〈硏究論文〉

Journal of the Korean Institute of Industrial Engineers
Vol. 12, No. 1, June 1986

多数 機械의 総費用을 最小化하는 최적작업순서, 공통납기일 및 작업완료일 결정을 위한 日程計劃

(Optimum Scheduling Algorithm for Job Sequence, Common Due Date Assignment and Makespan to Minimize Total Costs for Multijob in Multimachine Systems)

魯 仁 珪*
金 相 澈*

Abstract

This research is concerned with n jobs, m parallel identical machines scheduling problem in which all jobs have a common due date. The objective of the research is to develop an optimum scheduling algorithm for determining an optimal job sequence, the optimal value of the common due date and the minimum makespan to minimize total cost. The total cost is based on the common due date cost, the earliness cost, the tardiness cost and the flow time cost of each job in the selected sequence.

The optimum scheduling algorithm is developed. A numerical example is given to illustrate the scheduling algorithm.

1. 서 론

본 연구는 다수의 기계들이 병렬로 배치되고, 납기일이 공통인 다수의 작업들이 수행되는 경우에 발생하는 총비용을 최소화하는 작업순서와 최적공통납기일을 구하고, 전체 작업의 완료시간을 최소화하는 문제에 관한 것이다.

모델설정에 있어서 몇 가지의 제한 조건하에서 다음 네 가지 종류의 비용들이 발생한다. 현 시점에서 작업들의 공통납기일까지에 대한 비용(due date assignment cost)이 발생하고, 납기일 전에 완성된 작업에 대해서는 재고비용(earliness cost)이 발생하며, 납기일 후에 완성된 작업에 대해서는 지연비용(tardiness cost)이 발생하고, 각 작업을 수행하는 과정

*漢陽大學校 工科大學 産業工學科

에서 흐름시간비용(flow time cost)이 발생한다. 이런 경우에 비용의 합을 최소화하는 작업순서와 최적공통납기일을 결정하고, 동시에 전체 작업들의 완료시간을 최소화하고자 하는 것이 본 연구의 목적이다.

본 연구와 관련된 기존 연구들은 다음과 같다. Sidney [14]는 한 대의 기계가 있고 각 작업의 시작시간과 만기시간이 주어진 경우에 시작시간보다 일찍 시작했을 때에 대한 비용과 만기시간보다 늦게 끝났을 때의 비용의 합을 최소화시키는 작업순서의 결정에 관해 연구하였다. Seidmann, Panwalkar, 및 Smith [11]는 공통의 납기일을 가지고 그에 대한 비용과 지연비용을 가진 작업들이 한 대의 기계에 의해서 실시될 때에 총비용이 최소가 되는 작업순서와 최적납기일을 결정하는 방법에 관해 연구하였다.

이를 기초로 다수의 기계에 의해서 작업들이 실시되는 경우에 대해서 연구 한 것이다.

2. 모델의 설정

2.1 전제조건

모델의 설정에 있어서 다음과 같은 조건을 전제로 한다.

(1) 병렬로 배치된 m개의 동일 기계들과 n개의 서로 다른 작업들이 있다. 단, $n > m$이다.

(2) 모든 작업들의 작업시간은 미리 정해진다.

(3) 작업을 수행하기 전의 준비시간과 작업들 간의 대기시간은 고려하지 않는다.

(4) 어느 작업을 먼저 실시하더라도 관계가 없다. 즉 작업들 간에는 상호 독립이다.

(5) 기계를 처음 사동하는 데에는 비용이 전혀들지 않는다.

(6) 납기일 전에 끝난 작업에 대한 물품들은 저장해 두었다가 공통납기일에 동시에 납품하고, 납기일 이후에 끝난 작업에 대한 물품들은 작업완료 즉시 납품한다.

(7) 운반비용은 전혀 고려하지 않는다.

(8) 모든 비용은 선형적이다. 즉 시간에 비례한다.

2.2 부호 설명

n개의 작업을 m개의 기계에 할당할 때에 임의의 기계에 배치되는 작업의 위치 (j)에 대해 기호는 다음과 같다.

- n_i : i 기계에 할당되는 작업의 수($i = 1, 2, \cdots\cdots, m$)
- d_i : i 기계에 할당된 작업들의 공통납기일
- σ_i : i 기계에 할당된 작업들의 임의의 한 작업순서
- K_i : i 기계에서 납기일 내에 실시해야 하는 작업의 수
- f_i : i 기계에서 납기일 이후에 실시되는 작업들의 총흐름시간
- $t_{i(j)}$: i 기계에서 j번째 위치에 있는 작업의 작업시간
- $C_{i(j)}$: i 기계에서 j번째 위치에 있는 작업이 완료된 후 납기일까지의 작업이 완료된 후 납기일까지의 시간, 즉 재고시간
- $T_{i(j)}$: i 기계에서 j번째 위치에 있는 작업의 지연시간
- $F_{i(j)}$: i 기계에서 j번째 위치에 있는 작업의 흐름시간
- p_1 : 현시점에서 공통납기일까지의 단위 시간당 납기비용
- p_2 : 단위 시간당 재고비용
- p_3 : 단위 시간당 지연비용
- p_4 : 단위 시간당 흐름시간비용
- $W_{i(j)}$: i 기계에서 j번째 위치에 있는 작업의 단위 시간당 비용

2.3 목적함수의 설정

m개의 기계중에서 임의의 한 기계 i에 대해서 고려해 보면 다음과 같다.

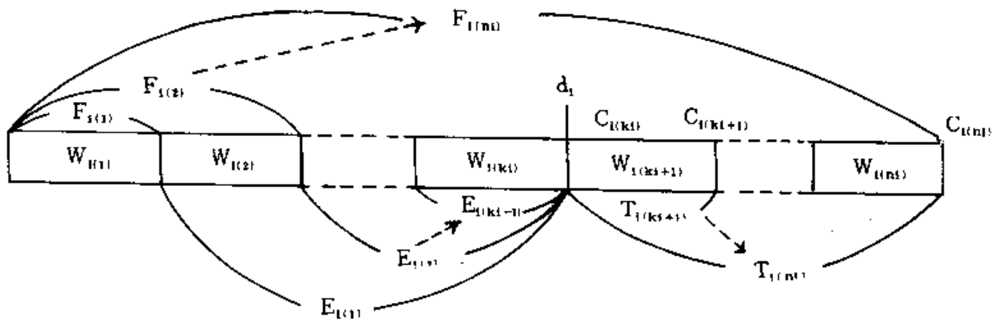


그림 1. 임의의 한기계에 대한 제한 조건

임의의 한 기계 i에 할당된 작업들의 공통 납기일과 전체 작업의 완료일은

$$d_i = \sum_{j=1}^{Ki} t_{i(j)} \quad \cdots\cdots (2.1)$$
$$f_i = d_i + \sum_{j=Ki+1}^{ni} t_{i(j)} \quad \cdots\cdots (2.2)$$

이고, d_i에 따라 i기계에서 위치에 따른 재고시간, 지연시간 및 흐름시간은 다음과 같다.

$$E_{i(j)} = \max(0,\ d_i - C_{i(j)}) \quad \cdots\cdots (2.3)$$
$$T_{i(j)} = \max(0,\ C_{i(j)} - d_i) \quad \cdots\cdots (2.4)$$
$$F_{i(j)} = t_{i(1)} + t_{i(2)} + \cdots\cdots + t_{i(j)} = C_{i(j)} \quad \cdots\cdots (2.5)$$

이에 따라 i기계에서의 임의의 한 작업순서 σ_i에 대한 총비용은

$$Z(d_i, \sigma_i) = \sum_{j=1}^{ni} (P_1 d_i + P_2 E_{i(j)} + P_3 T_{i(j)} + P_4 F_{i(j)}) \quad \cdots\cdots (2.6)$$

이고, 이식은 다음과 같이 나타낼 수 있다.

$$Z(d_i, \sigma_i) = \sum_{j=1}^{ni} W_{i(j)} t_{i(j)} \quad \cdots\cdots (2.7)$$

이같은 개념을 m개의 기계인 경우에 적용시켜 보면 마찬가지 결과를 얻을 수 있는데 임의의 한 작업순서 σ에 대해서

$$Z(D, \sigma) = \sum_{i=1}^{m} \sum_{j=1}^{ni} (P_1 d_i + P_2 d_{i(j)} + P_3 T_{i(j)} + P_4 F_{i(j)}) = \sum_{i=1}^{m} \sum_{j=1}^{ni} W_{i(j)} t_{i(j)} \quad \cdots\cdots (2.8)$$

이 된다. 따라서 구하고자 하는 목적함수는 다음과 같다.

$$\text{Minimize } Z(D, \sigma) \quad \cdots\cdots (2.9)$$
$$D = \max d_i,\ i = 1, 2, \cdots, m \quad \cdots (2.10)$$
$$F = D + \max f_i,\ i = 1, 2, \cdots, m \quad (2.11)$$

3. 최적작업순서의 결정

3.1 기초이론 및 결정방법

m개의 기계에 대해서 분석하기 전에 Seidmann [12]등에 의해서 연구된 바를 기초로 하여 임의의 한 기계 i에 대해서 분석해 보면 다음과 같다.

[결과1]

i기계에서의 최적납기일 d_i는

$$0 \leq d_i \leq C_{i(ni)} = \sum_{j=1}^{ni} t_{i(j)} \quad \cdots\cdots (3.1)$$

이고, $i \in m$ $j \in n_i$이다.

[결과2] 만약에 $P_1 \geq P_3$이면 $d_i = 0$이고, SPT법칙에 의한 작업순서가 최적이다.

[정리1] i기계에서 어떤 특정한 작업순서 σ_i에 대해서 최적 d_i값은 σ_i에 있는 어떤 한 작업의 완료시간과 일치한다.

[정리2] i 기계에서 어떤 작업순서 σ_i에 대해서 K_i가 $n_i(P_3-P_1)/(P_2+P_3)$의 값과 같거나 큰 최소 정수값일 때 K_i번째 작업의 완료시간 $C_{i(K_i)}$와 최적납기일 d_i는 일치한다.

이와 같은 결과들을 이용하면 임의의 어떤 작업순서에 대해서 K_i개의 작업들은 지연시간이 발생하지 않고, n_i-K_i개의 작업들은 재고시간이 발생하지 않음을 알 수 있다. 따라서 i 기계에서의 총비용은

$$Z(d_i,\sigma_i)=\sum_{j=1}^{n_i}(P_1d_i+P_2E_{i(j)}+P_3T_{i(j)}+P_4F_{i(j)})=\sum_{j=1}^{K_i}[n_iP_1+(j-1)P_2+(n_i+1-j)P_4]t_{i(j)}+\sum_{j=K_i+1}^{n_i}[(n_i+1-j)P_3+(n_i+1-j)P_4]t_{i(j)} \quad\cdots\cdots(3.2)$$

이고, 이식을 단위 시간당 비용으로 나타내면 다음과 같다.

$$Z(d_i,\sigma_i)=\sum_{j=1}^{n_i}W_{i(j)}t_{i(j)} \quad\cdots\cdots(3.3)$$

$$W_{i(j)}=\begin{cases}n_iP_1+(j-1)P_2+(n_i+1-j)P_4; & j\leq K_i \quad\cdots\cdots(3.4)\\(n_i+1-j)\ (P_3+P_4); & K_i+1\leq j\leq n_i\end{cases}$$

i 기계에서 최적작업순서를 찾기 위해서는 i 기계에 할당된 작업들의 가능한 모든 작업순서 σ_i중에서 최소 총비용을 갖는 작업순서를 결정해야 한다.

[정의1] $Z(d_i,\sigma_i)=\sum_{j=1}^{n_i}W_{i(j)}t_{i(j)}$ 이므로 가장 큰 w값을 가지는 위치에 작업시간 t가 가장 작은 작업을 할당하고, 그 다음 큰 w값을 가지는 위치에 그다음 작은 작업시간 t를 가지는 작업을 할당한다. 이와 같은 과정을 반복하여 i 기계에 있는 모든 작업을 할당하면 총비용을 최소화시키는 작업순서를 구할 수 있다.

[정리3] 만약에 $P_2<P_4$이면 $d_i=0$이고 SPT 법칙에 의한 작업순서가 최적이다.

지금까지는 임의의 한 기계 i에 대해서 분석해 보았는데 이와 같은 특성들을 이용하여 m개의 기계에 대해서 분석해 보자.

m개의 기계에 대한 총비용은

$$Z(D,\sigma)=\sum_{i=1}^{m}\sum_{j=1}^{n_i}W_{i(j)}t_{i(j)} \quad\cdots\cdots(3.5)$$

$$W_{i(j)}=\begin{cases}n_iP_1+(j-1)P_2+(n_i+1-j)P_4; & j\leq K_i \quad\cdots\cdots(3.6)\\(n_i+1-j)\ (P_3+P_4); & K_i+1\leq j\leq n_i\end{cases}$$

이므로 다음과 같은 정의를 내릴 수 있다.

[정의2] m개의 기계에 대한 총비용은 $Z(D,\sigma)=\sum_{i=1}^{m}\sum_{j=1}^{n_i}W_{i(j)}t_{i(j)}$ 이므로 가장 큰 w값을 가지는 위치에 작업시간 t가 가장 작은 작업을 할당하고, 그 다음 큰 w값을 가지는 위치에 그 다음 작은 작업시간 t를 가진 작업을 할당한다. 이와 같은 방법을 되풀이하여 작업을 할당하면 총비용을 최소화하는 최적 작업순서를 결정할 수 있다.

그런데 [정리4]에 의해서 작업을 할당하는데 있어서 각각 기계에 할당할 작업의 수를 결정해야 하는데 그 방법은 다음가 같다.

[정리4] 다수의 기계에 의해서 작업들이 실시되는 경우 총비용이 최소값을 가지기 위한 필요조건은 모든 한 쌍의 기계(i, k)에 대해서

$$|n_i-n_k|=1 \quad\text{-----------------}\ (3.7)$$

이다. 여기서 n_i, n_k는 기계 i, k에 할당되는 작업의 수이다.

(증명) 기계 i와 k에서 위치에 따른 단위비용은 다음과 같다.

$$W_{i(j)}=\begin{cases}n_iP_1+(j-1)P_2+(n_i+1-j)P_4; & j\leq K_i\\(n_i+1-j)\ (P_3+P_4); & K_i+1\leq j\leq n_i\end{cases}$$

$$W_{k(j)}=\begin{cases}n_kP_1+(j-1)P_2+(n_k+1-j)P_4; & j\leq K_k\\(n_k+1-j)\ (P_3+P_4); & K_k+1\leq j\leq n_k\end{cases}$$

여기서 $n_i \leq n_k+1$인 경우와 $n_i > n_k+1$인 경우를 비교해 보면 위치당 단위비용이 $n_i > n_k+1$인 경우가 크다. 그런데 총비용은 $Z=\sum_{i=1}^{m}\sum_{j=1}^{n_i} w_{i(j)}\, t_{i(j)}$ 이므로 각 작업의 작업시간은 변동이 없는데 단위비용이 커지므로 총비용은 증가한다. 그러므로 각 기계에 할당하는 작업의 수의 차가 적을수록 단위비용은 작아지고 그에 따라 총비용도 작아진다. 따라서 $|n_i - n_k| < 1$일때 총비용은 최소가 된다.

m개의 기계에서 각 기계에서의 위치에 따른 단위비용은 비용의 종류 P_1, P_2, P_3, P_4, 와 각 기계에 할당되는 작업의 수 n_i에 따라 여러 상황이 발생한다. $n_i - n_k = 1$인 임의의 한 쌍의 기계 i, k에서 주어진 비용에 따라 각 기계에서 납기일 이전에, 즉 K_i 이전과 K_k 이전에 각 기계의 다른 위치에서 단위비용이 같은 것이 발생하는 경우 혹은 전혀 다른 경우가 발생하고, n_i와 n_k에 따라 K_i와 K_k가 같은 경우 혹은 다른 경우가 발생한다.

먼저 처음의 경우에 대해서 고려해 보면 작업의 수가 n_i인 기계의 첫번째 위치와 작업의 수가 $n_k(=n_i-1)$인 기계의 r번째부터 같은 단위비용이 발생하는 경우이다. 따라서 r에 대해

$$n_i(p_1+p_4)=n_k(p_1+P_4)+(r-1)(p_2-P_4) \quad \text{-------------------------} \quad (3.8)$$

이 성립하고, 여기서 $n_k=n_i-1$ 이므로

$$r=(p_1+p_4)/(p_2-P_4)+1 \quad \text{-----} \quad (3.9)$$

이고, 단 r은 정수이고 $r \leq K_k$이다.

두번째 경우는 작업의 수가 다르면 같은 단위비용이 발생하지 않는 경우이다.

[방법1] 식 (3.9) 를 만족하는 경우이며,

표 1. 작업의 수가 달라도 같은 단위비용이 발생하는 경우

(i) $K_k=K_i-1$인 경우

	1	………	r	………	K_i-r	………	K_i-1	K_i	K_i+1	………	n_i-1	n_i
1	$W_{i(1)}$	………	$W_{i(r)}$	………	$W_{i(ki-r)}$	………	$W_{i(ki-1)}$	$W_{i(ki)}$	$W_{i(ki+1)}$	………	$W_{i(ni-1)}$	$W_{i(ni)}$
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
i	$W_{i(1)}$	………	$W_{i(r)}$	………	$W_{i(ki-r)}$	………	$W_{i(ki-1)}$	$W_{i(ki)}$	$W_{i(ki+1)}$	………	$W_{i(ni-1)}$	$W_{i(ni)}$
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
k	$W_{k(1)}$	………	$W_{i(1)}$	………	$W_{i(r)}$	………	$W_{i(ki-r)}$	$W_{i(ki+1)}$	$W_{i(ki+2)}$	………	$W_{i(ni)}$	
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	
m	$W_{k(1)}$	………	$W_{i(1)}$	………	$W_{i(r)}$	………	$W_{i(ki-r)}$	$W_{i(ki+1)}$	$W_{i(ki+2)}$	………	$W_{i(ni)}$	

(ii) $K_i=K_k$인 경우

	1	………	r	………	K_i-r+1	………	K_i-1	K_i	K_i+1	………	n_i-1	n_i
1	$W_{i(1)}$	………	$W_{i(r)}$	………	$W_{i(ki-r)}$	……	$W_{i(ki-1)}$	$W_{i(ki)}$	$W_{i(ki+1)}$	………	$W_{i(ni-1)}$	$W_{i(ni)}$
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
i	$W_{i(1)}$	………	$W_{i(r)}$	………	$W_{i(ki-r}$	………	$W_{i(ki-1)}$	$W_{i(ki)}$	$W_{i(ki+1)}$	………	$W_{i(ni-1)}$	$W_{i(ni)}$
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
k	$W_{k(1)}$	………	$W_{i(1)}$	………	$W_{i(r)}$	………	$W_{i(ki-r)}$	$W_{i(ki+r)}$	$W_{i(ki+2)}$	………	$W_{i(ni)}$	
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	
m	$W_{k(1)}$	………	$W_{i(1)}$	………	$W_{i(r)}$	………	$W_{i(ki-r)}$	$W_{i(ki+r)}$	$W_{i(ki+2)}$	………	$W_{i(ni)}$	

n_i, n_k에 따라 K_i와 K_k가 같은 경우 혹은 다른 경우가 있다.

먼저 K_i이전과 K_i이후에서 단위비용이 같은 것이 발생하면 납기일을 줄이기 위해 작업시간이 작은 작업을 K_i이전에 작업시간이 더 큰 작업을 K_i이후에 할당한다.

납기일 이전의 작업의 할당은 먼저 임의로 단위비용에 따라 작업을 할당한 후 같은 단위비용의 위치들에 대해서는 다시 조종한다. 단위비용이 다른 위치에 할당된 작업들의 각 기계에서의 작업시간의 누적치를 계산한 다음, 누적치를 비교하여 가장 큰 값에 가장 작은 작업시간을 가진 작업을 할당하고, 그 다음 큰 값에 그 다음 작은 시간을 가진 작업을 할당한다. 이와 같은 방법을 단위비용이 큰 위치에서부터 실시하면 된다.

납기일 이후의 작업의 할당은 단위비용이 큰 위치부터 할당하며 같은 단위비용의 위치에 대해서는 할당된 작업들의 작업시간의 누적차를 비교하여 그 값이 가장 큰 위치에 작업시간이 가장 작은 작업을 할당하고, 그 다음 큰 값에 그 다음 작은 작업시간을 가진 작업을 할당하는 방법을 반복하면 된다.

[방법2] 식 (3.9)를 만족하지 않는 경우이며, n_i, n_k에 따라 K_i와 K_k가 같은 경우 혹은 다른 경우가 발생한다.

납기일 이후의 작업의 할당은 **[방법1]** 과 같다. 납기일 이전의 경우에는 단위비용이 큰 위치부터 작업을 할당하는데 기계당 할당되는 작업의 수가 같으면 같은 위치에서는 단위비용이 같으므로 할당되는 작업의 수가 같은 기계끼리 할당된 작업의 작업시간의 수적치를 비교

표2. 작업의 수가 다르면 단위비용이 다른 경우

(i) $K_k = K_i - 1$인 경우

	1	·········	$K_i - 1$	K_i	$K_i + 1$	·········	$n_i - 1$	n_i
1	$W_{1(1)}$	·········	$W_{1(ki-1)}$	$W_{1(ki)}$	$W_{1(ki+1)}$	·········	$W_{1(ni-1)}$	$W_{1(ni)}$
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
i	$W_{i(1)}$	·········	$W_{i(ki-1)}$	$W_{i(ki)}$	$W_{i(ki+1)}$	·········	$W_{i(ni-1)}$	$W_{i(ni)}$
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
k	$W_{k(1)}$	·········	$W_{k(Kk)}$	$W_{i(ki+1)}$	$W_{i(ki+2)}$	············	$W_{i(ni)}$	
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	
m	$W_{k(1)}$	·········	$W_{k(Kk)}$	$W_{i(ki+1)}$	$W_{i(ki+2)}$	·········	$W_{i(ni)}$	

(ii) $K_k = K_i$인 경우

	⋮	·········	$K_i - 1$	K_i	$K_i + 1$	·········	$n_i - 1$	n_i
1	$W_{1(1)}$	·········	$W_{1(ki-1)}$	$W_{1(ki)}$	$W_{1(ki+1)}$	·········	$W_{1(ni-1)}$	$W_{1(ni)}$
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
i	$W_{i(1)}$	·········	$W_{i(ki-1)}$	$W_{i(ki)}$	$W_{i(ki+1)}$	·········	$W_{i(ni-1)}$	$W_{i(ni)}$
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
k	$W_{k(1)}$	·········	$W_{k(Kk-1)}$	$W_{k(Kk)}$	$W_{i(ki+1)}$	·········	$W_{i(ni)}$	
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	
m	$W_{k(1)}$	·········	$W_{k(Kk-1)}$	$W_{k(Kk)}$	$W_{i(ki+2)}$	·········	$W_{i(ni)}$	

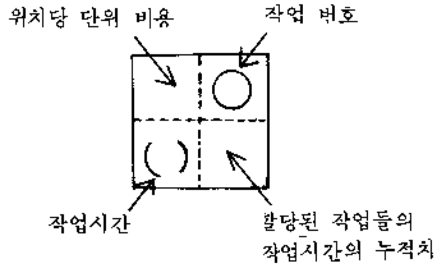
하여 같은 방법으로 할당한다. 이와 같은 방법을 적용하여 작업을 할당하면구하고자 하는 최적작업순서를 구할 수 있다.

[방법1] 과 **[방법2]** 의 차이점을 다음 예제를 통해 구분해 보고자 한다. 작업수는 18, 기계수는 4, 각 비용은 $P_1=1$, $P_2,=7$, $P_3=10$, $P_4=3$이고 각 작업의 작업시간은 다음과 같다.

작업번호	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
작업시간	12	6	27	3	2	19	27	25	5	22	20	16	13	12	10	7	8	20

$n_1=n_2=5$, $n_3=n_4=4$이고, $K_1=K_2=K_3=K_4=3$이다. 위치에 따른 단위비용 W는 다음과 같고, $r=(1+3)/(7-3)+1=2$이므로 **[방법1]** 에 의해서 작업을 할당한다.

작업의위치 / 기계	1	2	3	4	5
M_1	20 ⑭ (12)	24 ⑯ (7)	28 ⑤ (2)	26 ⑨ (5) 5	13 ⑧ (25) 30
M_2	20 ⑬ (13)	24 ⑰ (8)	28 ④ (3)	26 ② (6) 6	13 ⑩ (22) 28
M_3	16 ⑪ (20)	20 ⑫ (16)	24 ⑮ (10)	13 ③ (27) 27	
M_4	16 ⑱ (20)	20 ⑥ (19)	24 ① (12)	⑬ ⑦ (27) 27	



작업의 위치 / 기계	1	2	3	4	5
M_1	20 ⑭ (12)	24 ⑯ (7)	28 ⑤ (2)		
M_2	20 ⑬ (13)	24 ⑰ (8)	28 ④ (3)		
M_3	20 ⑫ (16)	24 ⑮ (10)	16 ⑪ (20)		
M_4	20 ⑥ (19)	24 ① (12)	16 ⑱ (20)		

작업의 위치 / 기계	1	2	3	4	5
M_1	20 ⑫ (16) 30	24 ① (12) 14	28 ⑤ (2) 2		
M_2	20 ⑥ (19) 32	24 ⑮ (10) 13	28 ④ (3) 3		
M_3	20 ⑬ (13) 40	24 ⑯ (7) 27	16 ⑪ (20) 20		
M_4	20 ⑭ (12) 40	24 ⑰ (8) 28	16 ⑱ (20) 20		

작업의 위치 / 기계	1	2	3	4	5
M_1	20 ⑫	24 ①	28 ⑤	26 ⑨	13 ⑧
M_2	20 ⑥	24 ⑮	28 ④	26 ②	13 ⑩
M_3	16 ⑪	20 ⑬	24 ⑯	13 ③	
M_4	16 ⑱	20 ⑭	24 ⑰	13 ⑦	

[방법2] 에 대한 예제는 제4장에서 제시하고 있다.

그런데 이런 과정을 거쳐 구한 결과는 부분적인 최적은 될 수 있지만 항상 모든 경우에 대해서 최적은 아니다. 왜냐하면 각 기계에서의 위치에 따른 단위비용은 그 기계에 할당될 작업의 수 n_i에 의해서 정해지는데 n_i가 클수록 W 값은 커지므로 [정의4] 에 의해서 단위비용이 큰 것부터 작업시간이 작은 작업을 할당하면 단위비용이 작은 것들만이 있는 기계에서는 즉 할당되는 작업의 수 n_i가 적은 기계에서는 작업시간이 비교적 긴 작업들이 할당된다. 그러다 보면 $D = \max_{1 \le i \le n} d_i$에서 D 값이 크게 된다. 따라서 D 값을 줄일 수 있다면 줄어드는 납기일에 대해 단위 시간당 nP_1 만큼 그 전의 모델보다 총비용이 상대적으로 감소된다고 판단할 수 있다. 다라서 작업시간이 0 (zero)인 가상작업을 임의로 만들어 작업의 수를 n +1로 만든다. 이렇게 하면 어떤 한 기계 k ($n_k = n_i - 1$)에서는 할당되는 작업의 수가 $n_k + 1 (= n_i)$로 되고 단위 시간당 비용은 증가하나 이로 인해 납기일의 감소를 꾀할 수 있다. 따라서 전의 모델에서의 총비용과 가상작업을 넣은 모델에서의 총비용에서 감소된 납기일에 대한 비용 절감 DnP_1만큼 뺀 값을 비교한 후 더 적은 총비용의 값을 가지는 모델을 선택한다.

이와 같은 과정은 모든 한쌍의 기계 i, k 에 대해서 $|n_i - n_k| = 0$인 경우까지 반복하여 비교해서 선택한 모델이 총비용을 최소화시키는 최적작업순서이며 그 때의 납기일 및 작업완료일이 최적값이 된다.

3.2 최적작업순서의 결정과정

단계1. n 개의 작업을 SPT순서로 배열한다.

단계2. n 개의 작업에 대해 m 개 기계 각각에 할당된 작업의 수 n_i를 계산한다.

단계3. $P_2 \geq P_4$이고 $P_1 < P_3$이면 단계4. 로 간다.

그렇지 않으면 모든 $K_i = 0$으로 하고 단계5. 로 간다.

단계4. 각 기계에서 납기일까지 수행해야 할 작업의 수 K_i를 계산한다.

$$K_i = [n_i (P_3 - P_1) / (P_2 + P_3)], \ [\ ] ; \text{Gauss 기호} \quad \cdots\cdots (3.10)$$

단계5. 각 기계에서의 위치에 따른 단위비용을 계산한다.

$$W_{i(j)} = \begin{cases} n_i P_1 + (j-1) P_2 + (n_i + 1 - j) P_4 ; \\ j \leq K_i \quad \cdots\cdots (3.11) \\ (n_i + 1 - j)(P_3 + P_4) ; K_i + 1 \leq j \leq n_i \end{cases}$$

단계6. 식(3.12)를 만족하면 [방법1] 에 의해서 작업들을 할당한다.

그렇지 않으면 [방법 2]에 의해서 작업들을 할당한다.

$r = (P_1 + P_4) / (P_2 - P_4) + 1$, r은 정수이고 $r \leq K_k$ ---------- (3.12)

단계7. 이 모델에 대한 총비용, 납기일 및 작업완료일을 계산한다.

$$WZ = \sum_{i=1}^{m} \sum_{j=1}^{n_i} W_{i(j)} t_{i(j)} \quad \cdots\cdots (3.13)$$

$$WD = \max_{1 \le i \le m} \left(\sum_{j=1}^{K_i} t_{i(j)} \right) \quad \cdots\cdots (3.14)$$

$$WF = WD + \max_{1 \le i \le m} \left(\sum_{K_i+1}^{n_i} t_{i(j)} \right) \quad \cdots\cdots (3.15)$$

단계8. 모델들의 총비용을 비교하여 더 작은 모델을 선택한다.

만약에 $Z > WZ - n(D - WD) P_1$이면 $Z = WZ$, $D = WD$, $F = WF$이다.

그렇지 않으면 $Z = Z$, $D = D$, $F = F$이다.

단계9. $n_1 \neq n_m$이면 t =0인 가상작업을 만들고 단계1. 로 간 다음 다시 반복한다.

그렇지 않으면 끝낸다.

4. 수학적인 예제

작업의 수는 19, 기계의 수는 4, 각 비용은

$P_1=5$, $P_2=8$, $P_3=20$, $P_4=4$이고 각 작업의 작업시간은 아래의 표와 같다.

작업번호	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
작업시간	12	6	27	3	2	19	27	25	5	22	20	16	13	12	10	7	29	20	8

단계1. 작업들을 SPT 순서로 배열한다.

작업번호	5	4	9	2	16	19	15	1	14	13	12	6	11	18	10	8	3	7	17
작업시간	2	3	5	6	7	8	10	12	12	13	16	19	20	20	22	25	27	27	29

단계2. $n_1=5$, $n_2=5$, $n_3=5$, $n_4=4$

단계3. $P_2 > P_4$, $P_1 < P_3$ ⇨ 단계4.

단계4. $K_1 = K_2 = K_3 = [5(20-5)/(8+20)] = 3$

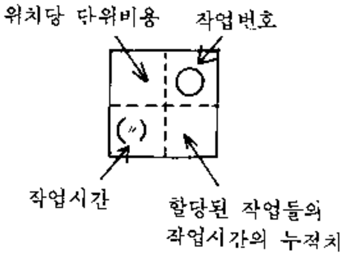
$K_4 = [4(2-5)/(8+20)] = 3$

단계5. 각 위치에 따른 단위비용 W값을 계산한다.

작업의위치 / 기계	1	2	3	4	5
M_1	45	49	53	48	24
M_2	45	49	53	48	24
M_3	45	49	53	48	24
M_4	36	40	44	24	

단계6. $r = (5+4)/(8-4) = 2.25$

따라서 r값이 정수가 아니고 식(3.12)를 만족하지 않으므로 **[방법2]** 에 의해서 작업을 할당하면 다음과 같은 결과를 얻을 수 있다.



작업의위치 / 기계	1		2		3		4		5	
M_1	45	⑥	49	⑲	53	⑤	48	⑮	24	⑦
	(20)	30	(8)	10	(2)	2	(10)	10	(27)	37
M_2	45	⑫	49	⑯	53	④	48	①	24	③
	(19)	29	(7)	10	(3)	3	(12)	12	(27)	39
M_3	45	⑬	49	②	53	⑨	48	⑭	24	⑧
	(13)	34	(6)	11	(5)	5	(12)	12	(25)	37
M_4	36	⑩	40	⑱	44	⑪	24	⑰		
	(22)	62	(20)	40	(20)	20	(29)	29		

단계7. $WZ = \sum_{i=1}^{4} \sum_{j=1}^{n_i} W_{i(j)} t_{i(j)} = 10,596$

$WD = \max(30, 29, 34, 62) = 62$

$WF = 62 + \max(37, 39, 37, 29) = 101$

단계9. $n_1 \neq n_4$이므로 $t_{20}=0$인 가상작업을 만들고 단계1. 로 간다.

단계1. ~단계6. 같은 방법으로 적용해 가면 다음과 같은 결과를 얻을 수 있다.

작업의위치 / 기계	1		2		3		4		5	
M_1	45	⑩	49	⑮	53	⑳	48	①	24	⑰
	(22)	32	(10)	10	10	(0)	0	(12)	12	(29)
M_2	45	⑱	49	⑲	53	⑤	48	⑭	24	⑦
	(20)	30	(8)	10	(2)	2	(12)	12	(27)	39
M_3	45	⑪	49	⑯	53	④	48	⑬	24	③
	(20)	30	(7)	10	(3)	3	(13)	13	(27)	40
M_4	45	⑥	49	②	53	⑨	48	⑫	24	⑧
	19	30	(6)	11	(5)	5	(16)	16	(25)	41

단계7. $WZ = 10,830$

$WD = \max(32, 30, 30, 30) = 32$

$WF = 32 + \max(41, 39, 40, 41) = 73$

단계8. $10,830-(62-32)\times5\times19=7,980<10,595$

$D=32$, $F=73$

단계9. $n_1=n_4$이므로 끝낸다.

이와 같은 과정을 통하여 최적작업순서는 두 번째 모델임을 알 수 있고, 이에 따라 최적납기일은 32, 전체 작업의 완료일은 73이 된다.

5. 결 론

본 연구는 다수의 기계들이 병렬로 배치되고, 납기일이 공통인 다수의 작업들이 수행되며, 납기일 결정비용, 재고비용, 지연비용 및 흐름시간비용이 발생할 경우에 총비용을 최소화하는 최적작업순서를 결정하고, 최적공통납기일과 전체 작업의 완료일을 산출하는 방법을 제시하고 있다. 이것은 작업의 수, 기계의 수 및 비용의 종류에 따라 발생하는 모든 경우를 고려하여 가장 일반적인 경우에 최적해를 구할 수 있는 방법이다.

그러나 본 연구는 모델설정에 있어서 몇가지 조건을 전제로 하고 있다. 따라서 다음과 같은 경우에 최적작업순서를 결정하는 데에는 적용이 곤란하므로 여기에 대해 더 많은 연구를 해볼 필요가 있다.

(1) 주어지는 비용이 선형이 아닌 경우

(2) 납기일을 주문자가 정해주는 경우

(3) 기계를 가동하는데 드는 비용을 고려하여 먼저 가동할 기계의 수를 결정할 필요가 있는 경우

위와 같은 연구를 진행하는데 본 연구가 도움이 되기를 바란다.

References

1. Baker, K. R., *Introduction to Sequencing and Scheduling,* John Wiley and Sons, New York, 1974.
2. Baker, K. R., "Sequencing Rules and Due Date Assignments in a Job Shop," *Manage. Sci.,* Vol. 30, No. 9, pp. 1093-1104, 1984.
3. Eilon, S. and I. G. Chowdhury, "Due Dates in Job Shop Scheduling," *Int. J. Prod. Res.,* Vol. 14, pp. 223-237, 1976.
4. Eilon, S. and I. G. Chowdhury, "Minimizing Waiting Time Variance in the Single Machine Problem," *Manage. Sci.,* Vol. 23, No. 6, pp. 567-575, 1977.
5. Merten, A. G. and M. E. Muller, "Variance Minimization in Single Machine Sequencing Problems," *Manage. Sci.,* Vol. 18, No. 9, pp. 518-528, 1972.
6. Moore, J. E., "Sequencing in Jobs in One Machine to Minimize the Number of Tardy Job," *Manage. Sci.,* Vol. 14, No. 2, P. 102, 1968.
7. Rinooy Kan, A. H. G., B. J. Lagewey, and J. K. Lensta, "Minimizing Total Cost in One Machine Scheduling," *Ops. Res.,* Vol. 23, No. 5, P. 908, 1975.
8. Kanet, J. J., "Minimizing the Average Deviation of Job Completion Times about a Common Due Date," *Nav. Res. Logist. Q.,* Vol. 28, No. 4, pp. 643-651, 1981.
9. Schrage, L., "Minimizing the Time-in-System Variance for a Finite Jobset," *Manage. Sci.,* Vol. 21, No. 5, pp. 540-544, 1975.
10. Seidmann, A. and M. L. Smith, "Due Date Assignment for Production System," *Manage. Sci.,* Vol. 27, No. 5, pp. 571-580, 1981.

11. Seidmann, A., S. S. Panwalkar, and M. L. Smith, "Optimal Assignment of Due Dates for a Single Processor Scheduling Problem," *Int. J. Prod. Res.*, Vol. 19, No. 4, pp. 393-399, 1981.
12. Seidmann, A., S. S. Panwalkay, and M. L. Smith, "Common Due Date Assignment to Minimize Total Penalty for the One Machine Scheduling Problem," *Ops. Res.*, Vol. 30, No. 2, pp. 391-399, 1982.
13. Smith, W. E., "Various Optimizers for Single Stages Production," *Nav. Res. Logist. Q.*, Vol. 3. No. 1, pp. 56-60, 1956.
14. Sidney, J. B., "Optimal Single Machine Scheduling with Earliness and Tardiness Penalties," *Ops. Res.*, Vol. 25, No. 1, pp. 63-69, 1977.
15. Sundararaphavan, P. S. and M. U. Ahmed, "Minimizing the Sum of Absolute Lateness in Single Machine and Multimachine Scheduling," *Nav. Res. Logist. Q.*, Vol. 31, No. 2, pp. 325-333, 1984.
16. Weeks, J. K., "A Simulation Study of Productable Due Date," *Manage. Sci.*, Vol. 25, No. 3, pp. 363-373, 1979.
17. Weeks, J. K. and J. S. Fryer, "A Methodology for Assigning Minimum Cost Due Dates," *Manage. Sci.*, Vol. 23, No. 8, pp. 872-881, 1977.