

Dispatching Rule based Job-Shop Scheduling Algorithm with Delay Schedule for Minimizing Total Tardiness

Jae-Gon Kim* · June-Young Bang**†

*Incheon National University, Department of Industrial and Management Engineering, Incheon, Korea

**Sungkyul University, Department of Industrial and Management Engineering, Gyeonggi-do, Korea

지연 스케줄을 허용하는 납기최소화 잡샵 스케줄링 알고리즘

김재곤* · 방준영**†

*인천대학교 산업경영공학과

**성결대학교 산업경영공학부

This study focuses on a job-shop scheduling problem with the objective of minimizing total tardiness for the job orders that have different due dates and different process flows. We suggest the dispatching rule based scheduling algorithm to generate fast and efficient schedule. First, we show the delay schedule can be optimal for total tardiness measure in some cases. Based on this observation, we expand search space for selecting the job operation to explore the delay schedules. That means, not only all job operations waiting for process but also job operations not arrived at the machine yet are considered to be scheduled when a machine is available and it is need decision for the next operation to be processed. Assuming each job operation is assigned to the available machine, the expected total tardiness is estimated, and the job operation with the minimum expected total tardiness is selected to be processed in the machine. If this job is being processed in the other machine, then machine should wait until the job arrives at the machine. Simulation experiments are carried out to test the suggested algorithm and compare with the results of other well-known dispatching rules such as EDD, ATC and COVERT, etc. Results show that the proposed algorithm, MET, works better in terms of total tardiness of orders than existing rules without increasing the number of tardy jobs.

Keywords : Job-Shop Scheduling, Total Tardiness, Heuristic Algorithm

1. 서 론

본 연구는 총지연도를 최소화하는 잡샵(job shop) 일정 계획 문제를 다룬다. 본 문제에 대하여 총지연도를 추정하여 총 지연도를 최소화할 수 있도록 하는 디스패칭 기반의 휴리스틱 알고리즘을 제안하였다. 본 연구에서 제안

한 일정계획 방법론은 설비에 할당할 다음 작업을 선택할 때 아직 설비에 도착하지 않은 작업들을 선택 후보에 포함시켜 총지연도 추정치가 최소가 되도록 일정계획을 작성함으로써 지연 일정계획 생성이 가능하도록 하였다.

최근의 시장의 흐름은 소비자의 다양성과 트렌드 변화에 주목하는 기업들이 수익을 안정적으로 확보하고 있다. 이에 따라 각 기업의 생산 현장은 기존의 소품종 대량생산 체계에서 다품종 소량 생산으로 빠르게 전환되어가고 있다. 이를 위하여 생산 공정 또한 컨베이어 벨트 방식의 연속 생산에서 보다 유연한 잡샵 형태로 전환하고 있다. 또한

Received 5 November 2018; Finally Revised 13 December 2018;
Accepted 3 January 2019

† Corresponding Author : jybang@sungkyul.ac.kr

빠르게 변화하는 시장 요구에 대응하기 위하여 주문의 납기 준수는 기업의 이익과 직결되는 중요한 목표가 되었다. 납기를 맞추지 못한 주문은 시장에서 초기 점유율을 확보하지 못하고, 판매되지 못한 상품들은 정가에서 대폭 할인된 가격으로 처분하게 된다. 이러한 변화는 패션 분야의 글로벌 SPA(Specialty retailer of Private label Apparel) 브랜드의 생산 방식 등에서 이미 자리를 잡고 있고, 첨단 기기의 제품 생산 유통 또한 유사한 형태를 따르고 있다.

전산망의 발전으로 정보교환이 용이한 현재의 공급망에서도 공급자의 원자재 품질 및 납기의 불확실성으로 인해 소매적 효과(Bullwhip effect)는 여전히 주요 비용 발생 요인이 되고 있다. 이러한 상황에서 주문의 총지연도를 최소화하기 위한 노력은 공급자 입장에서는 정시에 생산하여 고객에게 저렴한 방식으로 제 시간에 납품할 수 있게 됨으로써 상품 배송 물류비용을 줄일 수 있다. 반대로 주문에 대해서 생산이 늦은 경우에는 생산 지연 물량을 빠른 물류 방식을 선택하여 고객이 요구한 시점에 납품하게 된다. 이때 긴급 물류에 따른 추가 비용이 발생한다. 예를 들어, 정시에 생산을 완료했다면 가격이 저렴한 배편이나 정기 항공편으로 운송 가능하나, 지연 생산 시 긴급 배송을 위하여 임시 항공편 등의 별도의 배송 방안을 위해 인건비와 물류비용을 추가로 지불해야 한다.

위와 같은 납기 준수의 중요도에도 불구하고 총지연도를 최소로 하는 잡샵 일정 계획 문제의 최적해를 찾는 알고리즘에 대해서는 다른 비용함수에 비해 비교적 적은 양의 연구가 진행되어 왔다.

반면, 잡샵 일정계획 문제 중 목적식으로 모든 작업을 종료하는 시간, 즉 makespan을 최소화하는 목적의 문제에 많은 연구가 집중되어 왔다. 현재 주어진 모든 작업이 종료되는 시간을 최소화하는 목적은 단지 빠르게 업무를 종료하는 효과뿐만 아니라 주어진 설비의 가동률을 최대한으로 하기 위한 목적과 일맥상통한다. 이는 장비의 감가상각비용이 매우 큰 장치산업에서 충분한 고객의 수요가 확보된 소품종 대량 생산 체제에 적합한 목표라고 할 수 있다. Chaudhry and Khan[5]이 이러한 표준 잡샵 문제에 대한 방대한 양의 연구들을 정리하였다.

본 연구에서 다루는 작업들의 총지연도와 관련한 문제에 대해서 대부분의 연구들이 최적해를 구하는데 집중하였다. Singer and Pinedo[12]가 총지연도를 최소화하는 분지 한계법을 제안하였다. 이어서 동일한 문제에 대하여 shift bottleneck 방법론을 제안하여 적용하였다[11]. 이와 관련하여 Jayamohan and Rajendran[8]가 관련 연구를 체계적으로 정리하였다.

또한 본 문제는 주로 메타휴리스틱 방법론과 local search 방법론이 많이 적용되었다. Vepsalainen and Morton[13]은 주문의 납기가 정해져 있는 job shop 문제에 대해서 Appa-

rent Tardiness Cost rule(ATC)를 제안하여 지연도를 고려한 디스패칭 규칙 기반의 일정 계획 방법론을 개발하였다. Armentano and Scrich[1]는 동일한 문제에 대해서 tabu search 알고리즘을 개발하여 적용하였다. Mattfeld and Bierwirth[9]이 유전 알고리즘은 일반적인 지연도 지표를 목적식으로 하여 문제에 적용할 수 있도록 개발하였다. de Bontridder[6]의 연구에서 총지연도를 목적식으로 하는 잡샵 문제의 쌍대문제를 기반으로 neighborhood search strategy를 사용하는 tabu search procedure를 제안하였다. Essafi, Mati, and Dauzère-Pérés[7]가 유전 알고리즘과 neighborhood search를 결합한 방법론을 제안하였고, Zhang and Wu[14]는 neighborhood search를 포함한 simulated annealing algorithm을 적용하였다.

Bülül[4]는 tabu search를 shifting bottleneck procedure 안에 적용한 형태의 방법론은 제안하였다. 이와 같이 hybrid 형태의 evolutionary algorithm이 최근 많이 적용되었다[3, 15-17]. 최근에는 가중 총지연도 목적식에 대한 jobshop 문제에 적용할 수 있는 Greedy Randomized Adaptive Search Procedure(GRASP)를 기반 알고리즘이 제안되었다.

위에서 언급된 메타휴리스틱이나 디스패칭 규칙 기반의 알고리즘들은 설비가 특정 작업이 도착할 때까지 다른 작업을 진행하지 않고 대기하는 지연 일정계획을 고려하기 어렵다. 또한 최적해를 구하는 알고리즘은 문제의 크기가 커짐에 따라 기하급수적으로 계산 시간이 증가하는 단점이 있다. 본 연구에서는 잡샵에서 총지연도를 최소화하기 위하여 지연 일정계획을 고려한 디스패칭 기반의 효율적이고 효과적인 알고리즘을 제안하였다.

다음 장에서 지연 일정계획 방법에서 최적해가 존재할 수 있는 성질에 대해서 소개하고 그 효과를 예제로 보여 주었다. 제 3장에서는 총지연도를 추정하는 방법과 본 연구에서 제안하는 일정계획 알고리즘을 기술하였다. 제 4장에서는 실험을 통하여 기존의 납기기반의 디스패칭 규칙과 제안한 알고리즘의 성능을 비교하였고, 제 5장에서 본 논문을 정리하고 제언을 추가하였다.

2. 지연 스케줄링

본 연구에서는 일반적인 잡샵 일정계획 문제에 대하여 총지연도를 최소화하기 위하여 디스패칭 방식의 작업 할당 알고리즘을 제안하였다. 본 연구에서 제안하는 알고리즘은 설비가 작업을 마치고 가용 상태가 되어 다음으로 진행할 작업을 선택할 때, 설비 앞에 대기하고 있는 작업을 포함하여 본 설비에 아직 도착하지 않은 작업들도 선택 후보로 고려한다. 위 작업들 중 임의의 작업이 설비에 할당되었다고 가정할 때 나머지 작업들의 납기

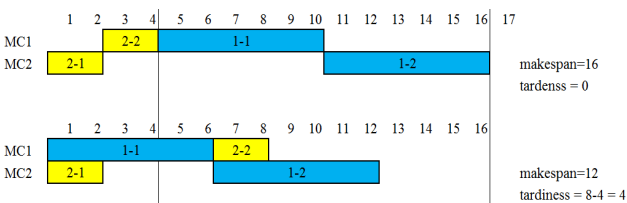
지연도를 추정한다. 이 총지연도의 추정치가 최소가 되는 작업을 선택하여 설비에서 작업을 진행하는 방법을 제안하였다. 아직 도착하지 않은 작업이 선택되는 경우, 설비는 이 작업이 도착할 때까지 다른 작업을 진행하지 않고 대기하게 되는 지연 일정계획이 작성된다.

기존의 우선순위 규칙은 현 시점에서 설비 앞에 대기하고 있는 작업 공정(operation)들 중에서 정해진 규칙에 따라 우선순위를 계산하여 우선순위가 가장 높은 것을 선택하여 지연 없이 바로 설비에 할당하여 진행하게 된다. 대기 작업이 없는 경우를 제외하면 지연 일정계획은 작성되지 않는다.

다음과 같은 간단한 예를 보면 지연 일정계획에서 최적해가 존재할 수 있다. <Figure 1>에서 주문 2의 납기가 주문 1보다 촉박할 경우, 시점 0에서 설비 1에 주문 1의 1번 작업(1-1)을 할당하는 것보다, 주문 2의 1번째 작업이 종료된 후 2-2를 1-1보다 먼저 가공하는 것이 더 적은 총지연도를 보여준다. 하지만 기존의 우선순위규칙은 현 시점에서 장비에 할당 가능한 작업들 중에서 고르기 때문에 작업 1-1이 선택된다. 결과적으로 지연 일정계획이 총지연도 목적에 대해서는 최적해가 되는 현상이 발생한다.

<Table 1> Two-Job Scheduling Example

job	operation	Machine	processing time	due date
1	1	MC1	6	16
	2	MC2	6	
2	1	MC2	2	4
	2	MC1	2	



<Figure 1> Gantt Charts of a Two-Job Scheduling Example

잡샵 문제에 대해서 디스패칭 규칙 기반의 접근 방법은 설비가 가용 상태가 되었을 때, 현재 대기하고 있는 작업(operation)들의 우선순위를 계산하여 가장 높은 우선순위를 갖는 작업을 해당 설비에 할당하여 진행하는 방식이다. 본 연구에서 제안하는 알고리즘은 해당 설비에서 진행해야하는 모든 작업의 우선순위를 계산한다. 즉, 설비 앞에 도착하지 않은 작업의 경우에도 우선순위가 높으면 설비가 해당 작업을 진행하기 위해 유휴 상태로 그 작업이 도착할 때까지 대기하게 된다. 이러한 지연 일정계획 방법론을 flow shop 형태의 반도체 공정 일부에

서 적용하여 우수한 결과를 보였다[2]. 또한 복잡한 생산 라인에서 지연 관리 기법 따른 공정흐름 및 생산성 개선이 됨이 보고되었다[10].

3. 제안 알고리즘

본 장에서는 지연 일정계획을 고려한 방법론에 대하여 서술한다. 본 연구에서 제안하는 알고리즘에서 사용되는 필요한 표기법은 다음과 같다.

Notation

1) Indices

- i, k 주문(Job)의 인덱스, $i, k = 1, \dots, N$
- j, l 작업(operation)의 인덱스
- m 설비의 인덱스, $m = 1, \dots, M$
- h 현재 할당을 고려하는 설비의 인덱스
- b_{ij} 주문 i 의 j 번째 작업을 가공할 때 사용하는 설비의 인덱스
- r_m 가장 최근 시점에 설비 m 에 할당된 주문의 인덱스

2) Parameters

- p_{ij} 주문 i 의 j 번째 operation의 가공 시간
- d_i 주문 i 의 납기
- J_i 주문 i 의 작업의 개수

3) Variables

- a_{ij} 주문 i 의 j 번째 작업이 설비에 할당되었으면 1, 그렇지 않으면 0
- c_{ij} 주문 i 의 j 번째 작업에 대한 가공이 완료되었으면 1, 그렇지 않으면 0
- es_{ij} 주문 i 의 j 번째 작업에 대한 가공을 시작할 수 있는 가능한 가장 빠른 시점
- es'_{ij} 지연 스케줄에서 es_{ij} 의 추정치
- CT 설비에 작업을 할당하고자 하는 현재 시점
- t_m 설비 m 이 진행 중인 작업을 마치고 다음 작업을 시작할 수 있는 가장 빠른 시점, 진행 중인 작업이 없다면, $t_m = CT$
- t'_m 지연 스케줄에서 t_m 의 추정치
- n_i 주문 i 의 작업 중 현재까지 설비에 할당된 작업의 개수
- τ_i 주문 i 의 지연도
- $\hat{\tau}_i$ 주문 i 의 추정 지연도
- $\hat{\tau}$ 총지연도의 추정치, $\hat{\tau} = \sum_i \hat{\tau}_i$
- τ 총지연도

알고리즘을 서술하기 전에 먼저 알고리즘에서 사용되는 중요 용어에 대해서 정의한다. 본 연구에서 제안한 추정 지연도는 설비가 가용이 되었을 때 해당 설비에 할당 가능한 임의의 작업을 선택하여 해당 설비에 할당하는 것으로 가정한 상황에서 다른 작업들의 지연도 추정치로써 다음과 같이 정의한다.

Definition. 작업 k 의 추정 지연도

주문 k 이 현시점 기준으로 이미 늦은 것으로 판단된 경우, 즉, $d_k < es'_{k,J_k} + p_{k,J_k}$ 이면, 주문 k 의 추정 지연도 $\hat{\tau}_k$ 를 다음과 같이 정의한다.

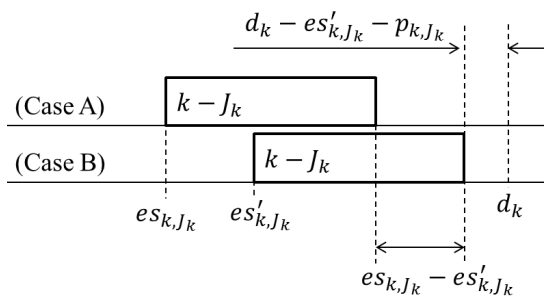
$$\hat{\tau}_k \equiv es'_{k,J_k} + p_{k,J_k} - d_k.$$

또한 주문 k 의 작업 대신 다른 주문의 작업이 이 설비에 할당 될 경우에도 주문 k 가 납기에 늦지 않는 것으로 판단된 경우에는 주문 k 의 여유시간 대비 작업 k 의 마지막 작업 J_k 의 완료 지연시간을 주문 k 의 추정 지연도 $\hat{\tau}_k$ 로 정의한다. 즉, $d_k \geq es'_{k,J_k} + p_{k,J_k}$ 일 때 주문 k 의 추정 지연도는

$$\hat{\tau}_k \equiv \frac{es'_{k,J_k} - es_{k,J_k}}{d_k - es_{k,J_k} - p_{k,J_k} + \delta} \times \mathbb{1} \quad (1)$$

로 정의한다.

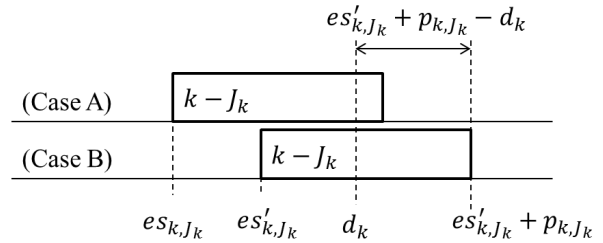
위 정의에서 δ 는 분모가 0이 되는 것을 방지하는 매우 작은 양수이고, $\mathbb{1}$ 은 Dimension이 시간과 일치하도록 도입된 단위 시간 상수(1초, 1분, 1시간 등)이다.



<Figure 2> Illustrations for Schedule of an Untardy Job

<Figure 2>에서는 주문 k 의 작업이 지연 없이 설비에 할당 되었을 경우(Case A)와 지연 일정계획을 고려한 경우(Case B)에 주문 k 의 마지막 작업 J_k 이 주문 k 의 납기 d_k 보다 먼저 종료되는 상황을 간트차트로 도식하였다. 식 (1)에서 분모 부분은 여유시간이고, 분자 부분은 다른 작업이 설비에 할당되어 주문 k 의 작업이 지연될 경우 여유시간의 감소량이다. 즉, 남은 여유시간이 적을수록 지연 시간이 클수록 작업이 지연될 가능성이 높고, 또한

지연도가 커지는 것을 반영하여 이 비율을 아직 지연이 확정되지 않은 작업의 지연도 추정치로 정의한다.



<Figure 3> Illustrations for Schedule of a Tardy Job

<Figure 3>에서는 작업 k 가 지연 없이 설비에 할당되었을 경우(Case A)와 지연 일정계획에서 발생한 경우(Case B)에 주문 k 의 마지막 작업 J_k 이 주문 k 의 납기 d_k 이후에 종료되는 상황을 간트차트로 도식하였다. <Figure 3>의 Case B와 같이 의사결정 시점에 이미 지연 작업으로 확정된 작업 k 에 대해서는 식 (1)로 계산되는 납기 대비 지연된 시간을 총지연도 추정량에 그대로 반영한다. 여기서 작업 k 의 지연도 추정치를 계산할 때 모든 공정이 지연 없이 설비에 할당 된다고 가정하여 종료시점을 추정하므로 실제 일정계획의 결과보다 보수적으로 산정하여 실제보다 적은 값이 된다.

다음은 위의 추정 지연도를 활용하여 지연 일정계획을 작성할 수 있는 알고리즘을 정리하였다.

Algorithm : Minimizing Estimated Tardiness(MET)

Step 1. 초기화

- (1) $c_{i0} \leftarrow 1, es_{i0} \leftarrow 0$ for $\forall i$,
- (2) $a_{ij} \leftarrow 0, c_{ij} \leftarrow 0, es_{ij} \leftarrow es_{i,j-1} + p_{i,j-1}$. for $\forall i, j$
- (3) $t_m \leftarrow 0, n_m \leftarrow 0$ for $\forall m$
- (4) $CT \leftarrow 0, h \leftarrow 0$

Step 2. $t = CT$ 에 대기(idle) 상태인 장비 선택

- (a) $h \leftarrow h + 1$.
- (b) 만약 $t_h \neq CT$ 이면, (a)를 반복한다.
- (c) 집합 $S \leftarrow \{(i, j) | a_{ij} = 0, b_{ij} = h\}$ 로 정의한다.

Step 3. 임시로 할당할 작업과 공정을 선택

S 에서 임의의 원소 (i, j) 를 선택한다. $\hat{\tau}$ 는 매우 큰 수로 설정한다.

Step 4. t'_m 계산

$\forall m$ 에 대해 t'_m 를 다음의 식으로 계산함

- (1) $m = h$ 이면, $t'_m \leftarrow es_{ij} + p_{ij}$
- (2) $m = h$ 이면, $t'_m \leftarrow t_m$

Step 5. es'_{kl} 계산

임의의 작업 k 의 모든 공정 l 에 대해서 es'_{kl} 계산

- (1) $k \neq i$ 이고 $a_{kl} = 0$ 이면, $es'_{kl} \leftarrow \max\{es'_{k,l-1} + p_{k,l-1}, t_m\}$.
여기서 $m = b_{kl}$.
- (2) 그 외의 경우, $es'_{kl} \leftarrow es_{kl}$.

Step 6. 작업 k 의 공정 l 를 설비에 할당할 때 지연도의 추정치 $\hat{\tau}_k$ 계산

임의의 작업 k 에 대하여 다음과 같이 $\hat{\tau}_k$ 를 계산함.

- (1) $d_k < es'_{k,J_k} + p_{k,J_k}$ 이면, $\hat{\tau}_k \leftarrow es'_{k,J_k} + p_{k,J_k} - d_k$.
- (2) 그 외의 경우,

$$\hat{\tau}_k \leftarrow \frac{es'_{k,J_k} - es_{k,J_k}}{d_k - es'_{k,J_k} - p_{k,J_k} + \delta} \times \mathbb{1}$$

여기서 δ 는 매우 작은 양의 수이고, $\mathbb{1}$ 은 단위 시간(1초, 1분, 1시간 등)이다.

Step 7. $\hat{\tau}_k < \sum_{k=1}^N \hat{\tau}_k$ 이면, $(i^*, j^*) \leftarrow (i, j)$ 으로 설정한다.
 $\mathbf{S} \leftarrow \mathbf{S} - \{(i, j)\}$. $\mathbf{S} \neq \emptyset$ 이면, Step 3으로 이동한다.

Step 8. (할당할 공정에 대한 정보 갱신)

$c_{i^*,j^*-1} = 0$ 이면, 즉, (i^*, j^*) 의 직전 공정이 아직 종료되지 않은 경우, **Step 9**로 이동한다.

그 외의 경우, 다음의 정보를 갱신한다.

- (1) $j^* \leftarrow n_{i^*} + 1$ (작업 i^* 의 공정 중 다음 할당해야 할 공정 인덱스), $h^* \leftarrow b_{i^*,j^*}$
- (2) $a_{i^*,j^*} \leftarrow 1$, $t_{h^*} \leftarrow es_{i^*,j^*} + p_{i^*,j^*}$, $r_{h^*} \leftarrow i^*$
- (3) $j^* = J_{i^*}$ 이면, $\tau \leftarrow \tau + \min\{t_{h^*} - d_{i^*}, 0\}$
- (4) 모든 작업 k 과 공정 l 에 대해서, $a_{kl} = 0$ 이면, $m \leftarrow b_{kl}$,
 $es_{kl} \leftarrow \max\{es_{k,l-1} + p_{k,l-1}, t_m\}$.

Step 9. (CT 변경)

- (1) CT 를 CT 시점 이후에 설비가 대기 상태가 되는 가장 가까운 시점으로 변경한다. 즉, $CT \leftarrow \min_m \{t_m \mid t_m > CT\}$.
- (2) 모든 설비 m 에 대해서, $t_m < CT$ 이면 $t_m \leftarrow CT$,
 $C_{r_m, n_{r_m}} \leftarrow 1$.

Step 10. (종료 조건 확인)

모든 공정이 설비에 할당되었다면, 즉, $\sum_{i=0}^N n_i = \sum_{i=0}^N J_i$ 이면, 본 알고리즘을 종료한다. 그렇지 않은 경우에는 $h \leftarrow 0$ 로 설정한 후 **Step 2**로 이동한다.

본 알고리즘에서 작업 i 의 공정 j 의 일정계획은 공정-설비 할당 관계 b_{ij} 와 es_{ij} 로 구할 수 있다. 또한 목적값인 총지연도는 τ 값이 된다.

본 연구에서 제안한 MET 알고리즘을 생산 현장에서 보편적으로 사용한 납기기반의 디스패칭 규칙들인 EDD, Slack, MDD, COVERT, ATC와 성능을 비교하였다. 각 디스패칭 규칙을 요약하면 다음과 같다.

$$\text{EDD} \quad \min_i(d_i)$$

$$\text{Slack} \quad \min_i\{d_i - CT - \rho_i\}$$

$$\text{MDD} \quad \min_i\{\max\{d_i, CT + \rho_i\}\}$$

$$\text{COVERT} \quad \max_i\left\{\frac{1}{\rho_i} \max\left[0, 1 - \frac{\max(0, d_i - CT - \rho_i)}{k\rho_i}\right]\right\}$$

$$\text{ATC} \quad \max_i\left\{\frac{1}{\rho_i} \exp\left(\frac{-\max(0, d_i - CT - \rho_i)}{\kappa\rho_i}\right)\right\}$$

여기에서 작업 i 의 잔여 작업량을 다음으로 정의한다.

$$\rho_i \equiv \sum_{j=n_{i+1}}^{J_i} p_{ij}$$

ATC rule에서 κ 는 사전 테스트를 통해 정하는 파라미터이고, $\bar{\rho}_i$ 는 대기하고 있는 작업의 잔여 작업시간의 평균치이다.

4. 실험결과 및 분석

본 연구에서 제안한 일정계획 알고리즘 MET의 성능을 평가하기 위하여 각 조건에 따라 임의의 잡샵 일정계획 문제 300개를 생성하였다. 생성된 실험 문제는 5개 수준의 작업 개수(10, 20, 30, 40, 50개), 3개 수준의 납기 tightness factor(loose, normal, tight)로 하여 15개의 조합으로 구성하였다. 각 실험 조합에 대해서 20개씩의 반복 실험을 시행하여 총 300개의 문제에 대해서 실험을 진행하였다. 각 실험 문제에서 관련 데이터는 다음의 파라미터로 확률적으로 생성하였다. 여기서 $DU(a, b)$ 는 $[a, b]$ 구간에서 이산균등분포, $U(a, b)$ 는 $[a, b]$ 구간에서 균등분포로부터 확률적으로 얻은 값으로 표현하였다. 실험 문제 생성 조건을 정리하면 다음과 같다.

- 1) 작업의 개수 : 10, 20, 30, 40, 50개(5 수준)
- 2) 납기의 긴급도 수준 : loose, normal, tight(3 수준)
- 3) 반복횟수 : 20회

- 4) 설비의 수 : $M = 3 \times \text{작업의 개수} / 10$
- 5) 작업 i 의 공정 개수 : $J_i = \min\{M, DU(1, 10)\}$
- 6) 작업 i 의 공정 j 의 가공시간 $p_{ij} \sim DU(1, 20)$
- 7) 작업 i 의 공정 j 를 가공하는 설비의 인덱스 : $b_{ij} \sim DU(1, M)$. 단, 작업 i 는 동일 설비를 재방문하지 않도록 한다. 즉, 임의의 작업 i 에 대하여 $j \neq k$ 이면, $b_{ij} \neq b_{ik}$ 이다.
- 8) 작업 i 의 납기 d_i 는 다음 식으로 생성한다.

$$d_i \sim \left[\sum_{j=1}^{J_i} p_{ij} \times U(1, dfactor) \right]$$

여기서, $\lfloor x \rfloor$ 는 x 보다 작은 정수이고, $dfactor$ 는 $\{3, 5, 7\}$ 의 값 중 하나를 동일한 확률로 가지며, 이 값은 납기의 긴급도인 $\{\text{tight, normal, loose}\}$ 에 각각 대응한다.

위의 실험 문제를 대상으로 본 연구에서 제안한 지연을 고려한 일정계획 알고리즘 MET와 실제 현장에서 많이 사용하는 디스패칭 기반의 알고리즘 EDD, Slack, MDD, COVERT, ATC와 비교하였다.

<Table 2>와 <Table 3>은 MET 알고리즘과 나머지 규칙과의 총지연도에 대한 Relative Deviation Index(RDI) 값을 비교한 것이다. 각 알고리즘 A 의 RDI 값은 $(T_{worst} - T_A) / (T_{worst} - T_{best})$ 로 정의한다. 여기에서 T_{worst} 는 동일 문제 결과에서 가장 큰 총지연도를 보여준 알고리즘의 총지연도, T_{best} 는 해당 문제 결과에서 가장 적은 총지연도를 보여준 알고리즘의 총지연도, T_A 는 알고리즘 A 의 해당 문제 결과에서 총지연도를 의미한다.

<Table 2> Test Results of Priority Rules with Respect to Number of Jobs : Total Tardiness

Number of jobs	EDD	SLACK	MDD	COVERT	ATC	MET
10	0.43	0.44	0.50	0.64	0.59	0.82
20	0.40	0.60	0.42	0.70	0.63	0.70
30	0.23	0.51	0.25	0.67	0.60	0.72
40	0.21	0.50	0.22	0.67	0.55	0.68
50	0.26	0.48	0.29	0.65	0.58	0.63
Average	0.31	0.50	0.33	0.67	0.59	0.71

<Table 2>에서 주문의 수에 따라서 전반적으로 MET 알고리즘이 좋은 결과를 주는 것을 확인할 수 있다. 특히 주문의 수가 적을 때(10개~30개)에 MET 알고리즘의 성능이 다른 규칙에 비해 월등한 성능을 보여준다. 주문의 수가 많은 경우(40개~50개)에서는 COVER 규칙과 MET 알고리즘이 유사한 성능을 보여 준다. 이는 주문의 수가

적은 경우에 지연 일정 계획이 최적 해가 될 가능성이 높으며, 주문의 수가 많으면 지연 일정계획 효과가 어느 정도 희석되는 것으로 판단된다.

<Table 3> Test Results of Priority Rules with Respect to Levels of Due Date Tightness of Jobs : Total Tardiness

Tightness	EDD	SLACK	MDD	COVERT	ATC	MET
Loose	0.35	0.54	0.33	0.57	0.44	0.92
Normal	0.25	0.52	0.22	0.71	0.62	0.72
Tight	0.33	0.45	0.46	0.72	0.70	0.49
Average	0.31	0.50	0.33	0.67	0.59	0.71

<Table 3>에서 보여주는 결과와 같이 주문들의 납기가 다양하게 분포되고 긴급도 수준이 낮을 때에 제안한 알고리즘의 성능이 좋은 것으로 나타난다. 그러나 주문의 긴급도가 Tight한 경우에는 COVERT 규칙이 우위에 있다. 이는 대부분의 주문이 납기를 만족시킬 수 없는 경우에는 지연 일정 계획이 오히려 성능을 저하시키는 것으로 판단된다.

<Table 4>와 <Table 5>는 MET 알고리즘과 나머지 rule을 적용한 경우에 대해서 전체 주문들 중 지연된 주문 수의 비율인 지연 주문 비율을 비교한 것이다. 이 목적 함수에 대해서는 MDD가 가장 좋은 결과를 보이고 SLACK이 가장 나쁜 결과를 보인다. 주문의 수가 적거나 납기에 충분한 여유가 있는 경우에만 MET 알고리즘의 성능이 우위에 있는 것으로 나타났다.

<Table 4> Test Results of Priority Rules with Respect to Number of Jobs : Percentage of Tardy Jobs

Number of jobs	EDD	SLACK	MDD	COVERT	ATC	MET
10	39.17	45.17	35.17	37.00	36.50	34.17
20	34.75	40.08	32.50	34.08	34.75	34.17
30	36.83	43.39	32.94	35.89	35.56	37.17
40	34.13	39.71	32.08	34.71	35.17	36.04
50	37.00	43.40	34.43	37.07	37.03	38.50
Average	36.38	42.35	33.43	35.75	35.80	36.01

<Table 5> Test Results of Priority Rules with Respect to Levels of Due Date Tightness of Jobs : Percentage of Tardy Jobs

Tightness	EDD	SLACK	MDD	COVERT	ATC	MET
Loose	12.77	14.00	12.24	11.98	12.75	8.81
Normal	28.31	33.37	25.38	26.31	25.97	29.01
Tight	68.05	79.68	62.65	68.97	68.69	70.21
Average	36.38	42.35	33.43	35.75	35.80	36.01

또한 <Table 5>의 지연 작업 비율에 관한 분산분석 결과에서 따르면 지연 작업의 비율의 평균의 차이가 유의미한 것으로 나타났다. 그러나 <Table 6>의 Pair-wise t test를 결과를 보면, SLACK 규칙만 통계적으로 유의미하게 낮은 성능을 보여 주는 반면에 나머지 알고리즘들의 지연 주문 비율이 통계적으로 유의미하게 차이가 있지 않는 것으로 나타난다. 즉, 지연 주문 비율의 목적식에 대해서는 성능 우위가 있는 알고리즘을 특정할 수 없는 것으로 보인다.

<Table 5> ANOVA Result for Percentage of Tardy Jobs

	Df	Sum Sq	Mean Sq	F value	Pr(> F)
rules	5	13,468	2,693.6	3.5734	0.003217**
Residuals	1,794	1,352,311	753.8		

<Table 6> Pairwise t-test Result among Rules for Percentage of Tardy Jobs

	ATC	COVERT	EDD	MDD	MET
COVERT	0.9816	-	-	-	-
EDD	0.7980	0.7802	-	-	-
MDD	0.2894	0.3000	0.1884	-	-
MET	0.9264	0.9081	0.8701	0.2494	-
SLACK	0.0035*	0.0033*	0.0078	7.10E-05*	0.0047

따라서, 제안된 MET 알고리즘은 합리적인 납기가 설정되어 있는 경우에 총지연도에서 다른 알고리즘에 비해 좋은 결과를 보여 준다. 또한, 지연 주문의 비율 지표에 대해서는 다른 알고리즘에 대해서 성능저하 없는 결과를 보여주는 것으로 확인되었다.

5. 결론 및 추후 연구

본 연구에서는 총지연도를 최소화하기 위한 목적의 일반적인 잡샵 문제를 다루었다. 이 문제에 대해서 할당할 수 있는 작업이 대기하고 있는 상황에서도 다른 작업을 진행하기 위해 설비가 대기하는 지연 일정계획에서 최적 해가 나타날 수 있음을 보였다. 이를 기반으로 빠르고 효율적인 솔루션으로 지연 일정계획을 고려한 디스패칭 규칙 기반의 알고리즘을 제안하였다. 기존의 디스패칭 규칙 기반의 스케줄링 알고리즘은 대기하고 있는 작업의 우선순위를 계산하여 일정계획을 수립하기 때문에 지연 일정계획이 해에서 배제되어 있다. 본 연구에서 개발한 알고리즘에서는 임의의 작업이 설비에 할당될 때 총지연도의 추정치를 예측하여 총지연도가 최소화 될 수 있도록 작업을 선택한다. 이때 아직 도착하지 않은 작업

도 설비에 할당할 대상에 포함시켜 지연도의 추정치를 계산함으로써 이런 작업이 지연도를 최소화하는 것으로 선택되었을 때는 지연 일정계획이 생성될 수 있다.

본 연구에서 제안한 알고리즘의 성능을 평가하기 위하여 다양한 문제에 대해서 일반적으로 총지연도를 최소화하는데 사용하는 디스패칭 규칙들과 결과를 비교하는 실험을 수행하였다. 총지연도 지표에 관하여서는 다른 디스패칭 규칙들보다 우수한 결과를 보였다. 동시에 지연된 작업의 수에 대해서는 통계적으로 유의미한 차이가 없는 결과가 도출되었다. 이는 지연 작업 비율을 유지하면서 총지연도를 효과적으로 감소시킨 것으로 분석할 수 있다.

본 연구에서 제시한 디스패칭 알고리즘을 실제 상황에 적용하기 위해서는 작업시간 정보, 현재 진행 중인 작업 상태 정보 등과 같은 기준 정보와 실시간 운영 정보의 정확성 확보가 선행되어야 한다. 또한 제안된 알고리즘에서 계산하는 총지연도의 추정치에 대하여 현장의 특성을 반영한 다양한 후보군을 만들어 평가 및 개선하는 연구가 수반되어야 한다.

Acknowledgement

This work was supported by the Incheon National University (International Cooperative) Research Grant in 2018.

References

- [1] Armentano, V.A. and Scrich, C.R., Tabu search for minimizing total tardiness in a job shop, *International Journal of Production Economics*, 2000, Vol. 63, No. 2, pp. 131-140.
- [2] Bang, J.-Y., Two-Level Hierarchical Production Planning for a Semiconductor Probing Facility, *Journal of Society of Korea Industrial and Systems Engineering*, 2015, Vol. 38, No. 4, pp. 159-167.
- [3] Braune, R., Zapfel, G., and Affenzeller, M., Enhancing local search algorithms for solving job shops with min-sum objectives by approximate move evaluation, *Journal of Scheduling*, 2013, Vol. 16, No. 5, pp. 495-518.
- [4] Bulbul, K., A hybrid shifting bottleneck-tabu search heuristic for the job shop total weighted tardiness problem, *Computers and operations Research*, 2011, Vol. 38, No. 6, pp. 967-983.
- [5] Chaudhry, I.A. and Khan, A.A., A research survey : review of flexible job shop scheduling techniques, *International Transactions in Operational Research*, 2016,

- Vol. 23, No. 3, pp. 551-591.
- [6] de Bontridder, K.M.J., Minimizing total weighted tardiness in a generalized job shop, *Journal of Scheduling*, 2005, Vol. 8, No. 6, pp. 479-496.
- [7] Essafi, I., Mati, Y., and Dauzere-Peres, S., A genetic local search algorithm for minimizing total weighted tardiness in the job-shop scheduling problem, *Computers and Operations Research*, 2008, Vol. 35, No. 8, pp. 2599-2616.
- [8] Jayamohan, M.S. and Rajendran, C., New dispatching rules for job shop scheduling : A step forward, *International Journal of Production Research*, 2000, Vol. 38, No. 3, pp. 563-586.
- [9] Mattfeld, D.C. and Bierwirth, C., An efficient genetic algorithm for job shop scheduling with tardiness objectives, *European Journal of Operational Research*, 2004, Vol. 155, No. 3, pp. 616 - 630.
- [10] Park, K.M. and Jeong, S.J., A study on the improvement of work flow and productivity in complex manufacturing line by employing the effective process control methods, *Journal of the Korea Academia-Industrial Cooperation Society*, 2016, Vol. 17, No. 5 pp. 305-315.
- [11] Pinedo, M.L. and Singer, M., A shifting bottleneck heuristic for minimizing the total weighted tardiness in a job shop, *Naval Research Logistics*, 1999, Vol. 46, No. 1, pp. 1-17.
- [12] Singer, M. and Pinedo, M.L., Computational study of branch and bound techniques for minimizing the total weighted tardiness in job shops, *IIE Transactions*, 1998, Vol. 30, pp. 109-119.
- [13] Vepsalainen, A.P.J. and Morton, T.E., Priority rules for job shops with weighted tardiness costs, *Management Science*, 1987, Vol. 33, No. 8, pp. 1035-1047.
- [14] Zhang, R. and Wu, C., A neighbourhood property for the job shop scheduling problem with application to hybrid particle swarm optimization, *IMA Journal of Management Mathematics*, 2013, Vol. 24, No. 1, pp. 111-134.
- [15] Zhang, R. and Wu, C., A PMBGA to optimize the selection of rules for job shop scheduling based on the Giffler-Thompson algorithm, *Journal of Applied Mathematics*, 2012, Vol. 2012, pp. 1-17.
- [16] Zhang, R. and Wu, C., A simulated annealing algorithm based on block properties for the job shop scheduling problem with total weighted tardiness objective, *Computers and Operations Research*, 2011, Vol. 38, No. 5, pp. 854-867.
- [17] Zhang, R., Song, S., and Wu, C., A hybrid artificial bee colony algorithm for the job shop scheduling problem, *International Journal of Production Economics*, 2013, Vol. 141, No. 1, pp. 167-178.

ORCID

Jae-Gon Kim | <http://orcid.org/0000-0002-4821-4441>

June-Young Bang | <http://orcid.org/0000-0003-4999-9161>