

공정납기를 이용한 Job Shop 일정계획의 발견적기법

A Heuristic Technique for Job Shop Scheduling Using Operation Due Dates

배상윤* · 김여근* · 김영균**

* 전남대학교 공과대학 산업공학과, ** 전주공업전문대학 공업경영과

Abstract

This paper presents a multi-pass heuristic for job shop scheduling with due dates. The proposed heuristic iteratively improves solutions based on operation due dates. To find a good solution efficiently, a method for searching the neighborhood of current schedule is developed. The heuristic is compared with two existing heuristics as well as several dispatching rules in terms of solution quality and computation time. The experimental results show that the proposed approach is promising.

1. 서론

일반적으로 job shop에서는 모든 부품이 납기를 갖는다. 이러한 납기를 만족하기 위한 job shop 일정계획에 관한 연구는 주로 짧은 계산시간에 일정을 산출할 수 있는 할당규칙(dispatching rules)에 관하여 이루어져 왔다. 공정납기를 구하는 방법은 CON(Constant Flow Allowance), SLK(equal SLACK), TWK(proportional to Total Work) 등이 제안되어 있다[2]. 본 연구에서는 공정납기를 구하기 위하여 TWK 방법을 사용한다. 공정납기를 사용한 할당규칙은 ODD(Operation Due Date)[4]가 제안된 이후 MOD(Modified Operation Due date)[3]와 다른 할당규칙과 결합된 HYBRID[7], CR(Critical Ratio)+SPT(Shortest Processing Time)와 S/RPT(Slack per Remaining Processing Time)+SPT[1], CEXSPT(Conditionally Expedited by the SPT rule)[8] 등이 개발되었다. 최근 컴퓨터 성능의 향상으로 발견적기법의 적용이 가능함에도 불구하고 이에 관한 연구는 미비한 실정이다. 본 연구에서는 납기만족을 목적으로 하는 job shop 일정계획을 위하여 공정납기(operation due dates)를 이용한 발견적기법을 제안한다. 제안한 발견적기법에서는 공정의 납기정보로부터 탐색을 효율적으로 할 수 있는 이웃해 생성방법을 제시하고, 이를 이용하여 해를 반복적으로 개선해 나간다.

본 연구에서 사용되는 기호는 아래와 같다.

- (i,j) : 부품 i의 j번째 공정
- $t^s(i,j)$: (i,j)의 시작시간
- $t^f(i,j)$: (i,j)의 완료시간
- pt_{ij} : (i,j)의 가공시간
- c_i : 부품 i의 가공완료시간
- d_i : 부품 i의 납기
- d_{ij} : 공정 (i,j)의 공정납기, 즉 $d_{ij} = d_{i-1}$

$+pt_{ij} \times d_i / r_{i1}$ [2], 여기서 $d_{i0} = 0$ 이고 r_{i1} 은 부품 i의 추가공시간을 나타낸다.

T_i : 부품 i의 순수납기지연, 즉 $T_i = [c_i - d_i]^+$

OT_{ij} : 부품 i의 j번째 공정의 납기지연,

즉 $OT_{ij} = t^f(i,j) - d_{ij}$

T_{tot} : 총순수납기지연(total tardiness),

즉 $T_{tot} = \sum_{i=1}^n [c_i - d_i]^+$

$[x]^+ : \max\{0, x\}$

2. 후보이동공정의 선택

이 장은 active chain을 이용하여 삽입이나 교환 등에 이용되는 후보이동공정의 선택방법과 선택된 공정의 효과적인 이동범위를 제안한다.

어떤 공정 (i,j)의 active chain은 active 일정에서 이 공정의 완료시간 $t^f(i,j)$ 에 영향을 주는 공정들을 차례로 나열한 것이다. 이는 (i,j)부터 차례로 시작 시간 $t^s(i,j)$ 을 결정하는 공정, 즉 (i,j)을 가공하는 기계상의 (i,j) 직전공정이나, (i,j) 직선행공정, (i,j-1)을 찾은 과정을 반복함으로써 active chain을 구할 수 있다[9].

현재의 active일정에서 늦은공정의 완료시간을 더 이르게 하기 위해서는 이 공정의 active chain에 있는 공정들의 가공순서가 변경되어야 한다[9]. $OT_{ij} > 0$ 인 공정을 늦은공정, $OT_{pq} > 0$ 인 공정을 이른공정이라 부른다. 본 연구에서는 순수납기지연을 줄이기 위하여 늦은공정은 현 일정보다 더 빨리 시작할 수 있도록 앞으로 삽입하거나 앞 공정들과 교환하는 이동방법을 사용하고, 이른공정은 다른 공정들이 더 빨리 시작할 수 있도록 뒤로 삽입하거나 뒤 공정들과 교환하는 이동방법을 사용한다. 늦은공정 (i,j)의 active chain에 있는 늦은공정 (p,q)와 이른공정 (u,v)의 적절한 이동범위를 다음과 같이 정의하여 사용한다.

(p,q)의 좌측이동범위: (p,q)가 부품 p의 첫번째 공정이면, 0부터 $t^s(p,q)$ 까지의 범위, 그렇지 않으면 $t^s(p,q-1)$ 부터 $t^s(p,q)$ 까지의 범위

(u,v)의 우측이동범위: (u,v)가 부품 u의 마지막 공정이면, $t^f(u,v)$ 부터 부품 u의 납기 d_u 까지의 범위, 그렇지 않으면 $t^f(u,v)$ 부터 $t^f(u,v+1)$ 범위에서 (u,v)를 가공하는 기계상에서 유휴시간 없이 연속적으로 가공되는 공정중에서 가장 늦은 공정의 완료시간

까지의 범위

P_{pq} 은 (p,q)의 좌측이동범위에서 가공이 시작되는 공정의 집합으로, S_{uv} 는 (u,v)의 우측이동범위에서 가공이 완료되는 공정의 집합으로 정의한다. 이동공정의 후보는 아래와 같이 둔다.

후보이동공정: 늦은공정 (i,j)의 active chain에서 $P_{pq} \neq \emptyset$ 인 임의의 늦은공정 (p,q)와 $S_{uv} \neq \emptyset$ 인 임의의 이른공정 (u,v)를 후보이동공정으로 한다. 그리고 이러한 공정의 집합을 MO_{ij} 로 둔다.

3. 이동방법

이동방법의 주요 개념은 공정납기 정보를 이용하여 늦은공정의 공정납기지연과 이른공정의 조기납기를 서로 상쇄시키는 데 있다. 이때 국부적인 탐색을 통하여 부분적으로 해가 개선될 수 있도록 하고자 한다.

3.1 전방이동방법

늦은공정 (p,q) $\in MO_{ij}$ 의 전방이동방법은 우선 (p,q)가 P_{pq} 내 공정들의 공정순서로 삽입이나 교환 이동되기 전과 후의 (p,q)와 P_{pq} 내 공정들의 총공정납기지연의 변동을 비교한다. (p,q)가 P_{pq} 의 공정 (r,s)의 앞으로 삽입이나 교환이동될 때, 이동전 현일정에서 공정 (p,q)와 (a,b) $\in PI_{pq}^{rs}$ 에 있는 공정들의 공정납기지연의 합, A, 즉 $A = [OT_{pq}]^+ + \sum_{(a,b) \in PI_{pq}^{rs}} [OT_{ab}]^+$

이 된다. 여기서 PI_{pq}^{rs} 는 P_{pq} 의 부분집합으로 (r,s)와 P_{pq} 에 있는 (r,s) 이후의 공정들의 집합이다.

(p,q)가 P_{pq} 의 공정 (r,s)의 앞으로 삽입이나 교환 이동된후, (p,q)의 공정지연의 잠재 감소량, B, 즉 $B = [OT_{pq} - (t^s(p,q) - t^s(r,s))]^+$ 로 나타낼 수 있다.

늦은공정 (p,q)를 (r,s)의 앞으로 삽입이동시킬 경우, 식(1)을 만족하면 공정 (p,q)와 PI_{pq}^{rs} 에 있는 공정들의 총공정납기지연은 감소한다.

$$B + \sum_{(a,b) \in PI_{pq}^{rs}} [OT_{ab} + pt_{pq}]^+ < A \quad (1)$$

늦은공정 (p,q)를 (r,s)와 교환이동시킬 경우 아래 식(2)를 만족하면 공정 (p,q)와 PS_{pq}^{rs} 에 있는 공정들의 총공정납기지연은 감소한다. 여기서 PS_{pq}^{rs} 는 P_{pq} 의 부분집합으로 (p,q)가 P_{pq} 에 있는 공정 (r,s)와 교환이동될 때, P_{pq} 에 있는 (r,s)이후의 공정들의 집합이다.

$$B + [OT_{rs} + (t^s(p,q) - t^s(r,s)) + pt_{pq} - pt_{rs}]^+ + \sum_{(a,b) \in PS_{pq}^{rs}} [OT_{ab} + pt_{pq} - pt_{rs}]^+ < A \quad (2)$$

이와 같이 늦은공정 (p,q)를 삽입, 교환이동할때, 이들 이동중에서 총공정납기지연을 가장 많이 감소시키는 이동을 선택한다.

3.2 후방이동방법

이른공정 (u,v) $\in MO_{ij}$ 의 후방이동방법은 우선

(u,v)가 S_{uv} 내 공정들의 공정순서로 삽입이나 교환 이동되기 전과 후의 (u,v)와 S_{uv} 내 공정들의 총공정납기지연의 변동을 비교한다.

(u,v)가 S_{uv} 의 공정 (r,s)의 뒤로 삽입이나 교환이동될 때, 이동전 현일정에서 공정 (u,v)와 (a,b) $\in SI_{uv}^{rs}$ 의 공정들이 갖는 공정납기지연의 합, C, 즉 $C = [OT_{uv}]^+ + \sum_{(a,b) \in SI_{uv}^{rs}} [OT_{ab}]^+$ 가 된다. 여기서 SI_{uv}^{rs} 는

S_{uv} 의 부분집합으로 (u,v)가 S_{uv} 의 공정 (r,s)의 뒤로 삽입이나 교환이동될 때, S_{uv} 에 있는 (r,s)와 (r,s)이전의 공정들의 집합이다.

공정 (u,v)가 S_{uv} 의 공정 (r,s)의 뒤로 삽입이나 교환이동된후, (u,v)의 공정납기지연의 잠재감소량, D, 즉 $D = [OT_{uv} + (t^s(r,s) - t^s(u,v)) + (pt_{rs} - pt_{uv})]^+$ 로 나타낼 수 있다.

이른공정 (u,v)를 (r,s)의 뒤로 삽입이동시킬 경우 아래 식(3)을 만족하면 공정 (r,s)와 SI_{uv}^{rs} 에 있는 공정들의 총공정납기지연은 감소한다.

$$D + \sum_{(a,b) \in SI_{uv}^{rs}} [OT_{ab} - pt_{uv}]^+ < C \quad (3)$$

이른공정 (u,v)를 (r,s)와 교환이동시킬 경우 아래 식(4)를 만족하면 공정 (u,v)와 SS_{uv}^{rs} 에 있는 공정들의 총공정납기지연은 감소한다. 여기서 SS_{uv}^{rs} 는 S_{uv} 의 부분집합으로 (u,v)가 S_{uv} 에 있는 공정 (r,s)와 교환이동될 때, S_{uv} 에 있는 (r,s)이전의 공정들의 집합이다.

$$D + [OT_{rs} - (t^s(r,s) - t^s(u,v)) + pt_{rs} - pt_{uv}]^+ + \sum_{(a,b) \in SS_{uv}^{rs}} [OT_{ab} + pt_{rs} - pt_{uv}]^+ < C \quad (4)$$

이와 같은 이른공정 (u,v)의 후방이동방법중에서 총공정납기지연을 가장 많이 감소시키는 이동을 선택한다.

4. 재일정계획 방법

앞에서 제시한 이동방법에 의하여 후보이동공정을 이동한 후, 변경된 가공순서를 유지하면서 각 공정이 가장 빨리 시작하는(공정을 left-shift 시킨) 일정은 active일정을 보장하지 못한다. 일정의 순수납기지연을 최소화하는 JSP의 해는 active일정이다 [10]. 따라서 active일정을 위한 재일정계획이 필요하게 된다. 본 연구는 계산시간을 줄이기 위하여 이동 후 active일정이 유지되지 못하는 공정들만을 재일정계획하는 active재일정계획 방법[10]을 사용한다.

5. 발견적기법

제안한 이동방법을 이용한 발견적기법의 절차는 아래와 같다. 제안한 발견적기법을 HOD (Heuristic technique for job shop scheduling Using Operation Due Dates)라 한다. TM은 후보이동공정의

로 선택된후 이동된 공정의 집합을 나타 낸다.

단계 0 : Active 초기가능해를 구하여 현재해로 둔다.

단계 1 : 현재해로부터 낮은공정의 집합 TO 를 구성한다. TM 을 비어 둔다.

단계 2 : (a) $TO = \phi$ 이면 절차를 종료하고, 그렇지 않으면 TO 에서 공정납기지연이 가장 큰 공정 (i,j)를 선택하고 TO 에서 이 공정을 제거한다.

(b) 선택된 공정 (i,j)의 active chain 을 구하고 이 active chain 으로부터 후보이동공정을 모두 구하여 MO_{ij} 로 둔다.

단계 3 : (a) $MO_{ij} = \phi$ 이면, 단계 2 의 (a)로 가고, 그렇지 않으면 MO_{ij} 에서 TM 의 공정이 아니고 가장 늦은 시작시간을 갖는 후보이동공정을 선택하고 MO_{ij} 에서 이 공정을 제거한다. 선택된 공정이 낮은공정 (p,q)이면 단계 3 의 (b)로 가고, 그렇지 않고 이른공정 (u,v)이면 단계 3 의 (c)로 간다.

(b) (p,q)의 전방이동방법에서 (p,q)가 이동가능하면, 가공순서를 변경하고 이를 TM 에 저장한다. 만약 어떤 이동조건도 만족하지 못하면 단계 3 의 (a)로 간다.

(c) (u,v)의 후방이동방법에서 (u,v)가 이동가능하면, 가공순서를 변경 하고 이를 TM 에 저장한다. 만약 어떤 이동조건도 만족하지 못하면 단계 3 의 (a)로 간다.

단계 4 : 기계 상의 가공순서가 변경된 일정에 active 재일정계획 방법을 적용하여 새로운 active 일정을 생성한다.

단계 5 : 생성된 active 일정이 현재해보다 더 좋으면 생성된 active 일정을 현재해로 두고 단계 1 로 가고 그렇지 않으면 단계 3 으로서 간다.

6. 비교분석

기존의 여러 기법들은 문제크기와 납기촉박성 [1, 2, 6, 8]에 의해 계산시간과 해의 성능이 달라짐을 보였다. 따라서 문제크기와 납기촉박성을 각각 세수준으로 분류하여 실험하였다. 실험은 기존의 할당규칙들과 기존의 발견적기법인 MEHA[5]와 GSP[6] 그리고 제안된 발견적기법인 HOD 에 대하여 수행되었다. 실험에 사용한 컴퓨터는 CPU 66Mhz 의 펜티엄이며 계산시간은 CPU 시간이다.

HOD 와 MEHA 에서 초기해는 순수납기지연 최소화 문제에서 좋은 할당규칙으로 알려진 MST, S/RPT, EDD, ODD, MDD, MOD, CEXSPT, HYBRID, CR+SPT 와 S/RPT+SPT 의 10 가지를 사용하여 해를 구한 후, 그 중에서 가장 좋은 해를 사용하였다.

GSP 방법에서는 초기 공정납기를 이용하여 초기해를 생성하고 있다. 따라서 GSP 에서의 초기해는 GSP 에서 제시한 방법을 사용하여 구하였다.

실험을 통하여 납기촉박성의 변화에 따른 해의 개선율과 문제크기의 변화에 대한 계산시간을 비교분석하였다. [표 1]은 문제크기와 납기촉박성에 의해 분류한 18 개 문제에 대해 할당규칙으로 구한 해와 이를 초기해로 한 HOD 의 해를 나타내고 있다. HOD 가 구한해는 문제크기와 납기촉박성에 상관없이 MEHA 와 GSP 보다 좋게 나타났다. 또한 HOD 는 같은 초기해를 사용하는 MEHA 에 비하여 초기해에 따른 영향이 적은것으로 나타났다. 이러한 결과는 MEHA 는 현 일정에서 다른 낮은부품이 발생되면서 일정의 순수납기지연을 줄이는 가능성을 다루지 않았고, 낮은부품의 순수납기지연을 줄이기 위하여 active chain 에 있는 공정들의 가공순서를 변경하지 않음으로 비효율적인 탐색을 한 것으로 보인다. GSP 에서는 각 공정의 좋은 공정납기를 구하여 MOD 를 반복 적용함으로써 극부적인 정보만을 이용하는 할당규칙의 한계를 극복하지 못한 것으로 보인다.

[표 2]은 문제크기와 납기촉박성 변화에 따른 해의 개선율을 비교한 결과를 나타내고 있다. 해의 개선율은 $(a - b) \times 100/a$ 로 정의한다. 여기서 a 는 초기해이고 b 는 발견적기법의 해이다. MEHA 와 HOD 에서 개선율은 각 수준의 문제크기에서 6 문제의 개선율의 평균을 사용하였다. GSP 의 개선율은 초기해에 대하여 음의 개선율을 보인 문제는 고려하지 않았다.

[표 3]은 문제크기와 납기촉박성의 변화에 따른 계산시간을 비교한 결과를 나타내고 있다. 각 기법의 계산시간은 각 수준의 문제크기에서 6 문제의 계산시간 평균을 사용하였다. HOD 의 계산시간은 GSP 보다 더 적게 소요되었지만 MEHA 보다는 약간 많은 계산시간을 소요하였다.

7. 결론

본 연구에서는 순수납기지연 최소화를 목적으로 갖는 job shop 일정계획을 위한 효율적인 발견적 기법을 제안하였다. 제안된 기법 HOD 는 현 일정에서 순수납기지연을 줄일 가능성이 높은 공정을 active chain 에서 선택하여 이를 공정납기 정보를 이용하여 그 기계 상의 적절한 위치로 이동시켜 순수납기지연을 줄이는 방법이다. 실험결과, HOD 에서 해의 개선율은 납기촉박성, 문제크기에 상관없이 MEHA, GSP 보다 좋게 나타났다. 계산시간면에서는 HOD 가 GSP 보다 우수하였고 MEHA 보다는 약간 많은 계산시간을 소요하였다

[표 1] T_{tot} 목적함수에서 할당규칙과 발견적기법의 성능

납기촉박성	기법	낮은				보통				높은				
		초기해	MEHA	HOD	GSP	초기해	MEHA	HOD	GSP	초기해	MEHA	HOD	GSP	
문 제 크 기	10 × 10	1	300	300	194*	311	785	785	717*	773	1901*	1901*	1901*	1901*
		2	125	125	30*	84	946	913	852*	1187	1813	1716*	1748	1848
		3	454	454	392*	454	1346	1346	1274*	1407	1725	1725	1628*	1666
		4	100	100	79*	100	941	941	847*	945	1434	1434	1281*	1508
		5	96	96	77*	165	636	636	548	523*	1879*	1879*	1879*	1923
		6	37	37	6*	105	664	664	563*	618	1597	1597	1585*	1631
	20 × 10	7	1017	1017	804*	1094	2196	2153*	2171	2240	4677	4558	4317*	4430
		8	134	130	92*	130	1625	1529	1237*	1244	3892	3845	3610*	3669
		9	224	224	152*	323	1992	1932	1752	1649*	2836	2835	2356*	3381
		10	1099	1099	820*	1507	2284	2278	2048*	2506	5884	5837	5728*	5994
		11	96	84	0*	204	1144	1144	1123*	1612	3646	3611*	3646	4052
		12	5	5	2*	60	294*	294*	294*	625	2254	2189	2118*	2860
	30 × 10	13	699	681	311*	887	2530	2510*	2530	2586	6801	6801	5912*	7472
		14	971	971	783*	1062	4253	4213	4063*	4208	8076	8076	7990	7847*
		15	453	407	343*	565	2402	2386	2113*	3365	5145	5091*	5145	6672
		16	357	339	259*	449	2063	1975*	2056	3005	7442	7298*	7430	7818
		17	213	190	116*	290	2996	2979	2941	2624*	5906	5737	5548*	7820
		18	915	883	677*	1189	4993	4941	4693	4967*	8969	8871	8737*	9211

*는 각 납기촉박성에서 각 문제에 대한 가장 좋은해를 나타냄.

[표 2] 여러수준의 문제크기와 납기촉박성에서 해의 개선율(%)

납기촉박성	기법	낮은			보통			높은		
		MEHA	HOD	GSP	MEHA	HOD	GSP	MEHA	HOD	GSP
문제	10 × 10	0.0	41.6	5.5	0.6	10.5	4.4	0.9	3.4	0.0
크기	20 × 10	2.6	45.0	0.5	1.9	8.2	6.8	1.4	6.8	1.8
	30 × 10	5.3	33.8	0.0	1.2	4.1	2.3	1.2	3.8	0.5

[표 3] 여러수준의 문제크기와 납기촉박성에서 계산시간(초)

납기촉박성	기법	낮은				보통				높은			
		DR*	MEHA	HOD	GSP	DR	MEHA	HOD	GSP	DR	MEHA	HOD	GSP
문제	10 × 10	0.1	0.1	0.3	2.0	0.1	0.2	0.5	2.9	0.1	0.4	0.3	2.8
크기	20 × 10	0.2	1.1	4.6	6.6	0.2	4.3	4.2	7.4	0.2	6.4	5.3	10.3
	30 × 10	0.2	7.5	18.5	19.5	0.2	19.2	20.0	21.3	0.2	19.0	19.8	24.7

* DR은 할당규칙을 나타냄

참고문헌

- [1] Anderson, E. J. and Nyirenda, J. C., "Two New Rules to Minimize Tardiness in a Job Shop", *International Journal of Production Research*, Vol. 28, No. 2(1990), pp. 2277-2292.
- [2] Baker, K. R., "Sequencing Rules and Due-Date Assignments in a Job Shop". *Management Science*, Vol. 30(1984), 1093-1104.
- [3] Baker, K. R. and Kanet, J. J., "Job Shop Scheduling with Modified Due Dates", *Journal of Operations Management*, Vol. 4(1983), pp. 11-21.
- [4] Conway, R. W., "Priority Dispatching and Job Lateness in a Job Shop", *Journal of Industrial Engineering*, Vol. 16, No. 4(1965), pp. 228-237.
- [5] He, Z., Yang, T. Y., and Deal, D. E., "A Multiple-Pass Heuristic Rule for Job Shop Scheduling with Due Dates", *International Journal of Production Research*, Vol. 31, No. 11(1993), pp. 2677-2692.
- [6] Raman, N. and Talbot, F. B., "The Job Shop Tardiness Problem: A Decomposition Approach", *European Journal of Operational Research*, Vol. 69(1993), pp. 187-199.
- [7] Raman, N., Talbot, F. B., and Rachamadugu, R. V., "Due Date based Scheduling in a General Flexible Manufacturing System", *Journal of Operations Management*, Vol. 8, No. 2(1989), pp. 115-132.
- [8] Schultz, C. R., "An Expediting Heuristic for the Shortest Processing Time Dispatching Rule", *International Journal of Production Research*, Vol. 21, No. 1(1989), pp. 31-41.
- [9] Sun, D., Batta, R., and Lin, L., "Effective Job Shop Scheduling through Active Chain Manipulation", *Computers & Operations Research*, Vol. 22, No. 2(1995), pp. 159-172.
- [10] 김여근, 배상윤, 이덕성, "Job Shop 일정계획을 위한 Tabu Search", *대한산업공학회지*, 제 21 권, 제 3 호(1995), pp. 409-428.