

납기를 갖는 Job Shop 일정계획의 효율적인 발견적기법

배상윤* · 김여근**

An Efficient Heuristic Technique for Job Shop Scheduling with Due Dates

Sang Yun Bae* · Yeo Keun Kim**

Abstract

This paper presents an efficient heuristic technique for minimizing the objectives related to tardiness such as total tardiness, maximum tardiness and root mean square of tardiness in the job shop scheduling. The heuristic technique iteratively improves an active schedule through exploring the schedule's neighborhood, using operation move methods. The move operation is defined on an active chain of tardy jobs in the active schedule. To find the move operations which have a high probability of reducing tardiness, we develop move methods by exploiting the properties of active chains. Our technique is compared with the two existing heuristic techniques, that is, MEHA(Modified Exchange Heuristic Algorithm) and GSP(Global Scheduling Procedure) under the various environments with the three levels of due date tightness and several sized problems. The experimental results show that the proposed technique outperforms the two existing techniques in terms of solution quality and computation time.

1. 서론

본 연구는 m 대의 기계에서 상이한 공정(operation)순서와 가공시간을 갖는 n 개의 부품(job)을 처리하는 job shop에서 납기(due date)가 있는 일정계획(scheduling) 문제(이하 JSP

라 함)를 다루고 있다. 납기는 흔히 제품이 가공완료되어야 하는 미리 정해진 시각으로 정의된다. 공장에서 부품 납기의 지연은, 즉 순수납기지연(tardiness)의 발생은, 납기지연벌금과 재공재고의 증가에 따른 제조원가의 상승과 고객만족도의 저하로 시장 점유율의 감소를 초래한다. 따라서 순수납기지연의 최소화는 총처리

* 전주공업전문대학 공업경영과

** 전남대학교 공과대학 산업공학과

시간(makespan)이나 평균처리시간(mean flow-time)의 최소화 보다 현장에서 더 중요한 목적으로 취급되고 있다[4, 8, 13].

Job shop에서는 자재공급의 착오, 기계고장, 불확실한 가공시간과 납기변경 등의 예기치 않은 상황이 빈번히 발생하므로 실시간(빠른) 일정계획이 요구된다[17]. 이러한 이유로 짧은 계산시간에 일정(schedule)을 산출할 수 있는 할당규칙(dispatching rules)에 관하여 많은 연구가 이루어져 왔다. JSP에서 납기와 관련된 할당규칙은 MST(Minimum Slack Time), S/RPT(Slack per Remaining Processing Time), EDD(Earliest Due Date)와 ODD(Operation Due Date)[7]가 제안된 이후, 이들을 이용하여 MDD(Modified job Due Date)[5]와 MOD(Modified Operation Due date)[6]가 제안되었다. 또한 여러 할당규칙을 조합적용하는 HYBRID[12], CR(Critical Ratio)+SPT(Shortest Processing Time)와 S/RPT+SPT[2], CEXSPT(jobs are Conditionally Expedited by the SPT rule)[14], COVERT(Cost OVER Time)[13]와 ATC(Apparent Tardiness Cost)[16] 등이 제안되었다.

할당규칙은 빠른 시간에 일정을 산출하는 장점이 있지만, 극부적인 정보만을 사용하여 단 한번(single-pass) 일정계획함으로써 좋은 일정을 산출하는 데에는 한계가 있다[10, 11]. 또한 납기의 여러가지 상황에서 어떤 할당규칙도 다른 할당규칙을 지배하지는 못하므로, 가장 유리한 할당규칙을 선택한다는 것은 실제 적용시 어려움이 있다[4, 14].

최근 컴퓨터 성능의 급속한 향상으로 납기를 갖는 JSP에서 할당규칙의 한계를 극복하는 방안으로 발견적기법의 적용이 가능하게 되었다. 발견적기법은 임의의 일정을 초기해로 하여 보

다 좋은 일정을 반복적(multi-pass)으로 구하는 방법으로 기존에 MEHA(Modified Exchange Heuristic Algorithm)[10]와 GSP(Global Scheduling Procedure)[11]가 제안 되어 있다. MEHA에서는 현 일정에서 늦은부품(순수납기지연된 부품)의 보다 빠른 완료시각을 찾기 위하여 다른 늦은부품을 발생시키지 않는 한도에서 이른부품(지연되지 않은 부품)의 완료시각을 늦추고 있다. 이는 현 일정에서 다른 늦은부품이 발생되면서 일정의 전체 순수납기지연이 줄어들 수 있는 가능성을 배제함에 따라 탐색이 불충분하다는 단점이 있다. GSP는 MOD가 비교적 좋은 일정을 산출하고 각 공정의 공정납기를 주는 방법이 목적함수에 영향을 준다는 점에 착안하여 모든 공정의 좋은 공정납기를 차례로 찾아나가는 방법이다. 이는 모든 공정의 공정납기를 구하는 과정에서 많은 탐색시간이 요구된다.

본 연구의 목적은 순수납기지연 최소화를 목적으로 갖는 job shop일정계획을 위한 효율적인 발견적기법을 개발하는데 있다. 제안한 발견적기법에서는 하나의 일정을 생성하여 이 일정에서 공정의 가공순서를 변경하여 해를 개선시켜 나간다. 이 기법에서 문제는 해를 개선시키기 위하여 어떤 공정을 선택하여 어떻게 가공순서를 변경하는 가에 있다. 본 연구에서는 우선 현 일정에서 순수납기지연이 큰 순서로 늦은부품을 선택하고, 이 부품의 완료시각을 결정하는 공정 중에서 시작시각이 늦은 공정부터 차례로 순수납기지연을 줄일 수 있는 공정을 선택한다. 선택한 공정을 그 기계 상의 적절한 위치로 삽입 또는 교환이동하여 변경된 일정의 가공순서를 유지하는 가능해를 생성한다. 여기서 공정을 이동시키는 기본 착상은 늦은부품의 완료시각을 좀더 이르게 하기 위하여

이른부품의 공정의 가공순서를 현 일정에서 뒤로 하는데 있다.

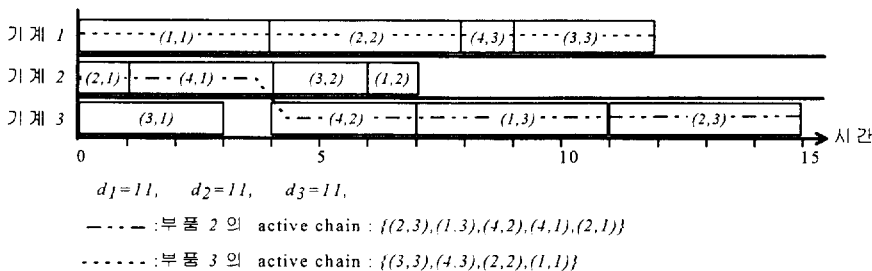
이러한 발전적기법을 개발하기 위하여 본 연구에서는 먼저 이동가능공정의 선택방법과 이공정의 적절한 이동방법(교환 및 삽입)을 제안한다. 그리고 이동에 의해 변경된 일정의 가공순서를 가능한 유지하는 효율적인 active재일정계획 방법을 개발한다.

본 연구에서 사용되는 기호의 정의는 아래와 같다.

- (i,j) : 부품 i 의 j 번째 공정
- $t^s(i,j)$: (i,j) 의 시작시각
- $t^e(i,j)$: (i,j) 의 완료시각
- $m(i,j)$: (i,j) 를 가공하는 기계
- $pm(i,j)$: (i,j) 를 가공하는 기계 상의 직전공정
- $sm(i,j)$: (i,j) 를 가공하는 기계 상의 직후공정
- pt_{ij} : (i,j) 의 가공시간
- c_i : 부품 i 의 가공완료시각
- d_i : 부품 i 의 납기
- $[x]^+$: $\max\{0, x\}$
- T_i : 부품 i 의 순수납기지연, 즉 $T_i = [c_i - d_i]^+$

2. 이동가능공정의 선택

이 장은 active chain을 이용하여, 가공순서를



[그림 1] 늦은부품의 active chain

변경함으로써 효율적으로 순수납기지연을 줄일 수 있는 공정의 선택방법과 선택된 공정의 이동범위를 제안한다.

2.1 Active Chain

Sun *et al.*[15]은 부품의 총처리시간을 최소로 하는 job shop 일정계획을 위한 tabu search 기법에서 active chain의 개념을 이용한 이동방법을 사용하였다. Sun *et al.*은 active일정에서 총처리시간을 줄이기 위한 필요조건은 가장 늦은 완료시각을 갖는 공정의 active chain에 있는 공정들의 가공순서를 변경하는 것임을 지적하였다. Active일정은 어떤 공정도 다른 공정의 지연 없이는 더 빨리 시작될 수 없는 일정이다. 어떤 공정 (i,j) 의 active chain은 active일정에서 이 공정의 완료시각 $t^e(i,j)$ 에 영향을 주는 공정들을 차례로 나열한 것이다. 즉 이 공정부터 차례로 시작시각 $t^s(i,j)$ 을 결정하는 같은 기계의 직전공정, $pm(i,j)$ 나 그 공정의 직선행공정 $(i,j-1)$ 을 찾는 과정을 반복함으로써 active chain을 구할 수 있다[9, 15]. 본 연구에서는 active chain을 공정별로 정의하지 않고 부품별로 정의한다. 즉, 현 일정에서 납기가 지연된 늦은부품의 active chain은 이 부품의 마지막 공정의 active chain을 의미한다.

늦은부품의 active chain의 예를 보자. [그림 1]에서 네 부품의 납기는 $d_i=11, i=1, 2, 3, 4$ 라 하자. 그러면 $T_2=4, T_3=1$ 로 부품 2와 3은 늦은부품이 된다. 늦은부품 2는 $t^s(1,3)$ 이 $t^f(1,2)$ 나 $t^f(4,2)$ 와 같으므로 두개의 active chain을 갖는다. 이들은 $\{(2,3),(1,3),(4,2),(4,1),(2,1)\}, \{(2,3),(1,3),(1,2),(3,2),(4,1),(2,1)\}$ 이다. 본 연구에서는 어떤 늦은부품 i 의 active chain이 여러개 존재하면, 편의상 $pm(i,j)$ 를 선택하여 하나의 active chain만을 생성하기로 한다. 이 경우 늦은부품 2의 active chain은 첫번째 경우가 된다.

2.2 이동가능공정

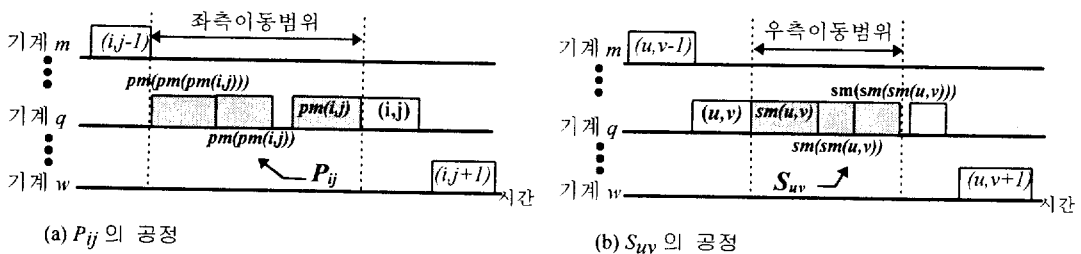
현재의 active일정에서 늦은부품의 완료시각을 더 이르게 하기 위해서는 이 부품의 active chain에 있는 공정들의 가공순서가 변경되어야 한다[15]. 본 연구에서는 순수납기지연을 줄이기 위하여 늦은부품 i 의 active chain에 있는 공정들의 가공순서를 변경한다. 가공순서를 변경하는 공정으로는 이 active chain에서 자신의 공정 (i,j) 나 다른 이른부품 u 의 공정 (u,v) 로 제한한다. 이렇게 공정을 제한하는 것은 중복된 해가 발생되지 않도록 하기 위함이다. 이때 공정 (i,j) 는 현 일정보다 더 빨리 시작할 수 있도록 앞으로 삽입하거나 앞 공정들과 교환하는 이동방법을 사용한다. 이른부품의 공정

인 (u,v) 들은 다른 공정들이 더 빨리 시작할 수 있도록 뒤로 삽입하거나 뒤 공정들과 교환하는 이동방법을 사용한다.

한편 이들 공정 (i,j) 나 (u,v) 를 임의의 위치로 이동한후 가능한 현 공정의 순서를 유지하는 active일정을 생성하면 이동전과 일정이 같을 수 있다. 예로 [그림 1]에서 $(1,3)$ 을 $(4,2)$ 의 앞으로 삽입이동하면 부품 1의 직선행공정, 즉, 공정 $(1,2)$ 의 제약에 의해 공정 $(1,3)$ 의 시작시각은 변함이 없게된다. 그리고 이때 공정 $(4,2)$ 는 다른 공정의 지연 없이 공정 $(1,3)$ 의 앞으로 이동될 수 있게 된다. 따라서 이경우 삽입이동후 active일정을 구하면 이동전과 일정이 같게 된다. 본 연구는 이러한 문제점을 해결하기 위하여 (i,j) 의 이동범위는 현 일정에서 앞으로 이동할 수 있는 좌측이동범위를, (u,v) 의 이동범위는 현 일정에서 뒤로 이동할 수 있는 우측이동범위를 다음과 같이 정의하여 사용한다.

(i,j) 의 좌측이동범위 : (i,j) 가 부품 i 의 첫번째 공정이면, 0부터 $t^f(i,j)$ 까지의 범위, 그렇지 않으면 $t^f(i,j-1)$ 부터 $t^f(i,j)$ 까지의 범위

(u,v) 의 우측이동범위 : (u,v) 가 부품 u 의 마지막 공정이면, $t^f(u,v)$ 부터 부품 u 의 납기 d_u 까지의 범위, 그렇지 않으면 $t^f(u,v)$ 부터 $t^f(u,v+1)$ 범위에서 기계 $m(u,v)$ 상에서 유휴시간 없이 연속적으로 가공되는 공정중에서 가장 늦은 공정



[그림 2] P_{ij} 와 S_{uv} 의 공정

의 완료시각까지의 범위

P_{ij} 는 (i,j) 의 좌측이동범위에서 가공이 시작되는 공정의 집합으로, S_w 는 (u,v) 의 우측이동범위에서 가공이 완료되는 공정의 집합으로 정의한다. [그림 2]에서 P_{ij} 는 $\{pm(i,j), pm(pm(i,j)), pm(pm(pm(i,j)))\}$ 이 되고 S_w 는 $\{sm(u,v), sm(sm(u,v)), sm(sm(sm(u,v)))\}$ 이 된다.

위에서 제시된 늦은부품 i 의 active chain과 이들 공정의 이동범위에 의하여 이동가능공정을 정의하면 아래와 같다.

이동가능공정 : 늦은부품 i 의 active chain에서 $P_{ij} \neq \emptyset$ 인 공정 (i,j) 와 $S_w \neq \emptyset$ 인 임의의 이른부품 u 의 공정 (u,v) 를 이동가능공정으로 한다. 그리고 이러한 공정의 집합을 MO_i 로 둔다.

다음은 이동가능공정이 이동범위에서 이동하면, 해당하는 부품의 active chain에 있는 공정들의 가공순서가 변경됨을 보인다. 늦은부품 i 의 이동가능공정 (i,j) 는 $P_{ij} \neq \emptyset$ 이므로 $t^i(i,j)$ 는 P_{ij} 에 있는 인접한 공정 $pm(i,j)$ 에 의해 결정된다. 또한 (u,v) 는 $S_w \neq \emptyset$ 이므로 $t^i(u,v)$ 는 S_w 에 있는 인접한 공정 $sm(u,v)$ 에 의해 결정된다. 따라서 (i,j) 를 P_{ij} 에 있는 임의의 공정 앞으로 삽입 또는 그 공정과 교환이동하거나, (u,v) 를 S_w 에 있는 임의의 공정 뒤로 삽입 또는 그 공정과 교환이동하면, 부품 i 의 active chain에 있는 공정들의 가공순서가 변경된다.

3. 이동방법

이 장에서는 이동가능공정의 이동방법을 다룬다. 이동방법의 주요 개념은 늦은부품의 순

수납기지연과 이른부품의 조기납기를 서로 상쇄시키는 데 있다. 이때 국부적인 탐색을 통하여 부분적으로 해가 개선될 수 있도록 하고자 한다. 앞으로 늦은부품의 공정은 늦은공정으로, 이른부품의 공정은 이른공정이라 부르기로 한다.

3.1 전방이동방법

늦은공정 $(i,j) \in MO_i$ 의 전방이동방법은 이른공정의 시작시각을 늦추면서 (i,j) 의 이른시작시각을 찾는 방법이다. (i,j) 를 P_{ij} 에 있는 임의의 다른 공정의 앞으로 삽입이동하거나 그 공정과 교환이동하면 시작시각이 늦어지는 다른 공정이 발생하게 된다. 이때 시작시각이 늦어지는 공정이 이른공정이면 일정의 순수납기지연이 줄어들 가능성이 크다. P_{ij}^* 는 P_{ij} 의 부분집합으로 (i,j) 가 P_{ij}^* 에 있는 공정 (r,s) 의 앞으로 삽입이동할 때, (r,s) 와 P_{ij}^* 에 있는 (r,s) 이후의 공정들의 집합으로 정의한다.

늦은공정의 전방이동방법 :

P_{ij} 에 이른공정이 있을 때, P_{ij} 의 공정 중에서 가장 빠른 시작시각을 갖는 이른공정 (u',v') 의 뒤에

(경우 1) 늦은공정이 있으면 (u',v') 와 (i,j) 를 교환이동하고,

(경우 2) 늦은공정이 없으면 (u',v') 바로 앞으로 (i,j) 를 삽입이동한다.

P_{ij} 에 이른공정이 없을 때, 즉 모두 늦은공정이면

(경우 3) P_{ij} 의 공정 중에서 시작시각이 빠른 순으로

$$[T_i - (t^i(i,j) - t^i(r,s))] + \sum_{(a,b) \in P_{ij}^*} (T_a + pt_{ij}) < T_i + \sum_{(a,b) \in P_{ij}^*} T_a \quad (1)$$

을 만족하는 공정 (r,s) 가 존재하면 (i,j) 를 (r,s) 의 앞으로 삽입이동한다.

이 이동방법에서 P_i 에 이른공정이 없고 늦은 공정들이 식(1)을 만족하지 않으면 (i,j) 를 이동하지 않는다.

전방이동방법에서 (경우 1)은 (i,j) 가 P_i 에 있는 이른공정과 교환이동되어 가공순서를 변경함으로써 (i,j) 는 더 빠른 시작시각을, 이른공정은 더 늦은 시작시각을 갖게 된다. (경우 2)는 (i,j) 가 다른 늦은공정에 영향을 주지 않고 이른 공정들 앞으로 삽입이동되어 빠른시작시각을 찾는다. (경우 3)은 P_i 에 이른공정이 없는 경우로서, 식(1)은 일정의 순수납기지연을 줄이는 가능성의 척도로서 사용한다. 식(1)의 우변은 현 일정에서 부품 i 와 $(a,b) \in P_i^s$ 의 부품들이 갖는 순수납기지연의 합을 나타낸다. 식(1) 좌변의 첫째항은 (i,j) 를 삽입이동한 후 부품 i 에 대한 순수납기지연의 잠재감소량을 나타낸다. 좌변의 둘째항은 (i,j) 의 삽입이동으로 시작시각이 늦어지는 공정들에 대한 순수납기지연의 잠재증가량을 나타낸다. 따라서 (i,j) 를 식(1)을 만족하는 (r,s) 의 앞으로 삽입이동하면 국부적으로 부품 i 와 $(a,b) \in P_i^s$ 의 부품들이 갖는 순수납기지연의 합이 이동전보다 감소하게 된다.

3.2 후방이동방법

이른공정 $(u,v) \in MO_i$ 의 후방이동방법은 $t^i(u, v)$ 를 늦추면서 부품 i 의 공정이나 S_w 에 있는 늦은공정의 이른시작시각을 찾는 방법이다. v 를 S_w 에 있는 임의의 공정 뒤로 삽입 또는 그 공정과 교환이동하여 더 늦은 시작시각을 찾으면 시작시각이 빨라지는 다른 공정이 발생하게 된다. 이때 다른 공정이 부품 i 의 공정이거나 적어도 부품 i 의 active chain에 있는 늦은공정일 때 순수납기지연을 줄일 가능성이 있다.

이른공정의 후방이동방법 :

S_w 에 늦은공정이 있을 때,

(경우 1) $sm(u,v)$ 가 늦은공정이면, S_w 에 있는 연속된 늦은공정중에서 가장 늦은공정 뒤로 (u,v) 를 삽입이동하고

(경우 2) $sm(u,v)$ 가 이른공정이면, S_w 에 있는 부품 i 의 공정 중에서 가장 이른시작시각을 갖는 공정과 (u,v) 를 교환이동한다.

이 이동방법에서 S_w 에 늦은공정이 없거나 늦은공정이 있지만 $sm(u,v)$ 가 이른공정이고, S_w 에 부품 i 의 공정이 없으면 (u,v) 를 이동하지 않는다.

후방이동방법에서 (경우 1)은 (u,v) 가 S_w 에 있는 연속된 늦은공정들 뒤로 삽입이동됨으로써 이 늦은공정들이 빠른시작시각을 갖게된다. 이때 이 늦은공정들 중에 부품 i 의 공정이 포함되어 있지 않을 수도 있으나 적어도 하나는 active chain의 공정이므로 부품 i 의 순수납기지연을 줄일 수 있는 가능성이 있다. (경우 2)는 (u,v) 가 S_w 에 있는 부품 i 의 공정과 교환이동됨으로써 부품 i 의 공정은 빠른시작시각을 갖는다.

4. 재일정계획 방법

앞에서 제시한 이동방법에 의하여 이동가능공정을 이동한 후, 변경된 가공순서를 유지하면서 각 공정이 가장 빨리 시작하는(공정을 left-shift 시킨) 일정은 active일정을 보장하지 못한다. 일정의 순수납기지연을 최소화하는 JSP의 해는 active일정이다[3]. 따라서 active일정을 위한 재일정계획이 필요하게 된다. 본

연구는 계산시간을 줄이기 위하여 이동 후 active일정을 유지하지 못하는 공정들 만을 재일정계획하는 active재일정계획 방법을 제시하고자 한다.

재일정계획을 위하여 아래와 같이 공정을 SO , NSO 와 FJ 의 집합으로 구분한다.

SO : 재일정계획을 하지 않는 공정의 집합으로,

(1) 공정 (ij) 의 전방이동인 경우에는, $t'(i, j-1)$ 이전에 가공이 완료되는 공정들의 집합이고,

(2) 공정 (u,v) 의 후방이동인 경우에는 공정 (u,v) 의 이동 전 시작시각 $t'(u,v)$ 이전에 가공이 완료되는 공정들의 집합이다.

NSO : 재일정계획을 필요로 하는 공정들로, SO 에 포함되지 않는 공정들의 집합이다.

FJ : NSO 의 공정중에서 직선행공정이 SO 집합에 포함되어 있는 공정의 집합, 즉

$$FJ = \{(ij) | (ij) \in NSO \text{ and } (ij-1) \in SO\} \text{이다.}$$

본 연구에서는 초기가능해로 active일정을 사용한다. 그리고 공정의 이동후 active일정을 유지함으로써 SO 에 있는 공정들의 현 일정은 active일정이 된다. 따라서 본 연구에서는 SO 에 속하는 공정의 일정을 그대로 두고, NSO 에 대해 현재의 가공순서를 가능한 유지하는 active일정을 생성하고자 한다. σ_{ij} 와 τ_{ij} 는 공정 $(ij) \in FJ$ 의 가공을 각각 가장 빨리 시작할 수 있는 시각과 가장 빨리 완료할 수 있는 시각을 나타낸다.

재일정계획의 절차는 아래와 같다.

〈재일정계획 절차〉

단계 1. 모든 공정을 SO 와 NSO 로 분류한

다.

단계 2. NSO 로 부터 FJ 를 구하고, SO 의 부분일정으로 부터 $(ij) \in FJ$ 의 σ_{ij} , τ_{ij} 를 구한다.

단계 3. $\tau^* = \min_{(ij) \in FJ} \{\tau_{ij}\}$ 를 구하고, τ^* 인 공정을 가공하는 기계 mc^* 를 구한다. τ^* 인 (ij) 가 둘이상이면 임의로 선택한다.

단계 4. $(ij) \in FJ$ 이고 $\sigma_{ij} < \tau^*$ 인 공정 (ij) 중, 기계 mc^* 에서 가장 빠른 가공순서를 갖는 공정 (s,t) 를 선택한다.

단계 5. $t'(s,t) = \sigma_{st}$ 로 두고, $t'(s,t)$ 를 계산한다. (s,t) 를 NSO 와 FJ 에서 삭제한다. 공정 (s,t) 가 부품 s 의 마지막 공정이면 단계 6으로 가고, 그렇지 않으면 FJ 에 $(s,t+1)$ 을 넣는다.

단계 6. $NSO = \emptyset$ 이면 종료하고, 그렇지 않으면 단계 3으로 간다.

제안된 재일정방법은 Baker[3]가 제시한 active일정 생성방법의 변형이다. 위의 절차 중 단계 3에서 공정 $(ij) \in FJ$ 을 할당가능공정으로 함으로써 부품 가공의 순서를 지키고, 단계 4에서 mc^* 에서 가장 빠른 가공순서를 갖는 공정 (s,t) 를 선택함으로써 현재의 일정을 가능한 유지하도록 한다.

5. 발견적기법

본 장에서는 제3장의 이동방법과 제4장의 재일정계획 절차를 이용하여 순수납기지연을 효과적으로 줄일수 있는 발견적기법을 제안한다. 이하 제안한 발견적기법을 HDD(Heuristic technique for job shop scheduling with Due Dates)라 한다. HDD의 절차를 단계별로 보면 아래와 같다.

〈HDD의 절차〉

단계 0. (초기화)

Active초기가능해를 구하여 현재해로 둔다.

단계1. (현재해의 늦은부품 선별)

현재해로 부터 늦은부품의 집합 TJ 를 구성한다.

단계2. (종료기준 검사와 active chain 생성)

(a) $TJ = \emptyset$ 이면 절차를 종료하고, 그렇지 않으면 TJ 에서 순수납기지연이 가장 큰 부품 i 를 선택하고 TJ 에서 이 부품을 제거한다.

(b) 선택된 부품 i 의 active chain을 구하고 이 active chain으로 부터 이동가능공정을 모두 구하여 MO_i 로 둔다.

단계 3. (이동가능공정의 이동)

(a) $MO_i = \emptyset$ 이면, 단계 2의 (a)로 가고, 그렇지 않으면 MO_i 에서 가장 늦은 시작시각을 갖는 이동가능공정을 선택하고 MO_i 에서 이 공정을 제거한다. 선택된 공정이 부품 i 의 공정이면 단계 3의 (b)로 가고 그렇지 않으면 단계 3의 (c)로 간다.

(b) 선택된 공정의 전방이동방법을 검사한다. 이동방법에 의해 가공순서를 변경한다. 만약 어떤 이동조건도 만족하지 못하면 단계 3의 (a)로 간다.

(c) 선택된 공정의 후방이동방법을 검사한다. 이동방법에 의해 가공순서를 변경한다. 만약 어떤 이동조건도 만족하지 못하면 단계 3의 (a)로 간다.

단계 4. (active재일정계획)

기계 상의 가공순서가 변경된 일정에 active재일정계획 방법을 적용하여 새로운 active일정을 생성한다.

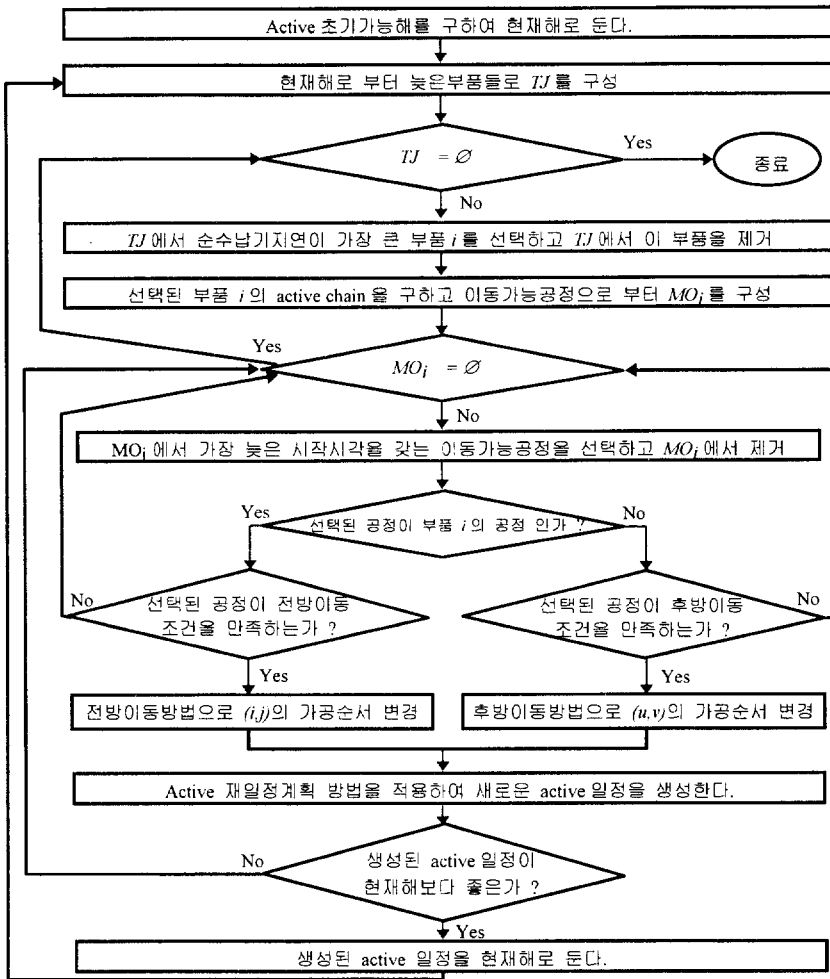
단계5. (현재해와 생성된 해의 비교)

생성된 active일정이 현재해보다 더 좋으면 생성된 active일정을 현재해로 두고 단계 1로

가고 그렇지 않으면 단계 3으로 간다.

제안된 발견적기법은 늦은부품의 순수납기지연과 이른부품의 조기납기를 서로 상쇄하여 해를 개선하므로 순수납기지연이 큰 부품을 먼저 선택할수록 계산시간의 면에서 해를 빨리 개선할 가능성이 높다. 선택된 늦은부품의 active chain에서 시작시각이 늦은 순으로 이동가능공정을 선택한다. 이는 일정의 후반에 계획된 공정의 시작시각을 빠르게 하면 해에 직접적인 영향을 줄 가능성이 크고 탐색초기의 계산시간이 단축될 수 있기 때문이다. 선택된 이동가능공정을 이동방법에 의해 이동한다. Active재일정계획 방법을 적용하여 변경된 기계 상의 가공순서를 가능한 유지하는 새로운 active일정을 생성한다. 해의 개선이 있으면 개선된 해로부터 다시 절차를 반복한다. 현재해의 모든 늦은부품의 active chain에 있는 이동가능공정의 이동에 대하여 해의 개선이 없으면 절차를 종료한다.

HDD의 절차를 흐름도로 보면 [그림 3]과 같다.



[그림 3] HDD의 절차

6. 비교분석

6.1 실험설계

기존의 여러 기법들은 목적함수[8, 13], 문제 크기와 납기촉박성[2, 4, 11, 14, 16]에 의해 계산시간과 해의 성능이 달라짐을 보였다. 따라서 문제크기와 납기촉박성을 각각 세수준으

로 분류하여 실험하였다. 또한 목적함수는 [표 1]에서 제시된 세가지 모두를 사용하였다. 실험은 기존의 할당규칙들과 기존의 발견적기법인 MEHA[10]와 GSP[11] 그리고 제안된 발견적기법인 HDD에 대하여 수행되었다. 실험에 사용한 컴퓨터는 CPU 66Mhz의 펜티엄이며 계산시간은 CPU시간이다.

[표 1] 목적함수 정의

기호	목적함수식	정 의
T_{tot}	$= \sum_{i=1}^n [c_i - d_i]^+$	총순수납기지연(total tardiness)
T_{max}	$= \max_{1 \leq i \leq n} \{[c_i - d_i]^+\}$	최대순수납기지연(maximum tardiness)
T_{rms}	$= (\sum_{i=1}^n ([c_i - d_i]^+)^2 / n)^{1/2}$	평균순수납기지연의 제곱근(root mean square of tardiness)

목적함수로는 총 순수납기지연 T_{tot} , 최대순수납기지연 T_{max} 와 각 부품의 순수납기지연의 편차 T_{rms} 을 최소로 하는 세 목적에 대하여 각 실험하였다.

문제크기로는 10(부품수)×10(공정수, 기계수), 20×10과 30×10인 세 종류를 사용하였다. 공정의 가공시간은 구간 5에서 100을 갖는 일양분포로써 발생시켰다. 각 수준의 문제는 임의로 30개씩 생성하였다.

납기촉박성의 척도로는 EDD할당규칙을 이용한 active일정에서의 총순수납기지연 시간과 부품의 추가공시간의 비율을 사용하였다. 납기촉박성은 세수준, 즉, 높은(tight), 보통(moderate), 낮은(loose) 경우로 분류하고, 각 경우에 대해 이 척도를 대략 40%, 20%, 10%로 두었다. 실험하는 문제에서 각 수준에 적합한 납기를 생성하기 위하여 P , Z , R 매개변수를 갖는 $P(1-Z-1/2R)$ 과 $P(1-Z+1/2R)$ 구간의 일양분포에서 각 부품의 납기를 발생시켰다 [11]. 예비실험을 통해 위에서 언급한 납기촉박성의 척도에 대한 적절한 매개변수 값은 P 는 총처리시간, R 은 0.4로 두고, Z 는 높은, 보통, 낮은 납기촉박성에 대해 각각 0.4, 0.3, 0.2로 두었다.

HDD와 MEHA에서 초기해는 순수납기지연 최소화 문제에서 좋은 할당규칙으로 알려진 MST, S/RPT, EDD, ODD, MDD, MOD,

CEXSPT, HYBRID, CR+SPT와 S/RPT+SPT의 10가지를 사용하여 해를 구한 후, 그 중에서 가장 좋은 해를 사용하였다. 할당규칙을 이용하면 비교적 큰 문제에서도 매우 빠른 시간(수초)에 일정을 산출할수 있으므로 여러 할당규칙으로 일정을 구한 후, 그 중에서 목적에 가장 적합한 일정을 선택하는 것도 한 방법 일수 있다. 위 할당규칙 중에서 ODD, MOD, CEXSPT와 HYBRID에서 공정납기를 주는 방법은 $d_{ij} = d_{ij-1} + pt_{ij} \times d_i / r_i$ 을 사용한다[4]. 여기서 $d_{i0} = 0$ 이고 r_i 은 부품 i 의 추가공시간을 나타낸다. GSP방법에서는 초기 공정납기를 이용하여 초기해를 생성하고 있다. 따라서 GSP에서의 초기해는 GSP에서 제시한 방법을 사용하여 구하였다.

6.2 실험결과 분석

실험을 통하여 각 목적함수와 납기촉박성의 변화에 따른 해의 개선율과 문제크기의 변화에 대한 계산시간을 비교분석하였다. 각 경우에서 30개 문제에 대해 실험하여 얻은 평균을 결과치로 사용하였다.

[표 2]는 총순수납기지연 T_{tot} 문제에서 할당규칙이 구한 초기해와 발견적기법의 해를 나타내고 있다. 해의 개선율은 $(a-b) \times 100/a$ 로 정의한다. 여기서 a 는 납기촉박성과 문제크기의

각 수준에서 가장 좋은 할당규칙의 해이고 b 는 발견적기법의 해이다. 발견적기법들의 개선율은 대체로 납기촉박성이 낮을수록 좋은 결과를 나타내었다. 이는 납기촉박성이 높을수록 해의 개선이 어려움을 나타낸다. HDD의 개선율은 납기촉박성과 문제크기의 전 수준에서 MEHA와 GSP보다 좋게 나타났다. MEHA는 현 일정에서 다른 늦은부품이 발생되면서 일정의 순

수납기지연을 줄이는 가능성을 다루지 않았고, 늦은부품의 순수납기지연을 줄이기 위하여 active chain에 있는 공정들의 가공순서를 변경하지 않음으로 비효율적인 탐색을 한 것으로 보인다. GSP에서는 각 공정의 좋은 공정납기를 구하여 MOD를 반복 적용함으로써 국부적인 정보만을 이용하는 할당규칙의 한계를 극복하지 못한 것으로 보인다.

[표 2] T_{tot} 목적함수에서 할당규칙과 발견적기법의 성능

납기촉박성 문제크기	낮은			보통			높은		
	10×10	20×10	30×10	10×10	20×10	30×10	10×10	20×10	30×10
MST	593	802	1382	1521	3139	5445	2392	6483	11180
S/RPT	615	953	1637	1506	3005	4918	2270	5810	9934
EDD	614	986	1583	1314	2573	4559	2138	5047	8601
ODD	540	1126	2003	1276	3072	6810	2203	6558	12864
MDD	645	988	1671	1303	2442	4432	2020	4789	8191
MOD	490	769	1344	1138	2176	4210	1850	4690	8499
CEXSPT	607	1448	2644	1150	2671	5308	1860	4653	8604
HYBRID	487	770	1260	1182	2220	3776	1924	4447*	8010*
CR+SPT	462*	470*	718*	1135*	2066*	3594*	1847*	4719	8578
S/RPT+SPT	607	1448	2624	1150	2671	5289	1860	4653	8609
MEHA	344	382	657	944	1766	3222	1690	4011	7374
GSP	406	440	762	993	2004	3718	1733	4476	8467
HDD	323	337	599	900	1685	3141	1683	3959	7237
MEHA	25.5	18.7	8.5	16.8	14.5	10.3	8.5	9.8	8.0
개선율 GSP	12.1	6.4	0.0	12.5	3.0	0.0	6.2	0.0	0.0
(%) HDD	30.1	28.3	16.6	20.7	18.4	12.6	8.9	11.0	9.7

*는 할당규칙중 가장 좋은해를 나타냄.

[표 3]은 각 목적함수에서 납기촉박성 변화에 따른 해의 개선율을 비교한 결과를 나타내고 있다. 각 기법에서 개선율은 납기촉박성의 각 수준에 있어서 세수준 문제크기의 개선율의

평균을 사용하였다. HDD에서 개선율은 목적함수와 납기촉박성에 상관없이 MEHA와 GSP보다 좋게 나타났다.

[표 3] 여러수준의 납기촉박성에서 해의 개선율(%)

목적함수	T_{tot}			T_{max}			T_{rms}		
	낮은	보통	높은	낮은	보통	높은	낮은	보통	높은
MEHA	17.6	13.9	8.7	21.4	12.9	5.4	17.1	8.8	6.4
GSP	9.3	5.2	2.1	6.4	0.5	0.0	8.4	4.1	2.5
HDD	25.0	17.3	9.8	29.3	17.4	8.5	25.6	11.8	7.6

[표 4]는 각 목적함수에서 문제크기 변화에 따른 계산시간을 비교한 결과를 나타내고 있다. 각 기법의 계산시간은 각 수준의 문제크기에서 세수준 납기촉박성의 계산시간 평균을 사용하였다. HDD의 계산시간은 MEHA에서 와 거의 비슷하였지만 GSP보다는 더 적게 소

요되었다. 이러한 결과는 MEHA는 해의 탐색 회수는 적지만 공정들의 이동에 대한 계산시간이 많이 소요되고, GSP에서는 모든 공정의 공정납기를 구하는 데 계산시간이 많이 소요된 것으로 보인다.

[표 4] 여러수준의 문제크기에 대한 계산시간(초)

목적함수	T_{tot}			T_{max}			T_{rms}			
	문제크기	10×10	20×10	30×10	10×10	20×10	30×10	10×10	20×10	30×10
MEHA		0.3	3.9	16.6	0.3	3.6	15.2	0.3	3.8	14.5
GSP		2.6	7.7	19.1	5.3	7.8	17.5	4.2	11.7	18.6
HDD		0.4	4.0	16.6	0.4	3.9	15.1	0.5	3.7	13.6

초기해를 구하는 과정에서 기존의 할당규칙의 성능을 부수적으로 분석할수 있었다. 할당규칙은 대체로 목적함수와 납기촉박성에 영향을 받았고 문제크기에는 거의 영향을 받지 않았다. 총순수납기지연 T_{tot} 문제에서는 기존의 연구[2, 4, 6]와 유사한 결과로 공정납기를 사용하고 공정납기가 지연된 공정은 가공시간이 적은 공정을 우선으로 할당하는 MOD, HYBRID와 CR+SPT 등이 비교적 좋은 해를 구하였다. 이러한 경향은 평균순수납기지연의 제곱근 T_{rms} 문제에서도 비슷하게 나타났다. 최대순수납기지연 T_{max} 문제에서는 납기의 긴급도

를 고려하는 MST과 ODD가 좋은 해를 유도하였다.

7. 결 론

본 연구에서 순수납기지연 최소화를 목적으로 갖는 job shop일정계획을 위한 효율적인 발견적기법을 제안하였다. 제안된 기법은 현 일정에서 순수납기지연을 줄일 가능성이 높은 공정을 active chain에서 선택하여 이를 그 기계

상의 적절한 위치로 이동시켜 순수납기지연을 줄이는 방법이다. 또한 이동에 의해 변경된 가공순서를 가능한 유지하는 active재일정계획 방법을 제안하였다. HDD의 성능을 보이기 위하여 세가지 목적함수, 세수준의 납기측박성, 그리고 문제크기의 변화에 대해 기존의 열가지 할당규칙과 두가지 발견적기법을 제안한 기법과 해의 성능 및 계산시간 면에서 비교분석하였다.

실험결과, HDD에서 해의 개선율은 목적함수, 납기측박성, 문제크기에 상관없이 MEHA, GSP보다 좋게 나타났다. 계산시간 면에서는 HDD가 MEHA와 거의 비슷하였으나 GSP보다는 우수함을 보였다.

개발된 HDD는 본 연구에서 다루지 않은 납기를 갖는 JSP의 타 목적함수인 순수납기지연된 부품수(number of tardy job)와 납기지연(lateness)등의 최소화 문제에서도 활용될 것으로 기대된다. 제3장에서 제안된 이동방법은 JSP를 위한 simulated annealing기법에서의 이웃해 생성방법에 적용될 수 있으며, JSP를 위한 유전알고리즘의 유전 연산자(genetic operators) 개발에도 응용될 수 있다.

참 고 문 헌

- [1] Adams J., E. Balas and D. Zawack, "The Shifting Bottleneck Procedure for Job Shop Scheduling", *Management Science*, Vol. 34, No. 3(1988), pp. 391-401.
- [2] Anderson, E. J. and J. C. Nyirenda, "Two New Rules to Minimize Tardiness in a Job Shop", *International Journal of Production Research*, Vol. 28, No. 12(1990), pp. 2277-2292.
- [3] Baker, K. R., *Introduction to Sequencing and Scheduling*, Ch. 2-8, John Wiley & Sons Inc., New York, 1974.
- [4] Baker, K. R., "Sequencing Rules and Due-Date Assignments in a Job Shop". *Management Science*, Vol. 30(1984), pp. 1093-1104.
- [5] Baker, K. R. and J. W. Bertrand, "A Dynamic Priority Rule for Scheduling against Due-Dates". *Journal of Operations Management*, Vol. 3(1982), pp. 37-42.
- [6] Baker, K. R. and J. J. Kanet, "Job Shop Scheduling with Modified Due Dates", *Journal of Operations Management*, Vol. 4(1983), pp. 11-21.
- [7] Conway, R. W., "Priority Dispatching and Job Lateness in a Job Shop", *Journal of Industrial Engineering*, Vol. 16, No. 4(1965), pp. 228-237.
- [8] Fry, T. D., P. R. Philipoom, and J. H. Blackstone, "A Simulation Study of Processing Time Dispatching Rules", *Journal of Operations Management*, Vol. 7, No. 4(1988), pp. 77-92.
- [9] Giffler, B. and G. L. Thompson, "Algorithms for Solving Production Scheduling Problems", *Operations Research*, Vol. 8(1960), pp. 487-503.
- [10] He, Z., T. Y. Yang and D. E. Deal, "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.
- [11] Raman, N. and F. B. Talbot, "The Job Shop Tardiness Problem: A Decomposition Approach", *European Journal of Operational Research*, Vol. 69(1993), pp. 187-199.
- [12] Raman, N., F. B. Talbot and R. V. Rachamadugu, "Due Date based Scheduling in a General Flexible Manufacturing System", *Journal of Operations Management*, Vol. 8, No. 2(1989), pp. 115-132.
- [13] Russell, R. S., E. M. Dae-El and B. W. Taylor III, "A Comparative of the COVERT job Sequencing Rule using Various Shop Performance Measures", *International Journal of Production Research*, Vol. 25, No. 10(1987), pp. 1523-1540.
- [14] 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.
- [15] Sun, D., R. Batta and L. Lin, "Effective Job Shop Scheduling through Active Chain Manipulation", *Computers & Operations Research*, Vol. 22, No. 2(1995), pp. 159-172.
- [16] Vepsalainen, A. P. J. and T. E. Morton, "Priority Rules for Job Shops with Weighted Tardiness Costs", *Management Science*, Vol. 33, No. 8 (1987), pp. 1035-1047.
- [17] Yamamoto, M. and S. Y. Nof, "Scheduling /Rescheduling in the Manufacturing Operating System Environment", *International Journal of Production Research*, Vol. 23, No. 23 (1985), pp 705-722.