정리 2가 성립한다.

정리 2

시간창을 가진 AND/OR 그래프가 주어졌을 때, AND SPLIT에서 AND JOIN까지 가장 빠른 노드 순서의 결정은 NP-hard 문제이다.

증명

이 문제에서 모든 시간창이 $\{0, \infty\}$ 로 주어지고, 노드와 노드를 연결하는 아크들 중

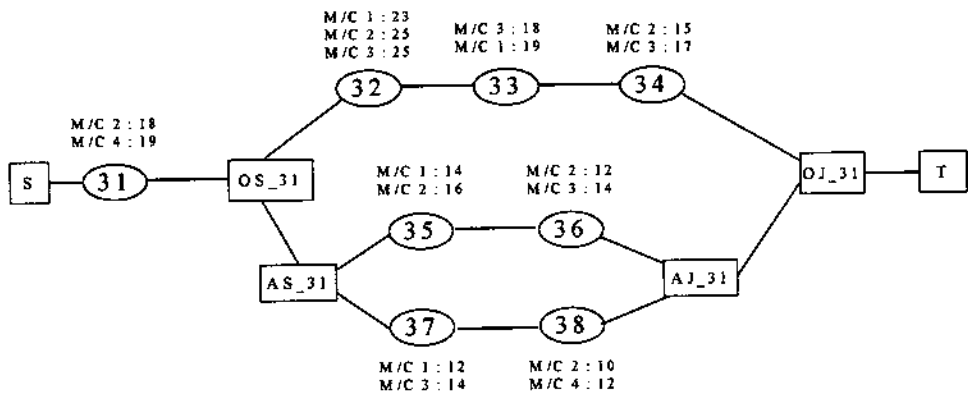


그림 6. 예제 부품 3의 AND/OR 그래프

예제 부품 3의 시간창을 가진 AND/OR 그래프는 다음 그림 7과 같다. 이 문제를 풀이하기 위해 4.2절에서 논의된 동적 계획법을

가장 짧은 거리를 갖는 아크만을 남겨두면 NP-hard로 알려진 최단 경로의 Hamiltonian path를 찾는 문제가 된다. 즉 NP-hard인 문제

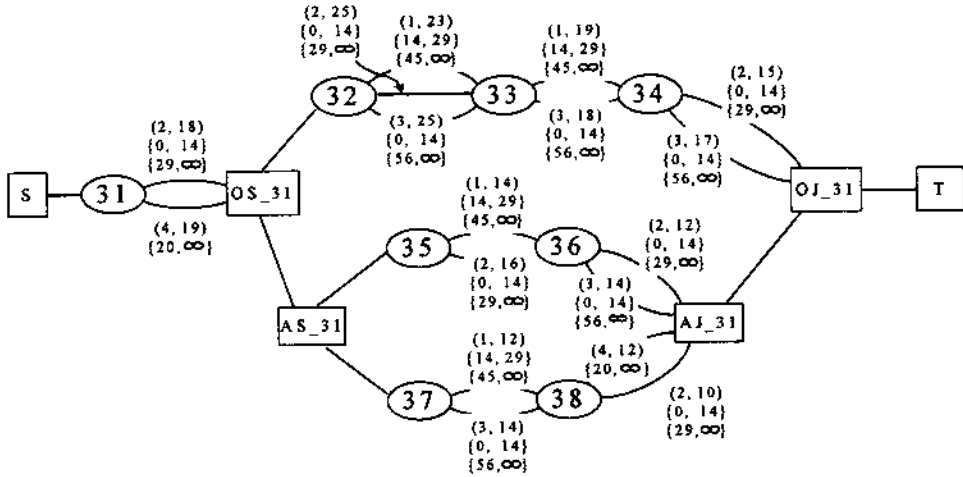


그림 7. 예제 부품 3의 시간창을 가진 AND/OR 그래프

가 이 문제의 특수한 형태로 바뀔 수 있다. 따라서 이 문제는 NP-hard이다.■

이 경우에는 AND SPLIT과 AND JOIN사이의 노드들에 대해서는 [문제 1]의 정수 계획법을 풀어야 한다. 이때 기업에 따라 가공해야 할 부품의 수나 종류가 다르겠지만, T 중공업의 경우 모든 부품의 공정 수가 25번을 넘지 않는 점, 일반적으로 논문에서 소개하는 예제들에 대한 부품에서도 공정의 수가 10번 내외인 점과 한 부품의 모든 공정을 고려하는 것이 아니라 AND SPLIT과 AND JOIN 사이의 공정만을 고려한다는 점에서 실시간에 해를 구할 수 있다.

위의 사실을 바탕으로 전체 문제를 풀기 위한 방법론을 살펴보면, 노드 S에서 출발하여 4.2절에서 설명한 동적 계획법을 이용하여 f_i 값을 찾으면서 진행한다. AND SPLIT i 를 만났을 때, 현재의 f_{AS_i} 의 값과 노드 AS_i 와 노드 AJ_i 사이의 노드들에 대한 시간창을 검사한다. 각 노드에서 제일 짧은 거리를 갖

는 아크들의 시간창이 f_{AS_i} 이후의 시간을 모두 포함하고 있으면 4.1절에서 설명한 최단 경로를 찾는 알고리즘으로 문제를 풀이한다. 그렇지 않은 경우에는 AS_i 과 AJ_i 까지의 노드들에 대한 정수 계획법을 이용하여 f_{AJ_i} 를 구한다. 이를 정리하면 다음의 알고리즘 2와 같다. 최단 경로를 찾는 알고리즘을 편의상 알고리즘 1이라고 명명한다.

알고리즘 2

단계 1 : 초기화

$i = S$

$f_i = 0$

$j = i$ 의 후행 노드

단계 2 : 노드 판별

j 가 노드 T 이면 끝

i 가 노드 AS_k 이면 단계 3 아니면 단계 5

단계 3 : 시간창 검사

AS_k 에서 AJ_k 에 속한 각 노드를 서로 연결하는 아크들 중 최소의 시간을 소요하는

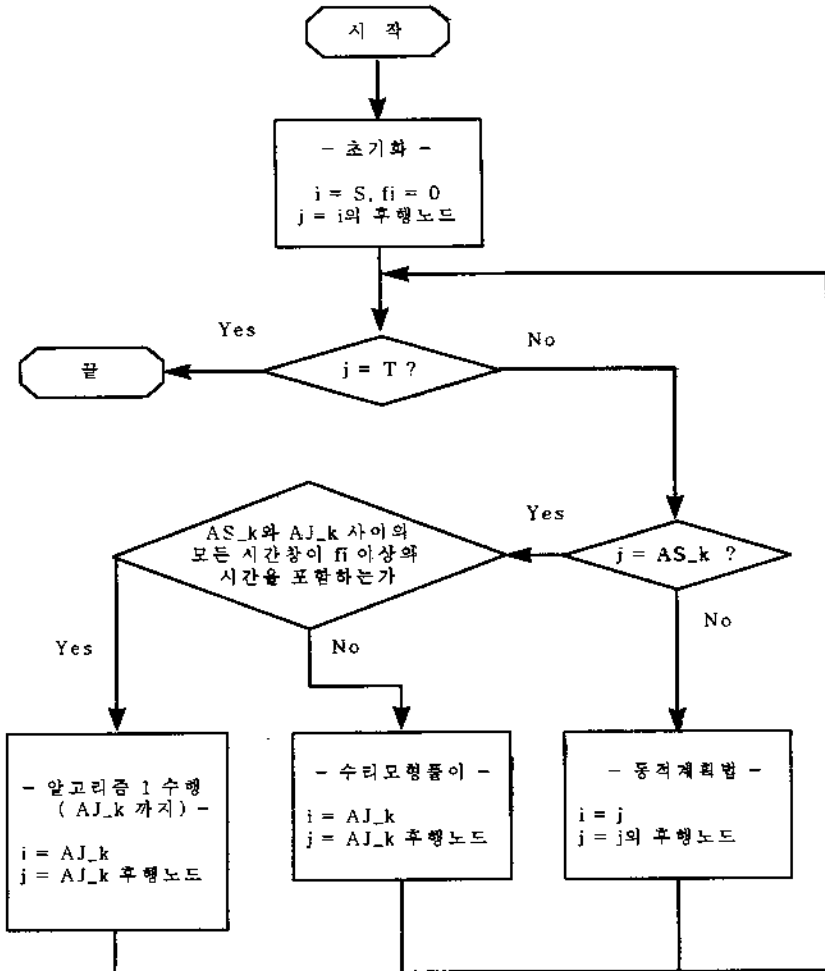


그림 8. 알고리즘 2의 흐름도

아크의 마지막 시간창의 시작값이 f_i 보다 작으면 다음을 수행, 아니면 단계 4

알고리즘 1을 AJ_k 노드까지 수행

$i = AJ_k$

$j = AJ_k$ 의 후행 노드

단계 4 : 수리 모형 풀이

AS_k 와 AJ_k 사이의 노드로 4.1절의 수리식을 형성하고 풀이

$i = AJ_k$

$j = AJ_k$ 의 후행 노드, 단계 2

단계 5 : f_j 계산

4.2절의 동적 계획법을 이용하여 f_j 계산

$i = j$

$j = j$ 의 후행 노드, 단계 2

다음의 그림 8에 알고리즘 2의 흐름도를 도시했다.

위의 예제 부품 3에 대해 알고리즘 2를 이용하여 풀이하여 본다. 풀이 과정을 다음의 표 4에 정리했다.

위에서 AS_1 에서 AJ_1 으로의 f_j 값 계산은

표 4. 예제 부품 3에 대해 풀이한 과정

j	i(선행노드)	k(호)	T_{jk}	비고	f_j
S3	null	null	0		0
31	S3	-	0	✓✓	0
OS_31	31	2	47		39
		4	39	✓✓	
32	OS_31	-	39	✓	39
33	32	1	68		64
		2	64	✓	
		3	81		
34	33	1	83		82
		3	82	✓	
AS_31	OS_31	-	39	✓✓	39
AJ_31	AS_31	-	89	✓✓	89
OJ_31	34	2	97		89
		3	99		
	AJ_31	-	89	✓✓	
T3	OJ_31	-	89	✓✓	89

[문제 1]의 수리식을 풀어서 나온 값이다. 위의 식을 풀이한 결과 최적해는 89분으로 나왔고 변수 Y352, Y362, Y371, Y382, W357 그리고 W358 는 1, 모든 L은 0으로 결정되었다. 즉 가공 순서는 공정 5, 공정 7, 공정 8, 공정 6의 순서이고, 공정 5는 기계 2에서 39분부터 55분까지, 공정 7은 기계 1에서 55분부터 67분까지, 공정 8은 기계 2에서 67분부터 77분까지, 공정 6은 기계 2에서 77분부터 89분까지 가공한다.

4.4 부품 투입 순서의 결정

본 연구에서는 전체 부품을 동시에 고려하는 것이 아니라 순차적으로 고려하기 때문에 부품의 투입 순서는 전체 수행도에 결정적인 영향을 줄 수 있다. 이에 대해 본 연구에서

는 우선 순위를 부여하여 부품의 투입 순서를 결정한다. 우선 순위는 주문의 우선 순위가 높은 부품, 평균 대체 기계(alternative machine)의 수가 적은 부품의 순서로 부품 투입에 대한 우선 순위가 부여된다. 이 우선 순위에 따라 모든 부품의 투입 순서가 미리 결정된다.

5. 실험 결과 분석

4절에서 소개한 알고리즘으로 풀이한 경우 최적해와의 차이를 분석하기 위한 실험을 수행했다. 4절에서 소개한 예제 부품과 동일한 AND/OR 그래프의 구조를 가진 부품 타입에 대해 각 공정을 가공할 수 있는 기계와 가공 시간을 무작위로 선택한 자료를 이용한 실험을 수행했다. 공장의 기계는 모두 다섯 대라고 가정했다. [실험 1]은 예제 부품 1의 구조, [실험 2]는 예제 부품 2의 구조, [실험 3]은 예제 부품 3의 구조를 가진 부품을 각각 3개씩 가공하는 경우이고, [실험 4]는 각 예제 부품을 하나씩 가공하는 경우이다. [실험 1]과 동일한 환경에서 20회의 예비 실험을 실시한 결과, 4절의 알고리즘에 의한 최종 가공 완료 시간의 평균은 51.65분, 분산은 8.76분이었다. [문제 1]의 수리 모형을 분지한계법으로 풀이한 결과 최적해의 최종 가공 완료 시간의 평균은 45.1분, 분산은 4.77분이었다. 유의도를 0.05, 허용오차를 0.05로 정하고 본 실험의 실험 회수를 산출하면 4절의 알고리즘에 의한 실험은 50회, 최적해의 실험은 20회로 결정된다. 실험 결과를 다음 표 5에 정리했다. 분지한계법 열의 값은 20개의 예제에 대한 최적해 실험 결과의 평균값이고, 4

표 5. 최적해와의 비교 실험 결과

구분/알고리즘		분지한계법 (최적해)	4절 알고리즘	최적해/4절 알고리즘
실험1	가공완료 시간(분)	45.35	52.28	0.867
	수행시간 (초)	493.39	0.17	2892.08
실험2	가공완료 시간(분)	45.10	57.32	0.786
	수행시간 (초)	70.03	0.05	1417.80
실험3	가공완료 시간(분)	58.35	73.02	0.799
	수행시간 (초)	784.40	0.24	3284.95
실험4	가공완료 시간(분)	51.15	65.78	0.777
	수행시간 (초)	200.31	0.14	1357.12
합계	가공완료 시간(분)	49.99	62.1	0.807
	수행시간 (초)	387.05	0.15	2237.99

절 알고리즘 열의 값은 50개의 예제에 대한 실험 결과의 평균값이다. 최적해/4절 알고리즘 열의 값은 최적해와 4절의 알고리즘과의 수행도 비교를 위해 최적해의 평균값을 4절의 알고리즘의 평균값으로 나눈 값이다. 결과적으로 본 연구에서 제시한 알고리즘에 의한 해는 최적해와 비교하여 80.7%의 수준인 반면, 수행 시간은 분지한계법으로 최적해를 그한 경우와 비교해서 1/2238이 소요된다. 더군다나 부품의 수가 증가할수록 분지한계법에 의한 최적해의 풀이에 걸리는 시간은 지수승으로 증가하게되지만, 본 연구에서 제시

표 6. bound 알고리즘과의 비교 실험 결과

구분/알고리즘		bound 알고리즘	4절 알고리즘	bound/ 4절
실험5	가공완료 시간(분)	284.92	274.68	1.04
	수행시간 (초)	0.28	0.12	2.25
실험6	가공완료 시간(분)	320.00	301.80	1.06
	수행시간 (초)	0.55	0.31	1.77
실험7	가공완료 시간(분)	514.60	511.80	1.01
	수행시간 (초)	52.16	1.17	44.58
실험8	가공완료 시간(분)	895.40	936.60	0.92
	수행시간 (초)	2283.34	3.27	698.27

한 알고리즘은 부품의 수에 비례하여 시간이 증가하므로 현실적으로 의미 있는 시간에 해를 구할 수 있다.

또한 부품별로 문제를 분해해서 계획을 수립하는 경우와 시간대별로 문제를 분해해서 계획을 수립하는 경우의 수행도 분석을 위해, Nasr[13]이 제시한 bound 알고리즘과의 비교 실험을 실시했다. Nasr이 제시한 알고리즘은 대체 기계만을 허용한 상태에서 적용될 수 있는 알고리즘이라서 본 실험에서는 공정은 결정되어있고 대체 기계만을 고려한다. 사용된 데이터는 T중공업의 공정계획 결과 자료이다. T 중공업은 축, 케이스, 기어 등을 생산하는 업체이다. 축은 보통 10-15 공수, 케이스는 10개 미만의 공수, 기어는 20-25 공수로 이루어진다. 한 공정당 소요시간은

10-20분 소요된다. 따라서 실험에 이용된 부품의 공수는 타입 1이 23, 타입 2가 13, 타입 3이 8, 타입 4가 4이다. [실험 5]는 각 타입별로 하나씩의 부품을 가공, [실험 6]은 타입별로 두 개씩의 부품을 가공, [실험 7]은 타입별로 다섯 개의 부품을 가공, [실험 8]은 타입별로 열 개의 부품을 가공하는 실험이다. 가공 기계는 모두 8대이고, 실험에 이용된 데이터는 난수를 이용하여 만들었다. 실험 회수를 결정하기 위해 [실험 5]와 동일한 환경에서 예비 실험을 50회 실시한 결과 4절의 알고리즘에 의한 최종 가공 완료 시간의 평균은 274.68분, 분산은 0.12분이었다.

Nasr의 bound 알고리즘에 의한 평균은 284.92분, 분산은 0.28이었다. 유의도를 0.05, 허용오차를 0.05로 정하고 본 실험의 실험 회수를 산출하면 4절의 알고리즘에 의한 실험은 3회, bound 알고리즘은 4회로 결정된다. 실험 결과를 다음 표 6에 정리했다. [실험 5]는 50개의 예제에 대한 평균값이고, [실험 6]-[실험 8]은 다섯 개의 예제에 대한 평균값이다. Bound 알고리즘의 수행 시간이 부품의 수가 증가할수록 큰 폭으로 증가하는 이유는 알고리즘의 내부에 할당문제(assignment problem)를 풀이하는 부분에서 소요되는 시간이다. 전체적으로 비슷한 수준의 해를 제공하고 있으므로 본 논문에서 제시한 알고리즘이 대체 기계만 고려한 경우도 좋은 수행도를 가진 해를 빠른 시간에 산출한다고 볼 수 있다.

모든 실험은 SUN Ultra 워크스테이션에서 수행되었으며, 최적해는 CPLEX의 IP optimizer를 이용해서 구했다.

6. 결론 및 추후 연구과제

본 연구에서는 생산 기술 영역과 생산 관리 영역의 통합을 위해 가장 중요한 부분 중 하나인 공정 선택과 기계 선택과 일정계획의 통합을 위한 수리 모형을 제시했다. AND/OR 그래프로 입력된 공정 정보와 공장의 부하 상태를 반영하여 최종 가공 완료 시간을 최소화 할 수 있는 수리 모형을 제시하였으며, 부품 단위로 문제를 분해하여 전체 문제를 풀이할 수 있는 알고리즘을 제시하였다. 또한 해법과 관련하여서는 그래프 이론을 확장하여 적용함으로써 실시간에 문제 해결이 가능하도록 하였다. 본 연구에서 제시한 방법론에 의한 해는 최적해와 비교해서 평균 80.7% 정도의 수행도를 보이고 있으며, 수행 시간 면에서는 평균 1/2238의 시간이 소요되었다.

추후 전체 문제를 공정결정, 기계결정 문제와 일정계획 문제로 분해한 후 문제를 풀고 결과를 분석하여 병목 기계를 선정하고 이를 바탕으로 각 기계에 가중치를 부여한 후 다시 공정결정, 기계결정 문제를 풀이할 수 있는 방법론에 대한 연구를 수행할 계획이다.

참 고 문 헌

- [1] Ahuja, R. K., P. L. Magnanti, and J. B. Orlin, *Network Flows*, Prentice Hall, 1993.
- [2] Brandimarte, P. and M. Calderini, "A hierarchical bicriterion approach to integrated process plan selection and job shop scheduling," *International Journal of Produc-*

- tion Research, Vol. 33, No. 1, pp. 161-181, 1995.
- [3] Chen, Q. and B. Khoshnevis, "Scheduling with flexible process plans," Production Planning & Control, Vol. 4, No. 4, pp. 333-343, 1993.
- [4] De Mello, H. L. S. and A. C. Sanderson, "AND/OR graph representation of assembly plans," Proceedings of AAAI-86, Vol. 2, pp. 1113-1119, 1986.
- [5] Halpern, J., I. Priess, "Shortest path with time constraints on movement and parking," Networks, Vol. 4, pp. 241-253, 1974.
- [6] Kruth, J. P. and J. Detand, "A CAPP system for nonlinear process plans," Annals of CIRP, Vol. 41, No. 1, 1992.
- [7] Kuhn, H., "A heuristic algorithm for the loading problem in flexible manufacturing systems," International Journal of Flexible Manufacturing Systems, Vol. 7, pp. 229-254, 1995.
- [8] Laliberty, T. etr., "A blackboard architecture for integrated process planning/production scheduling," <http://agile.cimds.ri.cmu.edu>, 1996.
- [9] Lee, S., R. A. Wysk and J. S. Smith, "Process planning interface for a shop floor control architecture for computer-integrated manufacturing," International Journal of Production Research, Vol. 33, No. 9, pp. 2415-2435, 1995.
- [10] Lenderink, A. and H. J. J. Kals, "The integration of process planning and machine loading in small batch part manufacturing," Robotics & Computer Integrated Manufacturing, Vol. 10, No. 1, pp. 89-98, 1993.
- [11] Liao, T. W., E. R. Coates, R. Aghazadeh, L. Mann and N. Guha, "Modification of CAPP systems for CAPP/scheduling integration," Computers and Industrial Engineering, Vol. 25, Nos 1-4, pp. 203-206, 1993.
- [12] Mettala, E. G. and S. B. Joshi, "A compact representation of process plans for FMS control activities," Penn State University IMSE working paper series, 1990.
- [13] Nasr, N. and E. A. Elsayed, "Job shop scheduling with alternative machines," International Journal of Production Research, Vol. 28, No. 9, pp. 1595-1609, 1990.
- [14] Ray, S. R. and A. B. Feeney, "A national testbed for process planning research," From <http://elib.cme.nist.gov/fasd/pubs/ray93.html>, 1993.
- [15] Sancho, N. G. F., "A dynamic programming solution of a shortest path problem with time constraints on movement and parking," Journal of Mathematical Analysis and Applications, Vol. 166, pp. 192-198, 1992.
- [16] Sancho, N. G. F., "Shortest path problems with time windows on nodes and arcs," Journal of Mathematical Analysis and Applications, Vol. 186, pp. 643-648, 1994.

- [17] Zhang, H.-C., "TPPM - A prototype to integrate process planning and job shop scheduling functions," Annals of CIRP, Vol. 42, No. 1, pp. 513-518, 1993.
- [18] Zijm, W. H. M., "The integration of process planning and shop floor scheduling in small batch part manufacturing," Annals of CIRP, Vol. 44, No. 1, pp. 429-432, 1995.
-

96년 11월 최초 접수, 97년 4월 최종수정