

단일설비 생산체제에서 부품의 일정계획에 관한 발견적 기법*

A Heuristic for Scheduling Production of Components at a Single Facility*

김봉진**

Bong jin Gim**

Abstract

We consider a single-machine scheduling problem dealing with the manufacture of components for subsequent assembly into end products. Each product requires both unique components and common components, and each production batch requires a setup. By making some assumptions on the data and the availability of the components for assembly, Baker provides an efficient dynamic programming algorithm for obtaining the optimal schedule. In this paper we do not impose any requirement on the data, and we solve the more complicated batching and sequencing problem. We suggest a simple heuristic method that is efficient and finds solutions that are optimal or close to the optimal solution.

1. 서론

본 논문에서는 단일설비에서 여러 부품(component)들을 생산하여 다음 공정인 최종 조립공정에서 이들 부품들을 여러 형태로 조립하여 최종제품들을 생산하는 일정계획 문제를 다룬다. 각 최종제품은 그 제품에만 필요한 고유 부품과 모든 제품에 소요되는 공

통 부품으로 구성되며, 이러한 부품들은 모두 한 설비에서 생산되는 시스템을 가정한다. 각 부품들의 소요량은 자재 소요계획(MRP)의 전개논리를 이용하여 얻을 수 있으며, 위와 같은 예는 자동차 산업의 엔진 조립공정에서 찾을 수 있다. 여러 제품에 같이 소요되는 공통 부품때문에 여러 부품들을 batch로 생산하는 것이 유리하며, 한 부품에서 다

* 본 연구는 1991년도 한국과학재단의 일반기초연구 연구비 지원에 의해 수행되었음.

** 단국대학교 산업공학과

른 부품으로 생산이 전환될 때마다 작업준비 시간이 발생한다. 작업준비 시간은 각 부품의 생산 순서에 관계없이 일정하고 batch의 크기에는 상한이 없는 것으로 가정하였다. 본 논문에서 다룰 문제는 모든 제품의 총 흐름 시간을 최소화하는 일정 계획을 수립하는 것이다. 작업이 완료된 부품의 이용가능 시점에 따라 각 제품의 흐름시간이 달라지며, 완성된 부품의 이용가능 시점에 대한 두 개의 대표적인 가정이 존재한다. 'item-availability'는 각 부품이 현 설비에서의 작업이 완료되면 즉시 이용가능한 경우이며, 'batch-availability'는 한 부품의 작업이 완료되더라도 batch 내의 모든 부품들의 작업이 완료되는 시점에서 모든 부품들이 동시에 이용가능한 경우이다. 이러한 가정들은 Santos와 Magazine [3]에 의해 정의되었으며, 어떠한 가정이 적합한지는 주어진 생산 공정의 자재취급 형태에 따라 결정된다.

Baker [2]는 공통 부품들이 그들의 production run이 끝난 후에 이용 가능하다는 가정하에 총 흐름시간을 최소화하는 문제를 다루었다. 또한, 그는 한 제품의 고유 부품의 총 작업시간이 다른 제품의 고유 부품의 총 작업시간보다 적으면, 공통 부품의 작업시간에 대해서도 항상 같은 관계가 성립하는 것으로 가정하였다. 이러한 가정은 각 제품(또는 작업)을 Shortest Processing Time(SPT) 순서대로 생산하도록 유도하며, 따라서 생산순서 결정(sequencing decision)과 배치 결정(batching decision)을 분리시킬 수 있게 만든다. 위와 같은 가정하에 Baker [2]는 모든 제품의 총 흐름시간을 최소화할 수 있는 유효한 동적계획법 알고리즘을 개발하였다. Aneja와 Singh

[1]은 각 제품이 하나의 공통 부품과 m ($m \geq 2$) 개의 서로 다른 공통 부품을 갖는 경우에 대하여 Baker의 알고리즘을 확장한 동적 계획법 해법을 개발하였다. Vickson, Magazine, Santos [4]는 'item-availability'와 각 작업의 SPT 생산순서 가정하에 총 흐름 시간을 최소화하는 동적계획법 알고리즘을 개발하였다.

본 논문의 문제는 'batch-availability'와 공통 부품들이 그들의 production run이 끝난 후에 이용 가능하다는 가정하에 총 흐름시간을 최소화하는 문제로서, Baker의 모형과 틀린 점은 각 제품의 고유 부품과 공통 부품의 생산 시간에 대하여 어떠한 제약도 없다는 점이다. 따라서, 본 논문의 문제는 Baker의 문제를 일반화시킨 문제이며, SPT 생산순서가 최적해를 도출한다는 것을 보장하지 못하는 경우이다. 즉, 본 논문의 문제는 Baker [2]가 이미 지적했듯이 생산순서 결정과 배치 결정을 분리시킬 수 없는 풀기 어려운 문제가 된다. 본 논문에서는 위와 같은 문제에 대한 최적해 또는 최적해와 근사한 해를 구할 수 있는 간단하고 능률적인 발견적 기법을 제시하고자 한다.

본 논문은 다음과 같이 구성된다. 제 2절에서는 단일설비 생산체제에서 부품의 일정 계획 문제에 대한 정의를 하고, 제 3절에서는 위 문제에 대한 분석을 한다. 제 4절에서는 제안된 발견적 기법을 서술하고, 제 5절에서는 수치 예를 이용하여 제안된 발견적 기법의 과정들을 구체적으로 설명한다. 제 6절에서는 제안된 발견적 기법의 해를 최적해와 비교한 근접도를 평가한다.

2. 문제의 정의

N 을 일정계획 대상인 제품의 수로 표시하자. 또한, 다음과 같은 기호를 사용하였다.

T =한 batch 내의 공통 부품의 작업준비 시간,

C_j =제품 j 의 공통 부품의 한 개당 생산 시간(run time),

U_j =제품 j 의 고유 부품의 한 개당 총 생산시간(setup and run time),

$S = \langle S_N(1), S_N(2), \dots, S_N(N) \rangle$: N 개의 제품의 생산 순서.

우리는 고유 부품에 대한 작업준비 시간을 총 생산시간 U_j 에 포함시켰다. 따라서, 본 논문에서 사용하는 작업준비 시간이라는 용어는 오직 공통 부품에 대한 작업준비 시간을 의미한다. 그러면, 세 개의 제품에 대한 일정계획 $TC_2C_1U_2U_1TC_3U_3$ 를 다음과 같이 해석할 수 있다: 먼저 작업준비를 하고 제품 2와 제품 1의 공통 부품들을 생산한 다음에 제품 2와 제품 1의 고유 부품들을 차례로 생산한다. 다시 작업준비를 한 다음에 제품 3의 공통 부품과 제품 3의 고유 부품을 생산한다. 여기서, 제품 2와 제품 1은 첫째 batch를 형성하며 제품 3는 단독으로 둘째 batch를 형성한다. 또한, 위와 같은 일정계획의 생산 순서, S ,는 $\langle 2, 1, 3 \rangle$ 이다.

하나의 작업(job)은 각 제품에 필요한 수만큼의 공통 부품과 고유 부품을 조립한 것으로 정의할 수 있다. 한 작업은 그에 필요한 공통 부품과 고유 부품이 모두 확보되었을 때 완료되었다고 말한다. 위와 같은 일정

계획 문제에 대한 기준은 N 개의 작업에 대한 총 흐름시간 또는 평균 흐름시간을 최소화하는 것이다. 각 제품의 완료 시간은 조립 단계의 각 작업의 완료 시간과 연계되어 결정되므로, 모든 작업의 총 흐름시간을 최소화하는 것은 총 재공품 재고비용을 최소화하는 것과 일치하게 된다. 따라서, 단일설비 생산체제에서 부품의 일정계획 문제는 N 개의 작업의 총 흐름시간을 최소화하는 최적 배치 결정과 최적 생산순서를 결정하는 것이다.

3. 분석

본 논문에서는 'batch-availability'를 가정하였다. 이러한 가정하에 Baker [2]는 각 batch 내에서 모든 공통 부품들을 고유 부품에 선행하여 생산하는 것이 최적임을 보였다. 또한, 공통 부품들의 생산순서는 임의로 선택하여도 총 흐름시간이 일정하게 된다. 각 제품의 생산 순서 S 가 주어졌다고 가정하자. 또한 $C_{[k]}$ 를 생산 순서 S 의 k 번째 공통 부품의 생산 시간으로 표시하자. 그러면, Baker의 동적계획법 알고리즘을 임의의 생산 순서에 대해 일반화하여, 다음과 같은 동적계획법 순환 관계를 이용한 최적 배치결정을 얻을 수 있다:

$$F(k) = \min_{1 \leq i \leq k} \{F(i-1) + v_{ik}\}, \quad k=1, 2, \dots, n. \quad (1)$$

$$\text{where } v_{ik} = (n-i+1)T - (k-i+1)(C_{[k+i]} + \dots + C_{[a]}), \text{ and } F(0) = 0.$$

여기서, 한 주어진 batch를 구성하고있는 작

업(또는 제품)이 알려졌다고 가정하자. 그러면, 다음 정리는 최적 일정계획의 일반적인 특성을 나타낸다.

정리 1. 어떠한 batch에 대해서도 batch 내의 고유 부품들은 그들의 총 생산시간을 기준으로 SPT 순서대로 생산하는 것이 최적이다.

증명. 한 batch 내의 두 작업 i 와 j 의 고유 부품들의 총 생산 시간이 $U_i > U_j$ 관계를 갖고 있을 때, 고유 부품 i 를 고유 부품 j 보다 먼저 생산하는 non-SPT 일정 계획을 생각해 보자. 여기서, 두 고유 부품 i 와 j 가 서로 인접하여 있다고 가정하여도 증명의 일반성을 잃지 않는다. 그런데 공통 부품들은 'batch-availability' 가정에 의하여 고유 부품보다 항상 선행하여 생산되므로, batch 내의 다른 부품들의 생산 순서는 고정시키고 두 개의 고유 부품 i 와 j 의 생산 순서만을 서로 바꾸는 일정계획의 총 흐름시간은 원래 일정계획의 총 흐름시간보다 ΔF 만큼 적게 된다. 여기서, $\Delta F = U_j - U_i > 0$ 이며 작업 i 와 j 를 제외한 batch 내의 다른 작업들의 완료 시간은 변하지 않는 것을 알 수 있다. 위와 같은 일련의 인접작업 교환(pairwise interchanges)을 통하여 우리는 고유 부품들의 최적 생산순서가 SPT 형태인 것을 도출할 수 있다. Q. E. D.

한 batch 내에서는 공통 부품들이 항상 고유 부품에 선행하여 생산되고 고유 부품들은 정리 1에 의하여 총 생산 시간이 적은 고유 부품부터 차례대로 생산하는 것이 최적이므로, 우리는 주어진 batch의 최적 일정계획을 쉽게 얻을 수 있다. 예를 들어, 제품 1과 제

품 2가 첫 번째 batch를 형성하며 $U_2 < U_1$ 인 경우를 생각해 보자. 그러면, 정리 1에 의하여 제품 2의 고유 부품이 제품 1의 고유 부품보다 먼저 생산되어야 하고, 제품 1과 제품 2의 공통 부품들은 고유 부품에 선행하여 생산되어야 하므로 $TC_1C_2U_2U_1$ 이 첫 번째 batch의 최적 일정계획이다. 또한, 고유 부품들의 생산 순서는 임의로 선택할 수 있으므로 $TC_2C_1U_2U_1$ 도 첫 번째 batch의 최적 일정계획이 된다.

4. 발견적 기법

본 논문에서 제시할 발견적 기법은 세 단계로 구성된다. 첫 단계는 N 개의 제품이 각기 다른 batch를 구성하고, 이러한 N 개의 batch에 대한 초기 생산순서를 다음과 같이 정렬하여 얻는다:

$$C_{[1]} + U_{[1]} \leq C_{[2]} + U_{[2]} \leq \dots \leq C_{[N]} + U_{[N]} \quad (2)$$

단계 2. 도전 일정계획(challenging schedule)을 만든다. 방어 일정계획(defending schedule)은 현재까지 발견된 가장 우수한 일정계획이며, 도전 일정계획은 방어 일정계획의 첫째 batch와 둘째 batch를 합병하고 기타 batch는 방어 일정계획과 동일하게 만든 일정계획이다. 만약, 합병할 batch 수가 1 이하이면 모든 과정을 끝낸다.

단계 3. 방어 일정계획의 총 흐름시간과 도전 일정계획의 총 흐름시간을 비교한다. 만약, 도전 일정계획의 총 흐름시간이 방어 일정계획의 총 흐름시간보다 적으면 도전 일정

계획을 다음 반복의 방어 일정계획으로 정한다. 그렇지 않으면, 다음 반복에서의 방어 일정계획은 현 방어 일정계획과 동일하다. 그러나, 이러한 경우에는 현 방어 일정계획의 첫째 batch를 고정된 batch로 생각하고, 현 방어 일정계획의 둘째 batch를 다음 반복에서의 방어 일정계획의 첫째 batch로 간주한다. 단계 2로 간다.

본 논문에서 제안된 발견적 기법은 하나의 가능한 일정계획에서 출발하여 batch의 합병을 통하여 더 나은 가능한 일정계획을 발견하는 일종의 개선 알고리즘 (improvement algorithm)이다. 단계 2에서 도전 일정계획은 방어 일정계획의 첫째 batch와 둘째 batch를 하나의 batch로 합병한 일정계획이므로, 정리 1의 결과를 이용하여 최적 도전 일정계획을 간단하게 구할 수 있다. 또한, 정리 1은 모든 batch 들의 최적 일정계획을 얻기 위하여 반복적으로 계속 이용된다. 한편, 도전 일정계획의 batch 수는 방어 일정계획의 batch 수에 비해 항상 1 만큼 적게 되며, 매 반복마다 첫째 batch와 둘째 batch가 합해 지거나, 또는 첫째 batch가 분리되므로 모든 과정을 끝내기 위해서는 N-1 번의 반복이 필요하게 된다.

5. 예제

제안된 발견적 기법의 과정들을 자세히 보여주기 위해서 다음과 같은 예제를 사용하였다. 여기서, 제품의 수는 4이고 작업준비 시간은 2이며, 다음과 같은 자료가 주어졌다고 가정하자:

	1	2	3	4
C_j	2	1	6	4
U_j	5	9	2	5

단계 1에서 우리는 (2)의 관계를 이용하여 초기 생산순서, $S = \langle 1, 3, 4, 2 \rangle$,를 얻을 수 있다. 또한, N 개의 제품을 별도의 batch로 가정하였으므로 발견적 기법의 초기 일정계획은 $TC_1U_1TC_3U_3TC_4U_4TC_2U_2$ 로 주어지며, 이러한 일정계획의 총 흐름시간은 100이 된다. 참고로, 초기 일정계획의 각 작업별 완료시간은 다음과 같다:

작업	완료시간
1	$T+C_1+U_1=9$
3	$9+T+C_3+U_3=19$
4	$19+T+C_4+U_4=30$
2	$30+T+C_2+U_2=42$

단계 2에서 방어 일정계획은 $TC_1U_1TC_3U_3TC_4U_4TC_2U_2$ 이고, 도전 일정계획은 $TC_1C_3U_3U_1TC_4U_4TC_2U_2$ 이 된다. 여기서, 도전 일정계획의 첫째 batch인 $TC_1C_3U_3U_1$ 는 방어 일정계획의 첫째 및 둘째 batch인 제품 1과 제품 3가 합쳐진 것이다. 또한, 공통 부품인 C_4 과 C_2 는 고유 부품에 선행하여 생산되며 $U_3 < U_1$ 이므로, 정리 1에 의하여 제품 3의 고유 부품이 제품 1의 고유 부품보다 먼저 생산되어야 한다. 이러한 도전 일정계획의 총 흐름시간은 97이다. 단계 3에서 도전 일정계획의 총 흐름시간이 방어 일정계획의 총 흐름시간보다 적으므로 $TC_1C_3U_3U_1TC_4U_4TC_2U_2$ 이 새로운 방어 일정계획이 된다.

두 번째 반복에서, 위와 같은 방법으로 도

전 일정계획은 $TC_1C_3C_4U_3U_4U_1TC_2U_2$ 이 된다. 위와 같은 도전 일정계획의 총 흐름시간은 104 이다. 도전 일정 계획이 방어 일정계획에 비해 열등하므로 방어 일정계획은 그대로 유지된다. 그러나, 다음 반복에서는 현 방어 일정계획의 첫째 batch인 $TC_1C_3U_3U_1$ 은 별도의 batch로 고정하고 둘째 batch인 TC_4U_4 를 첫째 batch로 간주한다. 따라서, 반복 3의 도전 일정계획은 $TC_1C_3U_3U_1TC_2C_4U_4U_2$ 가 된다.

표 1에 예제의 각 반복별 일정계획과 총 흐름시간을 수록하였다. 매 반복마다 방어 일정계획의 첫째 batch는 분리된 것으로 간주되거나 둘째 batch와 합해진다. 위와 같은 예제의 발견적 기법에 의한 최종 일정계획은 $TC_1C_3U_3U_1TC_2C_4U_4U_2$ 이며, 이러한 최종 일정계획의 총 흐름시간은 96 이다. 또한, 위와 같은 예제에 대하여 발견적 기법에 의한 일정계획과 최적 일정계획은 일치한다.

5, 6, 7, 8이 선택되었으며, 각 작업준비 시간 및 N 값에 대해 20개의 문제가 무작위로 발생되었다. 따라서, 각 N 값에 대해 40개의 문제가 발생되었으며 총 문제의 수는 200이다.

여러 문제들을 발생시킨 첫째 목적은 제안된 발견적 기법과 다른 발견적 기법의 해들을 비교하는 것이다. 하나의 합리적인 발견적 기법은 각 제품의 생산순서를 (2)에 따라 결정한 후, 주어진 생산순서에 대한 최적 배치 결정을 (1)에 의해 결정하는 것이다. 앞으로 위와 같은 발견적 기법을 SPT 발견적 기법으로 표시하자. 둘째 목적은 제안된 발견적 기법의 해와 최적 해를 비교하여 발견적 기법의 실험적 근접도를 측정하는 것이다. 각 문제의 최적 해는 모든 가능한 생산순서를 나열하여 얻을 수 있다. 제안된 발견적 기법, SPT 발견적 기법 및 총 나열 기법의 과정들은 모두 MS-FORTRAN 으로 프로그램이

표 1. 예제의 각 반복별 일정계획 및 총 흐름시간

반복	방어 일정계획	총 흐름시간	도전 일정계획	총 흐름시간
1	$TC_1U_1TC_3U_3TC_4U_4TC_2U_2$	100	$TC_1C_3U_3U_1TC_4U_4TC_2U_2$	97
2	$TC_1C_3U_3U_1TC_4U_4TC_2U_2$	97	$TC_1C_3C_4U_3U_4U_1TC_2U_2$	104
3	$TC_1C_3U_3U_1TC_4U_4TC_2U_2$	97	$TC_1C_3U_3U_1TC_2C_4U_4U_2$	96

6. 계산 결과

제안된 발견적 기법에 의한 해의 최적해에 대한 근접도를 측정하기 위하여 우리는 여러 set의 문제들을 발생시켰다. 각 문제에서 고유 부품과 공통 부품의 생산시간 및 총생산시간은 1과 10 사이의 정수 중에서 무작위로 선택하였다. 또한, 작업준비 시간은 2와 10이 같은 비율로 선택되었다. 제품의 수(N)는 4,

작성되었으며, 25 MHz 80386 마이크로 컴퓨터로 해를 구하였다. 위와 같은 문제에 대한 계산 결과를 표 2에 수록하였다.

표 2의 결과를 보면 제품의 수가 증가할수록 제안된 발견적 기법의 근접도가 떨어지는 경향을 보였으나, 일반적으로 제안된 발견적 기법의 근접도는 우수한 것으로 나타났다. 위와 같은 검사 문제들을 기준하여, 제안된 발견적 기법의 평균 근접도는 99.0 % 이고 가

표 2. 검사 문제에 대한 계산 결과

N	SPT 발견적 기법		제안된 발견적 기법		총 나열 기법
	시간	근접도	시간	근접도	시간
4	0.2	100.0 %	0.1	100.0 %	0.8
5	0.3	99.4 %	0.2	99.4 %	4.1
6	0.4	98.8 %	0.2	98.9 %	23.5
7	0.7	98.3 %	0.3	98.5 %	187.6
8	1.1	98.0 %	0.4	98.2 %	1512.8

시간: 편집 및 연결 시간을 제외한 평균 계산시간(초).

근접도: 최적해를 발견적 기법의 해로 나눈 평균 비율.

장 나쁜 근접도도 96.3 %인 것으로 나타났다. 또한, 제안된 발견적 기법의 해는 총 200개의 검사 문제중 163 (81.5 %) 개의 문제에서 최적 해와 일치하였다. 마이크로 컴퓨터의 기억용량의 제한때문에 N이 9 이상인 경우에는 총 나열기법으로 최적해를 얻을 수 없었다. 그런데, 특기할 사항은 제품의 수가 N일 때 단일 설비에서 생산되는 부품의 수는 2N이 된다. 따라서, 제품의 수가 8인 경우에 총 생산 부품의 수는 16이 되며, 단일 설비에서 16개의 부품을 생산하는 문제는 현실적으로 충분한 규모의 문제로 간주할 수 있다.

표 2의 결과의 결과를 보면 SPT 발견적 기법의 근접도도 우수한 것을 알 수 있다. SPT 발견적 기법의 평균 근접도는 98.9 % 로서 제안된 발견적 기법의 성취도보다는 조금 떨어지나 문제에 따라서는 제안된 발견적 기법의 해보다 더 우수한 해를 발견한 경우도 있었다. 따라서, SPT 발견적 기법을 제안된 발견적 기법과 동시에 사용하면 더욱 효과적일 것으로 생각된다. 제안된 발견적 기법의 우수성은 계산 시간에서 찾을 수 있다. 제품의 수가 8인 경우에 SPT 발견적 기법의 평균 계

산 시간이 1.1 초인 반면에 제안된 발견적 기법은 오직 0.4 초가 소요되었다. 가능한 생산 순서를 모두 나열하는 방법의 계산 시간은 제품의 수가 증가함에 따라 지수적으로 증가하여 제품의 수가 8인 경우에 평균 1513 초나 소요되었다. 제안된 발견적 기법은 제품의 수가 10인 경우에도 쉽게 손으로 해를 구할 수 있으며, 실 시간(real time) 으로 해를 구해야하는 경우에 더욱 적합할 것으로 사료된다.

7. 결론

본 논문에서는 단일설비에서 여러 부품들을 생산하여 이들 부품들을 여러 형태로 조립하여 최종 제품들을 생산하는 일정계획 문제를 다루었다. 본 논문에서 사용한 기준은 총 흐름 시간을 최소화하는 것으로서 이러한 기준은 재공품 재고를 최소화하는 기준과 일치하게 된다. 본 논문에서는 Baker의 모형을 일반화하여 각 제품의 고유 부품 및 공통 부품의 생산 시간에 어떠한 제약도 없는 것으로 가정하였으며, 이러한 문제는 배치 결정과 생산순서 결정을 분리시킬 수 없는 풀기

어려운 문제가 된다. 따라서, 우리는 위와 같은 일정계획 문제에 대한 최적 해 또는 최적 해와 근사한 해를 얻을 수 있는 간단하고 능률적인 발견적 기법을 제시하고자 하였다. 제안된 발견적 기법은 하나의 가능한 일정계획에서 출발하여 batch의 합병을 통하여 개선된 일정계획을 찾아가는 개선 알고리즘으로서, $N-1$ 번의 반복을 거쳐 최종 일정계획을 얻게 된다. 우리는 여러 검사 문제들을 통하여 제안된 발견적 기법의 근접도가 우수함을 보여주었으며, 제안된 발견적 기법은 제품의 수가 많고($N > 6$) 실 시간으로 해를 구해야 할 경우에 적합할 것으로 사료된다.

참 고 문 헌

- [1] Aneja, Y. P., and Singh, N., "Scheduling Production of Common Components at a Single Facility," IIE Transactions 22, 3, 234-237 (1990).
- [2] Baker, K. R., "Scheduling the Production of Components at a Common Facility," IIE Transactions 20, 1, 32-35 (1988).
- [3] Santos, C., and Magazine, M., "Batching in Single Operation Manufacturing Systems," OR Letters 4, 3, 99-103 (1985).
- [4] Vickson, R. G., Magazine, M. J., and Santos, C. A., "Batching and Sequencing of Components at a Single Facility," IIE Transactions 25, 2, 65-70 (1993).