經營學硏究 第17卷 第1號(1987年 9月)

中間貯藏所가 없는 흐름공정의 日程計劃 技法의 比較 硏究

——A study on flowshop scheduling problem with no intermediate storage——

姜 炳 瑞*

≪目 次≫

Ⅰ. 서 론

작업일정계획의 기본 임무는 작업순서를 결정하여 최적의 日程을 정하는 것이다. 작업순서의 決定 技法은 작업순서가 일정한 흐름工程 日程計劃問題(flowshop scheduling problem)와 그 순서가 불규칙한 個別工程 日程計劃問題(job-shop scheduling problem)에 관한 技法으로 大別할 수 있다. 本 硏究에서는 흐름 공정에 관한 일정계획문제를 다루기로 한다.

일반적으로 고전적인 흐름공정 일정계획 문제(classical flowshop scheduling problem)는 M개의 기계에서 가공되는 N개의 작업의 순서를 결정하는 문제이다. 이 고전전인 흐름공정 일정계획문제는 CFS라고 부르기도 한다. CFS 모형은 다음과 같이 작업, 기계, 가공시간, 그리고 납기일에 관한 가정에 근거하여 개발 되었다([1], [3], [4]).

i) N개의 작업내용은 미리 알려져 있다고 가정한다.

ii) 모든 작업은 일단 시작이 되면 한 기계에서 완성될 때까지 수행되어야 한다.

iii) 기계 사이의 在工品 在庫는 허용되며, 작업의 순서는 技術的으로 일정하게 정해져 있다.

iv) 기계들은 작업시간 전에 이용이 가능하고, 그 배치는 미리 정해져 있다.

v) 각 작업의 가공시간도 또한 미리 알려져 있다.

---

* 慶熙大學校 經營學科 助敎授

vi) 納期日은 마지막 작업이 기계에서 공정이 끝나는 시간으로 한다.

CFS 모형은 이상과 같은 가정하에서 개발되었다. 그런데 이 모형을 살펴보면 기계사이에 中間貯藏所가 있어 다음 기계가 이용 가능할 때까지 부분적으로 가공이 끝난 작업을 보관한다. 在工品 在庫는 기계들 사이에서 저장되게 된다. 만일 이 가정에 제한을 두게 되면, 이 CFS 문제는 중간저장소가 없는 흐름공정 일정계획 문제(flowshop scheduling problem with no intermediate storage)로 변형된다. 이 일정계획 문제는 이하 NIS 모형이라고 부른다. 이 모형에서는 일단 작업의 가공이 첫 기계에서 시작되면, 도중에 기다림이 없이 연속적으로 가공되도록 하여 마지막 기계에서 완료하여야 한다. 따라서 작업의 지체는 가공 시작 직전에만 일어나게 된다. 이 가정만 제외하고는 CFS 모형의 가정은 모두 NIS 문제에 적용된다.

NIS 모형은 실제적으로 철강이나 알루미늄 공장에서 발견될 수 있다([13]). 강철을 압연하는 동안에 강철은 全體 생산시스템을 통하여 높은 온도를 유지하여야 한다. 철판의 두께가 계획된 크기로 될 때까지 압연공정을 계속하여 철판을 얇게 누른다. 이 공정에서 철판은 한 압연기계에서 다음 기계까지 지체되지 않고 연속적으로 즉시 옮겨져야 한다. 만일에 지체된다면, 철판이 식어져서 압연작업을 계속하기가 힘들게 된다. 이러한 실제 상황에서 작업이 일단 시작되면 전체 생산 시스템을 통하여 계속적으로 공정이 이루어져야 한다.

중간저장소가 없는 생산모형을 가지게 되면 다음과 같은 이점을 가질 수 있다. 생산 공정 중에 있는 재공품이 없게 되므로 재공품을 위한 저장장소, 人力, 또는 돈의 필요성을 제거할 수 있다. 따라서 NIS 문제를 해결하면 자본투자를 줄일 수 있을 뿐 아니라 재공품 재고를 零의 수준으로 하는 zero-inventory 정책을 실행할 수 있다.

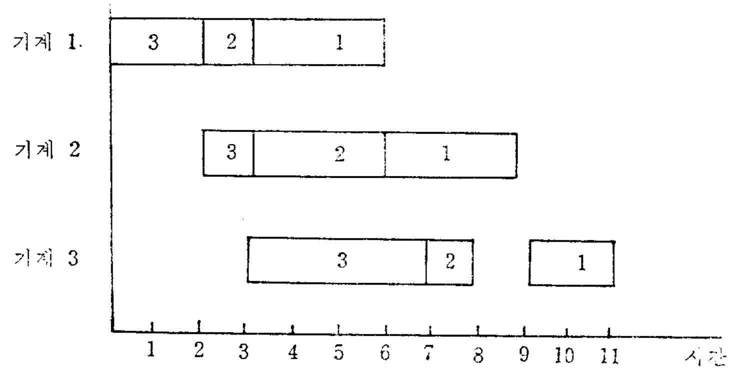
Ⅱ. CFS 모형과 NIS 모형의 비교

이 節에서는 고전적인 흐름공정 일정계획 (CFS)모형과 중간저장소가 없는 흐름공정 일정계획 (NIS)모형을 이해하기 위하여 간트 챠트(Gantt Chart)를 이용하여 설명한다. 가령 세 개의 작업과 세 종류의 기계를 가지고 있는 공장이 있다고 하자. 그리고 加工時間이 다음과 같이〈表 1〉에 주어졌다고 하자.

〈表 1〉　　가공시간 자료(단위 : 시간)

기계 \ 작업	1	2	3
1	3	1	2
2	3	3	1
3	2	1	4

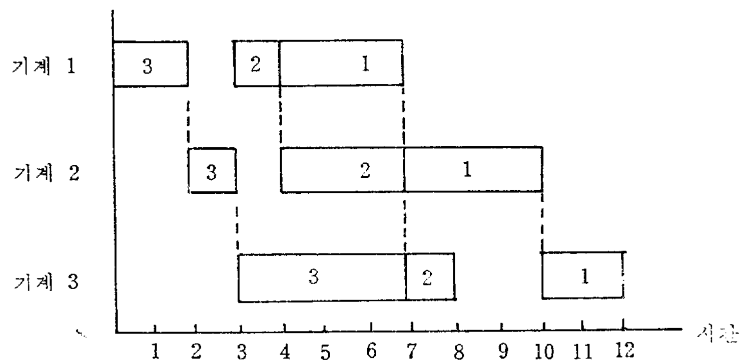
작업을 가공하는 기술적인 순서가 작업 3—작업 2—작업 1이라고 가정한다. 〈그림 1〉은 〈表 1〉에서 주어진 자료를 이용한 CFS 문제의 가공순서를 간트챠트로 나타낸 것이다.



〈그림 1〉 CFS 문제의 가공순서

위 그림에서 보면 작업 3이 기계 1에서 끝나면 그 다음 기계인 2와 3에서 계속 가공될 수 있다. 그 다음 작업인 작업 2는, 전 작업이 끝나자 마자 기계 1과 2에서 가공된다. 그러나 작업 2는 작업 3이 끝날 때까지 기계 3에서 가공되기 위하여 1시간 단위를 기다려야 한다. 이 CFS 모형의 경우에 마지막 작업은 첫 기계에서 끝나자 계속 그 다음 기계에서 가공될 수 있다. 작업순서 3-2-1에 따라 모든 작업의 공정이 끝나면 총공정시간은 11시간 단위가 된다. 그리고 기계休止시간(machine idle time)을 살펴보면, 기계 1은 없다. 기계 2는 시간 0에서 2 사이의 2시간 단위를 休止하며, 기계 3은 시간 0에서 3까지 그리고 시간 8에서 9까지 모두 4시간 단위를 休止한다. 이 경우에 CFS의 기계휴지시간은 총 9시간이 된다.

다음으로 NIS 문제의 가공순서를 나타내기 위하여 같은 자료를 이용하여 간트 챠트를 〈그림 2〉에 나타내 보자. NIS 문제에서는 한 작업이 일단 시작되면 전체 생산시스템을 통하여 지체없이 가공되어야 한다. 작업 3은 CFS 문제에서와 마찬가지 방법으로 처음에 가공된다. 그러나 작



〈그림 2〉 NIS 문제의 가공순서

업 2의 가공은 다르다. 중간에서 기다리지 못하는 조건을 충족시키기 위하여 기계 1은 작업 3의 완성시간과 작업 2의 시작시간 사이의 1시간 단위 동안 쉬어야 한다. 그리고 나서 작업 2의 가공시간은 기계 1에서 시간 4에 이루어 지며, 기계 3에서는 시간 8에 끝이 난다. 다른 작업도 이와 마찬가지로 가공된다. NIS 모형의 총공정시간은 12시간 단위이며, 이것은 CFS보다 1시간단위 더 길다. 그리고 NIS의 기계휴지시간은 모두 9시간이다.

Ⅲ. 문제해법의 설명

고전적인 흐름공정 일정계획 문제의 가정에 의하면 모든 작업은 똑같은 기술적인 순서에 따라 가공된다. 따라서 각 기계에서 각기 다른 방법의 작업순서에 걱정할 필요없이 하나의 순열순서에 의하여 일정계획이 결정된다. N개의 작업과 M개의 기계가 있을 때, 모든 기계에서의 공통적인 순서때문에 흐름공정 일정계획 문제의 해결 가지수는 다음과 같이 줄어든다. 즉, 일반적인 개별공정 문제의 경우에 $(N!)^M$ 가지의 순서가 있으나 이것은 고전적인 흐름공정 문제의 경우에 N!로 줄어든다. 앞의 예제인 3개의 작업을 3개의 기계에서 가공할 때 순서의 선택가지수는 $(3!)^3=216$에서 $3!=6$으로 줄어든다. 일정계획 문제는 최적의 작업순서를 이 6가지 중에서 하나를 선택하는 것인데, 본 연구의 선택기준은 총공정시간이 제일 짧은 작업순서를 선택하는 것이다. 앞의 예제의 경우에 3개의 작업을 순열로 나타내면 1-2-3, 1-3-2, 2-1-3, 2-3-1, 3-1-2, 그리고 3-2-1의 6가지가 생긴다. 이 중에서 예제는 3-2-1을 임의로 선택한 것이다.

CFS 문제에서 간단한 순서 선택 방법 덕분에 많은 연구가들은 여러종류의 해법을 개발하였다. Johnson은 두 개의 기계가 있는 경우에 최적해를 발견하였다([8]). 이 존슨기법을 토대로 하여 여러가지 技法이 개발되었다. 그 종류를 살펴보면, 휴지스틱기법(heuristic methods) ([3], [4]), 分枝限界法(branch-and-bound techniques) ([2], [7]), 整數計劃法(integer linear programming methods) ([11], [12]), 그리고 시뮬레이션 ([6])이 있다.

본 연구의 目的은 NIS 문제를 풀기위한 알고리듬을 찾는데 있다. 앞에서 말한 바와 같이 NIS 문제는 CFS 문제에서 수정변형되었음을 알아야 한다. 본연구에서는 CFS 문제를 해결한 여러기법 중에서 선택하여 NIS 문제를 해결하도록 하였다. 이 연구를 위하여 특별히 세 개의 흐름공정일정계획기법을 선택하였는데, 그것은 Dannenbring 알고리듬([4]), Campbell-Dudek-Smith 알고리듬([3]), 그리고 Stafford의 數理模型技法 ([10])이다. 이 세 기법을 차례로 설명하면 다음과 같다.

첫째, Dannenbring은 그의 일정계획법에서 매우 우수한 휴리스틱을 개발하였다. 이 방법은 그의 연구에서 RAES(Rapid Access with Extensive Search)라고 불리웠으나, 여기서는 DAN이라고 하겠다. 그의 방법은 2단계를 가진다. 첫 단계로서, 원래의 M개의 기계를 가진 문제는

간이적으로 두 개의 기계를 가진 문제로 전환된다. 그런 다음 존슨기법을 적용하여 시초의 작업순서를 쉽고 빠르게 결정한다. 그 다음에 두번째 단계로서, 시초의 순서에서 서로 이웃한 두 작업의 순서를 바꾼 후에 각각의 경우에 대하여 총공정시간을 계산한다. 총공정시간이 제일 짧은 작업순서를 선택한다. 이것을 다시 시초로하여 같은 방법을 반복한다. 그리하여 가장 좋은 解가 얻어지면 이 방법은 끝이 나게 된다.

둘째, Campbell-Dudek-Smith 알고리듬은 Dannenbring 의 기법보다 먼저 개발되어 고전적인 일정계획문제를 해결한 휴리스틱 기법이다. 이 기법은 RAES 와는 조금 다르게 "최적해"를 구하려고 시도하였다. 원본 문제를 $M-1$ 개의 두기계 문제로 축소시켜서 그 중에서 총공정시간이 제일 짧은 공정순서를 선택한다. 이때에 물론 존슨기법이 이용된다. 그런데 이 기법은 본 연구를 위해서 DAN 에서 쓰인 두번째 단계 개선방법을 도입하여 보다 나은 解를 구하고 있다. 본 연구에서는 이 기법을 CDS 라고 부른다.

마지막으로, 數理模型은 원래 CFS 문제를 해결하기 위하여 개발되었으나, 본 연구의 NIS, 문제 해결을 위하여 수정되었다. 이 기법은 MAT 로 부르기로 한다. NIS 모형에서 결정변수는 다음과 같다.

$$_{ij} = \begin{cases} 1 & \text{만일 작업 } i \text{가 위치 } j \text{에서 가공되면} \\ 0 & \text{그외} \end{cases}$$

$$(i, j=1, 2, \cdots\cdots, N)$$

T_j^k=기계 k 에서 위치 j 에 있는 작업의 가공시간

D_j^k=위치 j 에 있는 작업의 시작전 기계 k 에서의 지연시간

MAT 모형은 총공정시간 X_f 를 최소화하는 작업순서를 결정한다. 따라서 목적함수와 제약조건은 다음과 같다.

최소화 X_f

제약조건

$$\sum_{j=1}^{N} X_{ij}=1 \quad (i=1, 2, \cdots, N) \cdots\cdots (1)$$

$$\sum_{i=1}^{N} X_{ij}=1 \quad (j=1, 2, \cdots, N) \cdots\cdots (2)$$

$$(D^k_{j+1}+T^k_{j+1})-(T^{k+1}_j+D^{k+1}_{j+1})=0 \cdots\cdots (3)$$

$$(j=1, 2, \cdots\cdots, N-1;\ k=1, 2, \cdots, M-1)$$

$$\sum_{i=1}^{k} T_1^{\,i}-D_1^{k+1}=0 \cdots\cdots (4)$$

$$(k=1, 2, \cdots, M-1)$$

$$X_f-\sum_{j=1}^{N}T_j^M-\sum_{j=1}^{N}D_j^M=0 \quad\cdots\cdots(5)$$

모든 변수≥0

제약조건 (1)은 모든 순서위치가 반드시 하나의 작업을 차지하는 것을 설명하며, 그리고 제약조건 (2)에서는 각 작업은 반드시 한 위치에 할당되어야 한다. 그 다음의 제약조건을 보면, 기계 k와 $k+1$의 두 이웃 작업사이의 관계를 설명한다. 위치 $j+1$에 있는 작업을 가공하기 前 기계의 유지, 시간과 그 작업의 가공시간의 合은 기계 $k+1$에서 위치 j에 있는 작업의 가공시간과 위치 $j+1$에 있는 작업을 가공하기 前기계 $k+1$의 휴지 시간의 合과 같다. 제약조건 (4)를 살펴보면 작업의 시작전에 있는 기계$k+1$의 기계휴지시간은 기계 1부터 k까지에서 첫위치에 가공되는 작업들의 가공시간 合과 같다. 마지막으로, 총공정시간 X_f는 마지막 기계에서의 모든 작업의 가공시간과 기계휴지시간을 合한 것과 같다. 따라서 NIS수학적 모형은 결정변수, 목적함수, (1)부터 (5)까지의 제약조건, X_{ij}가 0 또는 1인 조건, 그리고 非陰조건으로 이루어진 혼합형 정수계획법이다.

Ⅳ. 실험 계획

이 연구에서는 비교적 작은 크기의 문제가 조사되었다. 문제의 크기란 작업의 수와 기계의 수의 組合을 뜻한다. 작업의 수는 10 이하이며 기계의 수는 5 이하이다. 실험계획을 위한 작업요인의 수준은 5, 7, 10이다. 또한 기계요인의 수준은 3, 4, 5이다. 각 문제의 가공시간 범위는 1~10시간 단위이다. 모두 9가지의 서로 다른 크기의 문제가 실험계획에서 고려되었으며, 각 문제에 대하여 10개의 표본이 임의로 FORTRAN Simulation에 의하여 만들어 졌다. 따라서 총 90 문제가 만들어진 셈이고, 이것들은 세가지 기법에 의하여 각각 그 解가 주어졌다.

이 연구의 초점은 CFS모형을 위해 개발된 기법들이 과연 NIS모형의 문제를 풀 수 있는가 하는 것이며, 만일 그렇다면 어느 것이 가장 좋은 解를 낼 것인가를 조사하는 일이다. 각 기법의 성과를 평가하기 위하여 총공정시간(makespan)과 기계휴지시간(machine idle time)이 고려되었다. 일반적으로 경영자는 생산운영비를 최소화 시키는 것에 관심을 많이 쏟는다. 이 비용은 생산시스템에서 소요되는 시간으로 계산될 수 있다. 따라서 총공정시간의 최소화가 성과측정기준으로 채택되었다. 또 다른 기준은 기계휴지시간으로 경영자는 기계를 최대한으로 이용하여 생산을 늘이려 한다. 따라서 기계의 휴지시간을 단축시키고자 한다.

실험자료는 컴퓨터를 이용하여 얻어졌다. 두 휴리스틱 기법인 DAN과 CDS는 FORTRAN에 의하여 프로그램되었으며, 수리모형인 MAT는 Mathematical Programming System Extended/370(MPSX/370)이라는 LP패키지를 이용하였다.

V. 결 과

중간저장소가 없는 흐름공정 일정계획기법의 성과를 조사하기 위한 실험결과를 다룰 다음과 같이 요약하였다. 실험계획의 9종류 문제에 대한 평균값이 〈表 2〉에 정리되었다. 작은 크기의 문제를 푸는데 있어 세가지 技法에 의한 총공정시간의 평균차이는 극히 작다. 그러나 문제의 규모가 커짐에 따라 차이가 난다. 전체문제를 통하여 MAT의 평균이 제일 작으며 그리고 CDS와 DAN의 평균은 MAT보다 큰 것을 알 수 있다. 또한 기계휴지시간의 평균치를 살펴보면 총공정시간의 경우와 비슷한 양상을 보인다. 따라서 두 성과 측정치 사이에는 비교적 큰 陽의 상관관계가 있어 총공정시간이 작으면 기계휴지시간도 작고, 전자가 크면 후자도 크다. 결론적으로 말하면, 혼합정수계획법은 실험계획의 모든 문제에 대하여 일관성있는 좋은 성과를 보였다. 반대로 두 휴리스틱방법은 작은 크기의 문제에서는 비교적 양호하였으나, 큰 문제에서는 수리모형에 비해 좋은 성과를 나타내지 못했다.

〈表 2〉 총공정시간과 기계휴지시간의 평균

작업의 수	기계의 수	DAN		CDS		MAT	
		총공정	기계휴지	총공정	기계휴지	총공정	기계휴지
5	3	43.3	33.8	42.8	32.0	42.7	31.5
5	4	51.7	61.5	51.6	61.1	51.2	61.1
5	5	64.9	127.3	64.6	125.8	64.1	123.1
7	3	54.0	33.0	54.0	33.0	53.7	32.1
7	4	64.8	94.0	65.3	96.1	64.4	93.4
7	5	70.2	145.2	69.4	141.2	67.5	133.4
10	3	81.3	47.8	80.3	45.6	77.9	36.8
10	4	85.2	97.1	86.1	97.7	82.8	87.3
10	5	95.1	165.6	95.1	164.6	91.2	145.1

다음 表는 각 알고리듬이 解에 도달하기 위해 걸린 CPU시간의 평균치를 정리한 것이다.

〈表 3〉에서 보면, MAT. 휴리스틱 기법인 DAN과 CDS보다 훨씬 많은 CPU시간을 필요로 한다. 작업의 수가 5개 혹은 7개인 경우에, MAT는 10~47초가 필요하였으며 DAN과 CDS는 불과 0.03초 정도 밖에 걸리지 않았다. 더우기, 가장 큰 문제를 풀기 위해서 MAT는 10분이나 걸렸음을 알 수 있다. 같은 경우에, DAN과 CDS는 0.1초 이내에 解를 구하였다. 휴리스틱 기법이 수학적 기법보다 훨씬 더 빠르게 解를 제공하므로 더 효율적이라고 보겠다.

이 연구의 특성은 CFS를 위해 개발된 기법을 변형 수정하여 NIS문제의 解를 구하는 데 있

〈表 3〉 평균 CPU 시간 (단위 : 초)

작업의 수	기계의 수	DAN	CDS	MAT
5	3	0.0052	0.0067	9.6250
5	4	0.0097	0.0103	11.8120
5	5	0.0105	0.0114	11.0940
7	3	0.0192	0.0158	14.8820
7	4	0.0171	0.0191	30.9180
7	5	0.0235	0.0298	46.3120
10	3	0.0613	0.0656	262.4310
10	4	0.0265	0.0379	290.6040
10	5	0.0629	0.0503	698.9610

다. 이 연구를 기초로 하여 NIS 문제를 해결하는 데 있어 다른 좋은 알고리듬을 개발될 수 있을 것 이다. 이 경우에 MAT는 성과비교를 위한 좋은 참고가 될 것이다. NIS 문제는 在工品在庫의 수준을 零으로 하려는 生產시스템에 유용하며, 특히 근간에 논의되고 있는 Just-in-time 생산모형에 이용할 수 있을 것이다. 끝으로, Reddi와 Ramamoorthy ([9]) 그리고 Wismer ([13])는 외판원문제(traveling Salesman problem)를 이용하여 NIS 문제를 定型化하였다. 외판원문제는 외판원이 각 도시를 한번만 방문하여 출발점까지 가장 짧은 거리를 돌아오는 경로를 찾는 문제이다. 두 도시사이의 거리는 NIS 문제의 첫 기계에서 생기는 지연시간으로 대치될 수 있다. 이 지연시간을 최소화하여 NIS 문제를 해결한다.

REFERENCES

1. Baker, K.R., *Introduction to Seqnencing and Scheduling*. New York: Wiley, 1974.
2. Brown, A.P.G. and Lomnicki, Z.A. "Some Applications of the Branch and Bound Algorithm to the Machine Scheduling Problem," *Operational Research Quarterly* 17, No. 2(June 1966):173-186.
3. Campbell, H.G.; Dudek, R.A.; and Smith, M.L., "A Heuristic Algorithm for the N Job M Machine Sequencing Problem." *Management Science* 16, No. 10(June 1970); B630-B637.
4. Dannenbring, D.G., "The Evaluation of Flowshop Sequencing Heuristics." *Management Science* 23, No. 11(July 1977): 1174-1182.
5. Gupta, J.N.D., "Optimal Flowshop Schedules with No Intermediaet Storage Space." *Naval Research Logistics Quarterly* 23, No. 2(June 1976): 235-243.
6. Heller, J., "Some Numerical Experiments for an m x Flowshop and the Decision-Theoretical Aspects," *Operations Research* 8, No. 2(Mar.-Apt. 1960): 170-184.
7. Ignall, E. and Schrage, L., "Application of the Branch and Bound Technique to Some Flowshop Scheduling Problems." *Operations Research* 13, No. 3(May-June 1965): 400-412.
8. Johnson, S.M., "Optimal Two-and Three-Stage Production Schedules with Set-up Times Included," *Naval Research Logistics Quarterly* 1, No. 1(Mar. 1954): 61-68.

9. Reddi, S.S. and Ramamoorthy C.V., "On the Flowshop Sequencing Problem with No Wait in Process." *Operational Research Quarterly* 23, No. 3(Sept. 1972): 323-331.
10. Stafford, E.F., "A Mixed-Integer Linear Programming Model for the Classical Flowshop Sequencing Problem." Working Paper, University of South Carolina, 1983.
11. Stafford, E.F.: Srobel, C.D. and Kang, B.S., "Comparison of Flowshop Scheduling Techniques With MILP," *Proceedings*—1984 Southeastern Meeting of the American Institute for Decision Sciences. Edited by R. Flood, Feb., 1984.
12. Wagner, H.M., "An Iateger Linear Programming Model for Machine Scheduling." *Naval Research Logistics Quarterly* 6, No. 2(June 1959): 130-140.
13. Wismer, D.A., "Solution of the Flowshop Scheduling Problem with No Intermediate Queues," *Operations Research* 20, No. 3(May-June 1972): 689-697.