〈研究論文〉

Journal of the Korean Institute of Industrial Engineers
Vol. 12, No. 2, Dec. 1986

準備時間이 종속인 多段階 生産시스템의 集団日程計劃

(Group Scheduilng Problem in Multi - Stage Manufacturing Systems with Dependent Setup time)

魯 仁 珪*
남 회 열*

Abstract

This research is concerned with group scheduling problems in multi-stage manufacturing system with dependent setup time. The objective of the research is to develop and evaluate a heuristic algorithm for determining group sequence and job sequence within each group to minimize total tardiness in multi-stage manufacturing systems with sequence dependent group setup time.

The group scheduling heuristic algorithm is developed and evaluated by comparisons with twenty-seven problems with the known optimum solutions and 144,000 random schedules of a large variety problems. The results indicate that the proposed heuristic algorithm gets the same optimum solutions for the problems and also provides the good solutions in comparison with the random schedules of the large variety problems. A numerical example is given to illustrate the heuristic algorithm.

1. 서 론

본 연구의 목적은 다단계 집단일정계획 문제에서 각 그룹에 대한 준비시간이 그룹순서에 종속인 경우 순수납기지연의 합을 최소화하는 발견적 기법을 개발하고 제시된 발견적 기법의 유효성을 평가하는데 있다.

그룹 스케쥴링 (Group Scheduling)은 대량 생산에서 얻을 수 있는 생산의 효율성을 다품종소량 생산에서도 얻을 수 있다는데 그 잇점이 있다. 그룹 스케쥴링은 그룹 테크놀러지 (Group Technology, GT)의 개념을 이용하여 부품을 그룹별로 분류하고 각 그룹내의 부품에 대하여 일정한 생산방식을 적용하여 생산을 행하는 것으로 일반적인 스케쥴링 문제와는 달리 다음과 같은 특성을 갖는다[2].

(1) 가공준비시간(Setup time)이 감소된다. 유사한 부품들이 그룹화되어 가공되므로 각 부품 가공의 준비시간은 감소되거나 제거될 수 있고 단지 각 그룹에 대한 준비시간만 소요된다.

(2) 그룹 스케쥴링에서의 job은 흐름생산 (Flow shop)의 형태를 갖는다.

(3) 수행할 작업의 순서를 정할 때, 그룹간 순서와 각 그룹안에 있는 job의 순서를 정해야 한다.

1968년 Petrov[9]가 그룹 스케쥴링의 이론

*漢陽大學校 工科大學 産業工學科

적 해석 및 다단계 생산 시스템 (System)에서의 총처리 사간(Makespan)을 줄이는 발견적 기법 (Heuristic Algorithm)을 개발한 이래 많은 연구가 단일공정과 다단계 공정에 대해 발표되었다.

Yoshida, Nakamura와 Hitomi[11]는 단일공정에서 총 소요시간, 평균처리시간(Mean Flow Time), 순수납기지연의 합(Total Tardiness)을 최소화하기 위한 최적화 기법을 개발하였고, Hitomi와 함[6]은 다단계 공정에서 총 소요시간을 최소화하기 위한 최적화 기법을 개발하였다. 그리고 Nakamura와 Hitomi[8]는 다단계 공정에서 총 소요시간, 평균처리시간 그리고 순수납기지연의 합에 대한 최적화 기법을 개발하였다.

그러나 이상의 연구들은 분지한계법(Branch and Bound Method)을 이용한 최적화 기법들로 많은 계산시간과 노력이 요구되며 그룹간 준비시간이 그룹순서와 독립이라는 가정하에 이루어진 것으로써 실제 상황에 적용이 어렵다.

이와 같은 문제를 해결하기 위하여 Yoshida와 Hitomi[10]는 그룹의 준비시간이 그룹순서에 종속인 경우 단일공정에 대한 각 평가기준에서의 최적화 기법을 분지한계법을 이용하여 개발하였다. Foo와 Wagner[5]는 단일공정에서 종속적인 준비시간을 갖는 문제를 총 소요시간에 대해 연구하였다. 조[1]는 순수납기지연의 합을 최소로 하는 문제에 대한 발견적 기법을 그룹준비시간이 그룹순서에 독립적인 경우에 대해 제시하였다.

본 연구에서는 다단계 공정에 있어서 각 그룹의 준비시간은 그룹간 순서에 따라 상이하다는 일반적 전제 하에서, 지금까지 시도되지 않았던 가공에 있어서의 순수납기지연의 합을 최소화 하기 위한 새로운 발견적 기법의 개발을 시도한다.

본 연구와 동일한 모델에 대해 최적해(Optimum solution)를 구하는 기법이나 발견적 기법이 아직 제시되어 있지 않으므로 본 연구에서 제시한 발견적 기법의 타당성을 평가하기 위해 수치예를 제시하고, 최적해를 알고 있는 문제의 해와 랜덤 스케쥴로 얻은 해를 제시한 발견적 기법으로 얻은 해와 비교한다.

2. 그룹스케쥴링 모델

2.1 전제조건

다단계 집단일정계획에서의 그룹의 준비시간이 그룹순서에 종속적인 경우, 순수 납기지연의 합을 최소화하는 문제에 대한 수리모델의 설정을 위해 다음과 같은 전제조건들을 제시한다.

(1) 총 N개의 부품 또는 job이 그룹 테크놀러지의 개념에 의해 M개의 그룹으로 분류된다(단 $M < N$).

(2) 생산공정은 K대의 기계로 구성되며 기계의 고장은 없다고 가정한다.

(3) 가공 중인 job은 가공이 끝날 때까지 기계에서 제거되지 않는다(No Preemption).

(4) 모든 작업은 시간 0에서 시작이 가능하다.

(5) 모든 기계에서 그룹순서 및 그룹 내 부품 가공순서는 동일하고, 한 그룹내의 모든 가공이 완료된 후 다음 순서의 그룹의 가공을 시작할 수 있다.

(6) 각 기계에서 그룹준비시간(Group Setup time) 및 각 부품의 가공시간(Processing time) 그리고 납기(Due Date)가 주어져 있다(Table 1).

(7) 그룹준비시간은 그룹순서에 대해 종속적(Dependent)이다(Table 2).

TABLE 1. REPRESENTATION OF PROCESSING TIME & DUE DATE

GROUP		1					2					…………	M				
JOB		J_{11}	J_{12}	J_{13}	……	J_{1n_1}	J_{21}	J_{22}	J_{23}	……	J_{2n_2}	…………	J_{M1}	J_{M2}	J_{M3}	……	J_{Mn_M}
PROCESS-TIME		P^k_{11}	P^k_{12}	P^k_{13}	……	$P^k_{1n_1}$	P^k_{21}	P^k_{22}	P^k_{23}	……	$P^k_{2n_2}$	…………	P^k_{M1}	P^k_{M2}	P^k_{M3}	……	$P^k_{Mn_M}$
STAGE	1	P^1_{11}	P^1_{12}	P^1_{13}	……	$P^1_{1n_1}$	P^1_{21}	P^1_{22}	P^1_{23}	……	$P^1_{2n_2}$	…………	P^1_{M1}	P^1_{M2}	P^1_{M3}	……	$P^1_{Mn_M}$
	2	P^2_{11}	P^2_{12}	P^2_{13}	……	$P^2_{1n_1}$	P^2_{21}	P^2_{22}	P^2_{23}	……	$P^2_{2n_2}$	…………	P^2_{M1}	P^2_{M2}	P^2_{M3}	……	$P^2_{Mn_M}$
	3	P^3_{11}	P^3_{12}	P^3_{13}	……	$P^3_{1n_1}$	P^3_{21}	P^3_{22}	P^3_{23}	……	$P^3_{2n_2}$	…………	P^3_{M1}	P^3_{M2}	P^3_{M3}	……	$P^3_{Mn_M}$
	⋮	⋮	⋮	⋮	……	⋮	⋮	⋮	⋮	……	⋮	…………	⋮	⋮	⋮	⋮	⋮
	K	P^K_{11}	P^K_{12}	P^K_{13}	……	$P^K_{1n_1}$	P^K_{21}	P^K_{22}	P^K_{23}	……	$P^K_{2n_2}$	…………	P^K_{M1}	P^K_{M2}	P^K_{M3}	……	$P^K_{Mn_M}$
DUE DATE		d_{11}	d_{12}	d_{13}	……	d_{1n_1}	d_{21}	d_{22}	d_{23}	……	d_{2n_2}	……	d_{M1}	d_{M2}	d_{M3}	……	d_{Mn_M}

주 : (1) 전체 job 수 $N=\sum_{i=1}^{M} n_i$, 단 r_i 는 그룹 i 에 속하는 job 의 수

(2) P^k_{ij} : 기계 k 에서 J_{ij} 의 가공시간

(2) d_{ij} : 그룹 i 내의 job j 의 납기

TABLE 2. GROUP SETUP TIME

GROUP STAGE	1				2				………	K			
r (FROM) i (TO)	1	2	………	M	1	2	………	M	………	1	2	………	M
1	S^1_{11}	S^1_{12}	………	S^1_{1M}	S^2_{11}	S^2_{12}	………	S^2_{1M}	………	S^K_{11}	S^K_{12}	………	S^K_{1M}
2	S^1_{21}	S^1_{22}	………	S^1_{2M}	S^2_{21}	S^2_{22}	………	S^2_{2M}	………	S^K_{21}	S^K_{22}	………	S^K_{2M}
⋮	⋮	⋮	………	⋮	⋮	⋮	………	⋮	………	⋮	⋮	………	⋮
M	S^1_{M2}	S^1_{M2}	………	S^1_{MM}	S^2_{M1}	S^2_{M2}	………	S^2_{MM}	………	S^K_{M1}	S^K_{M2}	………	S^K_{MM}

주 : (1) S^k_{ri} : 기계 k 에서 그룹 r 다음에 그룹 i 가 올 때의 그룹준비시간

(2) S^k_{jj} : 기계 k 에서 그룹 j 가 처음에 올 때, 그룹 G_i 의 준비시간

2.2 문제의 수학적 모형

2.2.1 부호 설명

i : 그룹 인덱스(index) ($i=1,2,\cdots\cdots,M$)

j : Job 또는 부품가공의 인덱스($j=1,2,\cdots\cdots,n_i$)

k : 공정 또는 기계의 인덱스($k=1,2,\cdots,K$)

M_k : k 번 기계 또는 공정

G_i : 그룹 i

J_{ij} : 그룹 i 의 Job j

P^k_{ij} : 기계 k에서 J_{ij} 의 가공시간

S^k_{ri} : 기계 k에서 그룹 r 이후 그룹 i 가 올 때 그룹 i 의 준비시간

C^k_{ij} : 기계 k에서 J_{ij} 의 가공완료시간

Q^k_i : 그룹 i 의 기계 k 에서의 그룹가공시간

d_{ij} : J_{ij} 의 납기

( ) : 그룹과 Job 의 위차를 나타내기 위해 사용하는 부호

2.2.2 수리모델의 설정

기계 k에서 Job $J_{(i)(j)}$ 의 가공완료시간 $C^k_{(i)(j)}$ 는 다음 식 (1)과 같다.

$$C^k_{(i)(j)}=\max(C^k_{(i)(j-1)}, C^{k-1}_{(i)(j)})+P^k_{(i)(j)} \cdots\cdots\cdots\cdots (1)$$

마지막 기계 K (Capital K) 에서 Job $J_{(i)(j)}$ 의 가공완료 시간은

$C^k_{(i),(j)} = \max(C^k_{(i),(j-1)},\ C^{k-1}_{(i),(j)}) + P^k_{(i),(j)}$ ……… (2)

로 주어진다.

다단계 공정인 경우 일반적으로 Job의 가공 시간과 각 공정의 그룹준비 시간과의 관계에 따라서 Job의 대기상태 또는 기계의 유휴상태가 생긴다. 이 둘은 양자간의 공통인 경우를 제외하고는 배반적인 관계가 존재한다.

제 K단계 공정의 그룹 $G_{(i)}$에 속한 Job $J_{(i),(j)}$에 대하여 대기상태가 발생 할 것인지 기계의 유휴상태가 발생할 것인지의 여부는 다음에 주어진 식(3)의 편차 $\delta^k_{(i),(j)}$의 부호에 의해서 정해진다.

$\delta^k_{(i),(j)} = C^{k-1}_{(i),(j)} - C^k_{(i),(j-1)}$ ……………………………… (3)

Job의 대기시간을 $W^k_{(i),(j)}$, 기계의 유휴시간을 $Z^k_{(i),(j)}$ 라고 하면 다음과 같은 관계를 알 수 있다.

i) $\delta^k_{(i),(j)} > 0$ 인 경우, 기계의 유휴상태가 발생한다(Figure 1).

$W^k_{(i),(j)} = 0$　　　　$Z^k_{(i),(j)} = \delta^k_{(i),(j)}$

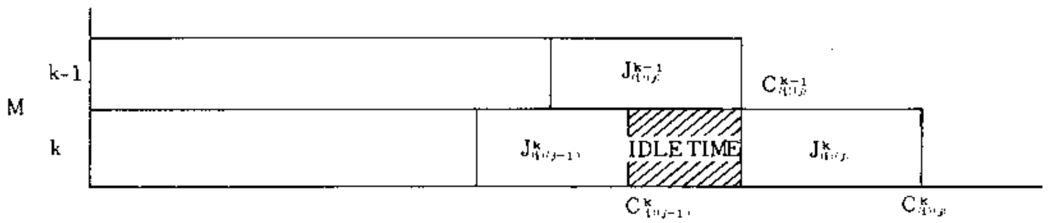

Figure 1. Case of Machine Idle time

ii) $\delta^k_{(i),(j)} < 0$ 인 경우, job의 대기상태가 발생한다(Figure 2).

$W^k_{(i),(j)} = -\delta^k_{(i),(j)}$,　　　　$Z^k_{(i),(j)} = 0$

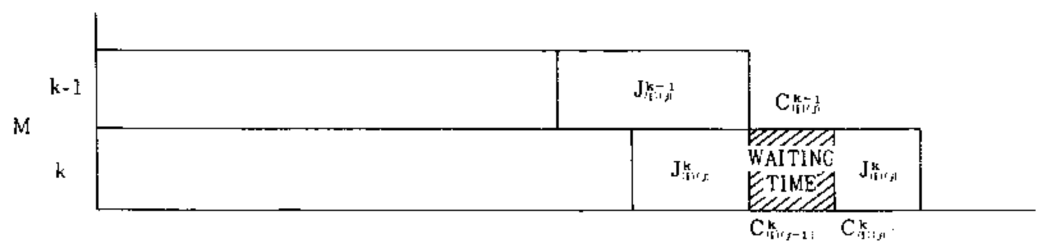

Figure 2. Case of Job Waiting time

iii) $\delta^k_{(i),(j)} = 0$ 인 경우, $W^k_{(i),(j)} = 0$, $Z^k_{(i),(j)} = 0$ 이다.

여기서 기계의 유휴(Idle)상태에 대해 살펴보면 식(4), (5)와 같다.

j=1인 경우

$$Z^k_{(i),(1)} = \begin{cases} C^{k-1}_{(i),(1)} - C^k_{(i-1),(n_{i-1})} - S^k_{(i-1),(i)} \\ \qquad ,C^{k-1}_{(i),(1)} > C^k_{(i-1),(n_{i-1})} + S^k_{(i-1),(i)} \cdots (4) \\ 0,\ \text{나머지 경우} \end{cases}$$

j≠1인 경우

$$Z^k_{(i),(j)} = \begin{cases} C^{k-1}_{(i),(j)} - C^k_{(i),(j-1)} \\ \qquad ,C^{k-1}_{(i),(j)} > C^k_{(i),(j-1)} \cdots\cdots (5) \\ 0,\ \text{나머지 경우} \end{cases}$$

이상의 수식을 이용하여 마지막 단계에서 각 그룹내의 각 job의 가공 완료시간을 계산할 수 있다. 그룹 i의 j번째 처리되는 job의 완료시간을 $C_{(i),(j)}$ 라고 하면 K번째 기계에

대한 $C^K_{(i),(j)}$는 다음 식(6)과 같다.

$$C^K_{(i),(j)} = \sum_{u=1}^{i-1} \left(\sum_{j=1}^{n_u} Z^K_{(u),(j)} + Q^K_{(u)}\right) + S^K_{(i-1),(i)} + \sum_{v=1}^{j} \left(Z^K_{(i),(v)} + P^K_{(i),(v)}\right) \quad \cdots\cdots (6)$$

그룹 i의 job j의 순수납기지연 $T_{(i),(j)}$는 식 (7)과 같이 쓸 수 있다.

$$T_{(i),(j)} = \max(0,\ C^K_{(i),(j)} - d_{(i),(j)}) \quad \cdots\cdots (7)$$

본 연구의 목적은 모든 job의 순수납기지연의 합을 최소화하는 것이므로 앞에서 제시한 부호 및 수식들을 이용하여 주어진 문제의 목적함수를 도출하면 식(8)과 같다.

$$\text{Minimize } T = \min_{S \in S}\left\{\sum_{i=1}^{M} \sum_{j=1}^{n_i} \max(0,\ C^K_{(i),(j)} - d_{(i),(j)})\right\} \quad \cdots\cdots (8)$$

단 s는 T를 최소화 하는 그룹 스케쥴이고, S는 $M!\,\pi^{M}_{i=1}(n_i!)$ 개의 그룹스케쥴의 집합니다.

3. 발견적 스케줄링 기법

3.1 기법의 유도

그룹 스케쥴링에서는 각 그룹의 처리순서(그룹간 순서)와 각 그룹에 속하는 job의 처리순서(그룹내 순서)를 합리적으로 결정하는 것이 문제이다. 본 연구에서는 평가기준이 순수납기지연의 합이므로 평균처리시간이나 총처리시간을 평가기준으로 하는 문제와는 달리 작업완료 시간과 납기와의 관계에 따라 작업순서를 정해야 한다.

일반적으로 단일공정에서는 그룹내 순서가 그룹간 순서에 영향을 받지 않는다. 그러나 다단계 공정에서는 그룹내 순서가 그룹간 순서에 영향을 받고 있으며 특히 순수납기지연의 합을 최소화 하는 문제에서는 앞의 그룹이나 job에서 발생하는 납기지연이 뒤에 오는 모든 그룹 또는 job의 납기지연에 큰 영향을 미치므로 그룹내 순서와 그룹간 순서를 독립으로 취급할 수 없다.

따라서 본 연구에서는 이러한 영향을 고려하여 그룹간 순서와 그룹내 순서가 서로 관련이 있으며, 그룹간 순서가 그룹내 순서에 영향을 미친다는 가정하에서 그룹간 순서에 따라 그룹내 순서를, 또한 그룹내 순서를 이용하여 그룹간 순서를 정하는 방법을 모든 그룹에 대하여 반복 적용한다.

M개의 그룹 중에서 이미 순서가 정해진 그룹들로 구성된 집합을 S라고 하고 순서가 정해지지 않은 그룹들의 집합을 S′이라고 하자. 또한 G를 $G = \{G_1,\ G_2 \cdots\cdots G_M\}$로 정의된 M개의 그룹의 집합이라고 하고 S에 속하는 그룹의 수를 [S], S′에 속하는 그룹의 수를 [S′]이라고 하면 $S \cup S' = G$, $[S] + [S'] = M$이다.

C_L를 S에 속하는 마지막 그룹 그룹 G_L의 가공완료시간(Completion time)이라고 하면 C_L는 식(9)와 같다.

$$C_L = \sum_{\substack{l=1 \\ l \in S}}^{L} (Z^K_{(l)} + Q^K_{(l)}) \cdots \quad \cdots\cdots (9)$$

여기서 $Z^K_{(l)}$는 기계 K에서 그룹(l)의 모든 기계의 유휴시간의 합계를 의미한다.

S 집합 내의 마지막 그룹을 G_L이라고 하고, S′에 속하는 임의의 그룹 i를 S의 끝에 첨가 했을 때, 이를 Si라고 표시하면 이 때 $J_{(i),(j)}$의 가공완료시간은 식(10)으로 나타낼 수 있다.

$$C^K_{(i),(j)} = C_L + S^K_{(L),(i)} + \sum_{m=1}^{j} (Z^K_{(i),(m)} + P^K_{(i),(m)}) \cdots (10)$$

그룹 i에 속하는 n_i개의 job가운데 이미 순서가 부여된 job을 순서대로 배치한 집합을 Job의 집합을 A_i라고 하고, A_i와 B_i에 속한 Bi, 그룹 i에서 아직 순서가 부여되지 않은 job의 수를 각각 [Ai], [Bi]라고 하면 $[Ai] + [Bi] = n_i$이다.

각 그룹에서 job순서를 결정하기 위하여 $J_{i,j}$의 순수납기지연의 상대적인 평가척도를 다음 식(11)과 정의한다.

$$\triangle i, j = d_{i,j} - \widetilde{C}_{i,j} \quad \cdots\cdots (11)$$

여기서 $\widetilde{C}_{i,j}$는 $J_{i,j}$의 가공완료시간의 가상치로 Si인 그룹 i내의 Job j를 임의의 순서에 둔다고 가정할 때의 가공완료시간이다. S내의 그룹에 대해서는 순서대로 준비시간을 부여하고, 그룹 i에 대해서는 마지막 그룹이 G_L 이므로 준비시간을 S^{*}_{Li}로 부여한다.

$d_{i,j} - \widetilde{C}_{i,j}$는 임의의 job이 (i), (j)에 있을 때 $J_{(i),(j)}$가 납기지연이 생기기까지의 시간적인 여유를 나타낸다. 따라서 납기지연을 발생시키지 않으려면 △i, j가 작은 job일 수록 작업 순서상 먼저 처리해야 한다.

△i, j의 부호에 따라 각 job은 다음과 같은 3가지의 경우 중, 한 가지 상태가 된다.

(i) △i, j =0이면 현재 (i), (j)에 있는 job의 가공완료시간과 납기가 같으며 이 job을 (i), (j) 이후에 처리하면 납기지연이 발생하는 경우이다.

(ii) △i, j >0이면 job $J_{(i),(j)}$가 (i), (j)의 위치에서는 납기까지 다소 여유가 있는 경우이다.

(iii) △i, j <0이면 현재 (i), (j)에 있는 job $J_{(i),(j)}$의 가공완료시간이 납기보다 크므로 납기지연이 발생하는 경우이다. 이런 job은 가능한 앞으로 보내어 납기지연을 방지하여야 한다.

주어진 문제의 그룹순서를 결정하기 위해서 Gi의 순수납기지연에 관계된 상대적인 평가척도를 △i로 표시하고 다음 식(12)와 같이 정의한다.

$$\triangle i = \sum_{j=1}^{n_2} \triangle' i, j / n_i \quad \cdots\cdots (12)$$

△'i, j는 그룹내 순서가 정해진 후 정해진 순서에 따라 job을 재배열 한 후, 각 job에 대한 가공완료시간과 납기간의 차 $d'_{i,j} - C'_{i,j}$를 나타낸다. 그룹 i내의 모든 job에 대한 △'i, j를 평균하여 그룹에 대한 척도로 만들어 줌으로써 그룹의 순서를 결정할 수 있다.

3.2 알고리즘 (Algorithm)

앞에서 고찰한 수리적 모델과 부호를 이용하여 순수납기지연의 합을 최소화 하는 발견적 기법을 제안한다.

단계1) S =0, S' = {G_1, G_2, ……… G_M}

(그룹내 순서의 결정)

단계2) S'에 속하는 그룹 i에 대해 Bi =0, Ai = {1, 2, 3……… n_i}로 둔다.

단계3) [Bi] < n_i −1이면 단계4로 간다. 그렇지 않으면 마지막 job을 n_i번째 위치시키고 단계6으로 간다.

단계4) Ai에 속하는 순서가 정해지지 않은 모든 job에 대하여 각 job이 ([S] + 1), ([Bi] + 1) 위치에 있다고 가정하고 △i, j를 계산한다.

단계5) △i, j가 최소인 job

$$\triangle\ ([S]+1),\ ([B_i]+1)^{*} = \min_{j \in A_i} (\triangle i, j)$$

을 찾아서 [Bi] +1번째 두고 $J_{i,j}$를 Ai에서 제외시킨다. 그리고 △i, j가 같을 때는 납기가 적은 job을 택한다. 단계3으로 가서 반복한다.

단계6) S'에 속하는 모든 그룹에 대해 단계 2 ~5를 반복한다.

(그룹간 순서의 결정)

단계7) 각 그룹 마다 정해진 순서에 따라 job을 재 배열한다.

단계8) [S'] =1이면 단계2~ 단계5 수행후 단계 11로 간다. 아니면 단계9로 간다.

단계9) S'에 속하는 각 그룹에 대해서 단계7에서 재 배열한 순서에 따라 △'i, j와 △i를 계산한다.

단계10) 최소의 △i를 갖는 그룹 Gi를 찾아서 그 그룹을 [S] +1번째 그룹순서에 위치시킨다. △i의 값이 같은 경우에는 그룹 내 job수가 적은 것을 택한다. 그룹내 job수가 같은 경우에는 납

기의 합이 적은 그룹을 택한다.

Gi를 S'에서 제외하고 단계2로 간다.

단계11) 순수납기지연의 합을 계산하고 과정을 정지한다.

3.3 수치예제

3단계 공정을 거치는 3개의 그룹 G_1, G_2, G_3에 대하여 Table3과 같이 job의 가공시간과 납기가 주어져 있고 각 공정마다 그룹간 순서에 따라 달라지는 준비시간이 Table4와 같이 주어져 있다. 이 두 데이타들을 이용하여 제시한 알고리즘의 적용예를 보인다.

Table 3. Processing time and Due Date

Group		Group 1		Group 2			Group 3		
Job		Job 11	Job 12	Job 21	Job 22	Job 23	Job 31	Job 32	Job 33
Proc. time		P_{11}	P_{12}	P_{21}	P_{22}	P_{23}	P_{31}	P_{32}	P_{33}
Stage	1	4	6	2	4	5	2	3	9
	2	5	2	4	3	8	5	8	2
	3	1	8	1	6	2	7	4	6
Due Date		55	68	10	27	21	46	48	37

Table 4. Group Setup time

Gr. \ St.	Machine 1			Machine 2			Machine 3		
r \ i	G_1	G_2	G_3	G_1	G_2	G_3	G_1	G_2	G_3
G_1	6	2	5	8	7	3	8	4	1
G_2	5	3	6	2	1	5	2	2	1
G_3	4	4	10	5	10	1	2	6	4

Iteration 1

단계1) $S=0$, $S'=\{G_1, G_2, G_3\}$

단계2) $Bi=0$, $Ai=\{1, 2, 3 \cdots\cdots n_i\}$에서 먼저 그룹 1에 대해서 알고리즘을 적용한다.

$B_1=0$, $A_1=\{1, 2\}$

단계3) $[B_1]=0<1$이므로 단계4로 간다.

단계4) $\triangle_{i,j}$값을 계산한다.

$\widetilde{C}_{1,1}=16$, $d_{1,1}=55$: $\triangle 1, 1=39$

$\widetilde{C}_{1,2}=22$, $d_{1,2}=68$: $\triangle 1, 2=46$

단계5) $\triangle 1, 1^*=39(=\triangle 1, 1)$

$B_1=\{1\}$, $A_1=\{2\}$, 단계3으로 간다.

단계3) $[B_1]=1$이므로 $B_1=\{1, 2\}$

그룹2와 그룹3에 대해서도 단계2−단계5를 반복하여 그룹내 순서를 정한다.

그룹2 : Job 2,1) $\widetilde{C}_{2,1}=10, d_{2,1}=10$ $\triangle 2,1=0$

Job 2,2) $\widetilde{C}_{2,2}=16, d_{2,2}=27$ $\triangle 2,2=11$

Job 2,3) $\widetilde{C}_{2,3}=18, d_{2,3}=21$ $\triangle 2,3=3$

$B_2=\{1\}$

Job 2,2) $\widetilde{C}_{2,2}=18, d_{2,2}=27$ $\triangle 2,2=9$

Job 2,3) $\widetilde{C}_{2,3}=20, d_{2,3}=21$ $\triangle 2,3=1$

$B_2=\{1, 3\}$

$B_2=\{1, 3, 2\}$

그룹3 : Job 3,1) $\widetilde{C}_{3,1}=24, d_{3,1}=44$ $\triangle 3,1=20$

Job 3,2) $\widetilde{C}_{3,2}=25, d_{3,2}=48$ $\triangle 3,2=23$

Job 3,3) $\widetilde{C}_{3,3}=27, d_{3,3}=37$ $\triangle 3,3=10$

$B_3=\{3\}$

Job 3,1) $\widetilde{C}_{3,1}=34, d_{3,1}=44$ $\triangle 3,1=10$

Job 3,2) $\widetilde{C}_{3,2}=34, d_{3,2}=48$ $\triangle 3,2=14$

$B_3=\{3, 1\}$

$B_3=\{3, 1, 2\}$

이상에서 정한 그룹내 순서로 그룹간 순서를 정한다.

단계7) 각 그룹마다 정해진 순서에 따라 job을 재 배열한다.

단계8) $[S'] \neq 1$이므로 단계9로 간다.

단계9) Group 1) $\triangle' 1, 1=39$, $\triangle' 1, 2=42$

$\triangle 1=40.5$

Group 2) $\triangle' 2, 1=0$, $\triangle' 2, 2=1$,

$\triangle' 2, 3=0$ $\quad \triangle 2=0.333$

Group 3) $\triangle' 3, 1=11$, $\triangle' 3, 2=12$,

$\triangle' 3, 3=10$ $\quad \triangle 3=11$

단계 10) [S] + 1번째 그룹은 G_2가 된다.

$S=\{G_2\}$, $S'=\{G_1, G_3\}$

단계2로 가서 순서가 정해지지 않은 그룹내의 Job순서를 다시 정한다.

Iteration 2

그룹 1 : Job 1,1) $\widetilde{C}_{1,1}=30, d_{1,1}=55$ $\triangle 1,1=25$

Job 1,2) $\widetilde{C}_{1,2}=37, d_{1,2}=68$ $\triangle 1,2=31$

$B_1=\{1\}$

$B_1=\{1, 2\}$

그룹 3 : Job 3,1) $\widetilde{C}_{3,1}=38, d_{3,1}=46$ $\triangle 3,1=11$

Job 3,2) $\widetilde{C}_{3,2}=38, d_{3,2}=48$ $\triangle 3,2=10$

Job 3,3) $\widetilde{C}_{3,3}=37, d_{3,3}=37$ $\triangle 3,3=0$

$B_3=\{3\}$

Job 3,1) $\widetilde{C}_{3,1}=44, d_{3,1}=46$ $\triangle 3,1=2$

Job 3,2) $\widetilde{C}_{3,2}=44, d_{3,2}=48$ $\triangle 3,2=4$

$B_3=\{3, 1\}$

$B_3=\{3, 1, 2\}$

Group 1) $\triangle' 1, 1=25$, $\triangle' 1, 2=29$

$\triangle 1=27$

Group 3) $\triangle' 3, 1=0$, $\triangle' 3, 2=2$,

$\triangle' 3, 3=0$ $\quad \triangle 3=0.667$

$S=\{G_2, G_3\}$, $S'=\{G_1\}$

Iteration 3

그룹 1 : Job 1,1) $\widetilde{C}_{1,1}=55, d_{1,1}=55$ $\triangle 1,1=0$

Job 1,2) $\widetilde{C}_{1,2}=59, d_{1,2}=68$ $\triangle 1,2=9$

$B_1=\{1, 2\}$

최종순서는 G_2 (1-3-2)-G_3 (3-1-2)-G_1(1-2)

단계11) 순수납기지연의 합을 계산한다.

본 알고리즘으로 구한 해의 순수납기지연의 합은 0으로 제시한 문제의 최적해와 같다.

4. 발견적 기법의 평가

실제상황에서 문제에 대한 해를 쉽고 빨리 얻기 위하여 최적해에 근사한 발견적 기법의 필요성이 증대되고 있으나 그 유효성을 평가하는 절대적인 방법론이 아직은 없다 [7, 12]. 본 연구에서는 개발한 발견적 기법을 FORTRAN으로 프로그램하고 VAX-11/780 컴퓨터를 이용하여, 그 유효성을 다음의 두 방법에 의해 평가하였다.

(1) 최적해를 아는 문제를 만들어 발견적 기법의 유효성을 평가한다.

(2) 랜덤 스케쥴에 의해서 얻은 해와 본 연구

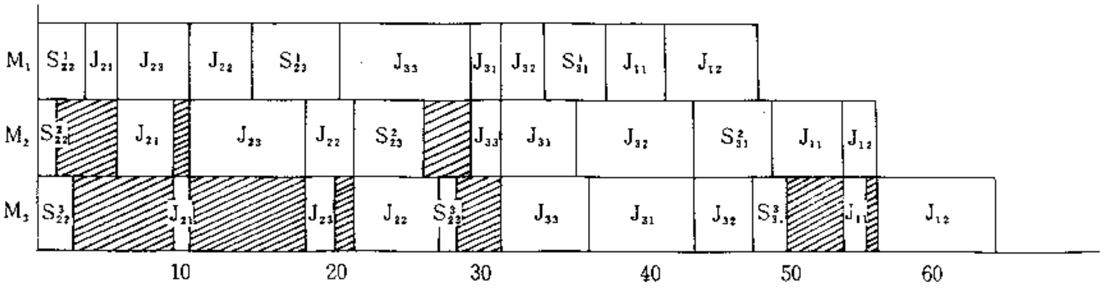

Figure 3. Gantt Chart for Example

에서 제시한 발견적 기법으로 얻은 해를 비교한다.

4.1 최적해를 아는 문제를 사용한 평가

Job의 순수납기지연은 정의에 의해 0보다 크거나 같으므로 순수납기지연의 합이 0이 되는 해는 주어진 문제에 대한 최적해가 된다. 따라서 주어진 문제에서 job의 가공시간 및 준비시간을 랜덤하게 부여하고, 각 job의 납기를 임의의 순서에 대해 job의 가공완료시간과 같게 설정하면 순수납기지연의 합이 0이 되는 최적해 문제를 만들 수 있다[3].

이와 같은 방법에 의하여 그룹스케쥴링 문제를 기계와 그룹을 3에서 5까지 변화시키고 그룹당 job수가 3, 4, 5인 총 27개의 문제를 만들어 평가하였다.

시험의 결과는 발견적 기법에 의해 얻은 해가 문제의 크기와 무관하게 최적해와 일치하였는데 이 결과는 job의 납기부여 방법과 발견적 기법의 특성에 기인한 것이다. 만들어진 문제에서 $d_{i,j}=C_{i,j}$ ($i=1, 2 \cdots\cdots M$; $j=1, 2 \cdots\cdots n_i$) 이므로 job순서 결정에 사용된 $\Delta i, j=0$이 되고 그룹순서 결정단계마다 평가척도인 $\Delta i=0$이 된다. 결국 초기에 제시한 임의의 순서와 같은 순서로 순수납기지연의 합을 0으로 하는 결과를 나타낸다.

위의 결과에 의해 제안된 발견적 기법의 유효성에 대한 일반적인 결론은 내릴 수가 없으나 최적해를 알고 있는 문제에 대해서는 발견적 기법이 좋다는 결론을 내릴 수 있다.

4.2 랜덤 스케쥴을 이용한 평가

스케쥴링 문제는 복잡한 조합의 문제로써 최적해를 열거법 (Permutation schedule)에 의해서 얻는다는 것은 실제로 불가능하다. 따라서 하나의 대안으로 발견적 기법에 의해서 얻은 해를 랜덤 스케쥴에 의해서 얻은 해와 비교하였다.

시험한 문제는 그룹수 3, 4, 5, 공정수 3, 4, 5 그리고 각 그룹내의 작업수를 3부터 10까지 변화시킨 72개의 문제를 반복하여 총 144개의 문제를 문제당 1000개씩의 랜덤 스케쥴과 비교하였다. 시험을 위한 문제는 VAX-11/780 컴퓨터에 내장된 난수발생기 (Random Generator)를 사용하여 가공시간과 준비시간을 1과 9, 납기를 11과 99사이의 구형분포 (Uniform Distribution) 난수로써 발생시켰다[4].

i) 발생빈도에 따른 평가

발견적 기법으로 구한 작업순서에 따른 순수납기지연의 합이 랜덤 스케쥴에 따른 순수납기지연의 합보다 작게 나타나는 발생빈도를 Table 5에 나타내었다.

Table 5. Comparison of Heuristic Algorithm & Random Scheduling with Frequency

발생빈도(%)	문제수	문제수의비율(%)	누적비율(%)
100.00	6	4.17	4.17
99.99-90.00	64	44.44	48.61
89.99-80.00	35	24.31	72.92
79.99-70.00	13	9.03	81.95
69.99-60.00	16	11.11	93.06
59.99-50.00	7	4.86	97.92
49.99-40.00	3	2.08	100.00
39.99 이하	0	0.00	....
Total	144	100.00	

Table 5에 나타난 바와 같이 시험한 총 144개의 문제 중 73%인 105개의 문제가 발생빈도 80% 이상의 근사최적해를 얻을 수 있었고 평균적으로 85.5%의 발생빈도로 근사최적해가 랜덤스케쥴보다 우수하였다.

이 시험 과정에서 발견적 기법의 유효성과 매개변수인 그룹수, 그룹내 job수, 공정수 사이에 어떤 관계가 있는지를 알아보기 위해 SPSS Package를 이용하여 분산분석을 행한 결과를 Table 6에 제시하였다.

분산분석의 결과 매개변수인 그룹내의 job수가 유의 수준 0.1에서 유의하다. 즉 그룹내 job수가 증가하면 랜덤 스케쥴보다 우수한 해의 발생빈도가 높아짐을 알 수 있는데, 그 이유

Table 6. Analysis of Variance

Source of Variation	Sum of Square	DF	Mean of Square	F
Group	204.694	2	102.347	0.565
Job	2,533.968	7	361.995	1.999*
Machine	392.959	2	196.480	1.085
Group×job	3,867.559	14	276.254	1.525
Group×Machine	901.200	4	225.300	1.244
Job×Machine	2,695.051	14	192.504	1.063
Group×Job×Machine	4,026.016	28	143.786	0.794
Error	13,039.990	72	181.111	
Total	27,661.438	143	193.437	

는 그룹내 job의 순서를 정할 때, 모든 job에 대하여 단계적으로 반복하는 방법을 택하기 때문이다.

ii) 단축비에 의한 평가

제시한 기법으로 구한 해가 랜덤 스케쥴 보다 얼마나 평가기준을 단축시킬 수 있는가를 알아보기 위하여 단축비를 이용한다.

단축비는 다음 식(13)에 의해 얻을 수 있다.

$$R = \frac{(\overline{T}_r - \overline{T}_h)}{\overline{T}_r} \times 100\,(\%) \quad \cdots\cdots(13)$$

여기서 $\overline{T}_r$은 1000개의 랜덤 스케쥴에 의해 얻어진 순수납기지연의 합의 평균이며, $\overline{T}_r$는 발견적 기법에 의해 얻어진 순수납기지연의 합의 평균을 의미한다.

R의 부호가 음수이면 발견적 기법의 해가 랜덤 스케쥴보다 나쁜 결과를 가져온 것이고 양수이면 좋은 결과를 가져온 것이다. 그 값이 클수록 랜덤스케쥴에 의한 순수납기지연의 합을 그 값 만큼 단축시킨 것이다.

각 문제 형태에 따른 평균 단축비를 Table 7에 나타내었다. 각 경우마다 6개의 문제를 평균하여 24가지 경우, 총 144문제에 대한 평균 단축비 $\overline{R}$는 11임을 알 수 있다. 그리고 단축비는 그룹내 job수의 변화에 영향을 받고 있음을 알 수 있다.

iii) 소요시간 계산

VAX-11/780 컴퓨터로 제시된 발견적 기법을 수행하는 경우에 필요한 CPU 시간을 다음 Table 8에 나타내었다. 발견적 기법으로 문제의 해를 구하는데 소요되는 시간은 표에서 본 바와 같이 그룹수와 job수의 변화에 크게 영향을 받고 있음을 알 수 있다.

Table 7. Reduction Ratio

Job \ Group	3	4	5
3	9.72	3.15	2.63
4	1.04	8.07	7.01
5	4.34	6.12	3.84
6	12.63	6.14	6.98
7	13.16	4.61	12.93
8	20.21	12.47	10.06
9	23.70	17.25	14.57
10	21.02	17.20	15.47

Table 8. CPU time of Heuristic Algorithm by VAX-11/780 Computer

GROUP = 3									
# OF JOB		3	4	5	6	7	8	9	10
# OF MACHINE	3	0.120	0.250	0.435	0.715	1.095	1.535	2.175	3.090
	4	0.150	0.300	0.560	0.900	1.430	2.015	2.800	3.530
	5	0.155	0.370	0.675	1.125	1.740	2.495	3.595	4.055
GROUP = 4									
# OF JOB		3	4	5	6	7	8	9	10
# OF MACHINE	3	0.205	0.460	0.815	1.370	2.065	3.230	4.410	6.045
	4	0.265	0.590	1.110	1.820	2.750	4.030	5.690	7.390
	5	0.325	0.750	1.400	2.325	3.485	5.225	7.190	8.100
GROUP = 5									
# OF JOB		3	4	5	6	7	8	9	10
# OF MACHINE	3	0.375	0.830	1.450	2.430	3.770	5.495	7.585	10.510
	4	0.470	1.030	1.920	3.015	4.905	7.410	10.055	14.470
	5	0.595	1.275	2.420	3.900	6.040	8.780	12.240	17.795

5. 결 론

본 연구에서는 그룹 스케쥴링 문제 중 다종소량 다단계 생산 시스템에서 그룹간의 준비시간이 그룹순서에 따라 변하는 경우 순수납기지연의 합을 최소화하기 위하여 모델을 설정하고 분석 및 고찰하였다.

문제의 최적해에 가까운 근사해를 구할 수 있는 발견적 기법을 개발하고 그 유효성을 평가하기 위하여 최적해를 알고 있는 문제의 해와 랜덤 스케쥴에 의한 발생빈도를 비교한 결과, 최적해를 알고 있는 경우 총 27개의 문제 모두가 최적해와 같은 해를 가졌고, 각각 1000개의 랜덤 스케쥴을 이용하여 비교한 144개의 문제에서는 약 85.5%의 발생빈도로 랜덤 스케쥴보다 우수하였다. 또한 발생빈도는 그룹내 job 수가 증가할 수록 좋아지는 결과를 나타내었다. 단축비에 의한 평가에서는 발견적 기법으로 구한 작업순서에 따른 순수납기지연의 합이 랜덤 스케쥴에 의해서 구한 순수납기지연의 합을 11% 단축시킴을 알 수 있었다.

앞으로 준비시간이 종속적인 경우에 또 다른 평가기준에 대하여 최적해를 구하기 위한 연구와, 더욱 효율성이 높은 근사 최적해를 구하기 위한 연구가 필요한 것으로 생각한다.

References

1. 조규갑, "다단계 생산 시스템에서의 그룹 스케쥴링에 대한 연구", 대한산업공학회지, 9권, 1호, pp.23-31, 1983.
2. Baker, K. R., *Introduction to Sequencing and Scheduling*. Wiley, New York, 1974.
3. Baker, K. R., and J. W. M. Bertrand, "A Comparison of Due-Data Selection Rules," *AIIE Transactions*, Vol. 13, No. 2, June, pp. 123-131, 1981.
4. Dannenbring, D. G., "An Evaluation of Flow-shop Sequencing Heuristics," *Management Science*, Vol. 16, No. 10, pp. B630-B637, 1970.

5. Foo, F. C., and J. G. Wagner, "Setup times in Cyclic and Acyclic Group Technology Scheduling Systems," *International Journal of Production Research*, Vol. 21, No. 1, pp. 63-73, 1983.
6. Hitomi, K., and I. Ham, "Operation Scheduling for Group Technology Application," *Annals of CIRP*, Vol. 25, No. 1, 1976.
7. Ignizio, J. P., "Solving Large-Scale Problems; A Venture into a New Dimension," *Journal of Operational Research Society*, Vol. 31, pp. 217-225, 1980.
8. Nakamura, N., and K. Hitomi, "Optimization of Group Scheduling for the Multiple Production Stages," (Japanese), *Transactions of Japan Society of Mechanical Engineers*, Vol. 42, No. 361, pp. 2964-2973, 1976.
9. Petrov, V. A., *Flowline Group Production Planning*, Business Publications Limited, London, 1968.
10. Yoshida. T., and K. Hitomi, "Optimization of Group Scheduling for Signle Stage Production with Dependent Setup time," (Japanese), *Transactions of the Japan Society of Mechanical Engineers*, Vol. 43, pp. 1531-1538, 1977.
11. Yoshida, T., N. Nakamura, and K. Hitomi, "Optimization of Group Scheduling for Single Stage Production," (Japanese), *Transactions of the Japan Society of Mechanical Engineers*, Vol. 39, pp. 1993-2003, 1973.
12. Zanakis, S. H., and J. R. Evans, "Heuristic Optimization: Why, When, and How to use It," *Interface*. Vol. 11, No. 5, October, pp. 84-90, 1981.