

재작업이 존재하는 이종병렬기계에서 생산효율을 위해 공정소요시간 단축을 목적으로 하는 작업할당

서정하¹ · 고효현² · 김성식³ · 백준걸^{3*}

¹삼성전자 제조기술센터 / ²고려대학교 정보통신기술연구소 / ³고려대학교 산업경영공학부

Dispatching to Minimize Flow Time for Production Efficiency in Non-Identical Parallel Machines Environment with Rework

Jung-Ha Seo¹ · Hyo-Heon Ko² · Sung-Shick Kim³ · Jun-Geol Baek³

¹Global Production Technology Center, Samsung Electronics, Suwon 443-742, Korea

²Research Institute for Information and Communication Technology, Korea University, Seoul, 136-701, Korea

³School of industrial management engineering, Korea University, Seoul, 136-713, Korea

Reducing waste for the efficiency of production is becoming more important because of the rapidly changing market circumstances and the rising material and oil prices. The dispatching also has to consider the characteristic of production circumstance for the efficiency. The production circumstance has the non-identical parallel machines with rework rate since machines have different capabilities and deterioration levels in the real manufacturing field. This paper proposes a dispatching method, FTLR (Flow Time Loss Index with Rework Rate) for production efficiency. The goal of FTLR is to minimize flow time based on such production environments. FTLR predicts the flow time with rework rate. After assessing dominant position of expected flow time per each machine, FTLR performs dispatching to minimize flow time. Experiments compare various dispatch methods for evaluating FTLR with mean flow time, mean tardiness and max tardiness in queue.

Keywords: Dispatching, Non-identical Parallel Machine, Rework, Flow Time

1. 서론

최근 기업 내외적으로 생산효율 향상에 대한 요구가 증가하고 있다. 고공행진 중인 유가와 원자재 가격의 상승이 기업 내부적으로 생산 비용 부담으로 작용하여 이윤이 축소되고 있다. 따라서 기업은 생산효율을 위해 최소자원으로 생산목적을 달성해야 하므로 생산 과정의 낭비요소 제거에 노력하고 있다. 반면, 시장에서 제품의 수명주기는 소비가 고도화되고 고객의 요구가 다양해지면서 점점 짧아졌다. 시장변화에 따라 기업은 고객이 원하는 시점에 원하는 제품을 공급하기 위해서 다양한 제품을 신속하고 효율적으로 공급해야 경쟁력을 갖출 수 있게 되었다. 따라서 기업은 생산효율을 요구하는 내외부적인 환경

에 맞춰서 생존과 경쟁력 창출을 위해 기술 및 설비투자, 품질 관리와 생산 관리의 개선에 노력하고 있다. 그러나 기술과 설비에 대한 투자는 고객의 투자비가 필요하며 투자비 회수에 장시간이 소요된다. 하지만 생산효율을 위한 품질 및 생산 관리는 큰 투자 없이 적은 비용으로 가능하며 시장 환경 변화에 신속한 대응이 가능하다. 따라서 투자에 따른 효율 증대보다 관리를 통한 생산 효율 증대가 기업에게는 더욱 절실하며 기업은 관리를 통한 생산효율 달성에 더욱 중점을 두고 있다. 예로, 반도체 공장은 신 설비의 구매와 라인 증설로 생산 능력을 증대시켰던 기존의 경쟁방식에서 벗어나고 있다. 특히, 기존 설비의 개조 및 공정 개선을 통해 불량 감소와 Cycle Time 단축을 달성하는 저비용 고효율의 경쟁 방식으로 전환하고 있다.

이 논문은 2011년도 정부(교육과학기술부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(No. 2011-0002528).

* 연락저자 : 백준걸 교수, 136-701 서울시 성북구 안암동 5가 고려대학교 산업경영공학부, Tel : 02-3290-3396, Fax : 02-929-5888,

E-mail : jungeol@korea.ac.kr

2011년 4월 19일 접수; 2011년 11월 2일 게재 확정.

또한, 생산효율을 위해 생산능력을 증대시키는 작업할당이 매우 중요하게 대두되고 있다. 즉, 동일한 작업이 더 빠른 시간에 처리될 수 있도록 하는 할당 기준을 요구하고 있다. 하지만 기업은 실제 공정 환경을 충분히 고려하지 못한 작업할당으로 인해, 다양한 관리와 노력에도 불구하고 작업투입 대비 실질적 효율성과를 크게 거두지 못하고 있다.

<Figure 1>은 일반적인 공정 환경 특성을 보여준다. 공정은 다양한 공정 특성으로 인해 <Figure 1>과 같이 가공시간, 준비시간 및 재작업확률의 작업할당 조건(Job Dispatching Conditions)이 모든 설비에서 서로 달라진다. 특히, <Figure 1>에서 나열한 설비 특성(Machine Characteristics)처럼 설비 본연의 성능(Specification)이나, 생산적합제품(Main Machining Model), 노후화 정도(Deterioration)가 동일 제품에 대해서도 가공시간, 준비 시간 및 재작업확률을 상이하게 만든다. 또한 설비 특성 중 연속한 제품의 가공유사도(Machining Similarity between Models)에 따라 가공되는 제품의 순서는 작업준비시간에 영향을 준다. 특히, <Figure 1>과 같은 공정 환경 특성은 실제적 공정 특성이며, 많은 제조공정 중 압축기 제품의 핵심 구성품인 Pump를 구성하는 부품의 제작공정이 좋은 예라고 할 수 있다. Pump 부품의 제작공정은 여러 대의 Milling Machine으로 구성되고, 주조품이 Milling Machine에서 정밀하게 절삭되어 가공부품이 된다. 가공부품의 규격은 엄격하기 때문에 검사를 통하여 규격에 맞지 않는 가공부품은 Milling Machine에 재투입되어 재작업 된다. 또한 연속된 가공부품의 가공유사도에 따라 Milling Machine 사용을 위한 조정정도가 달라진다.

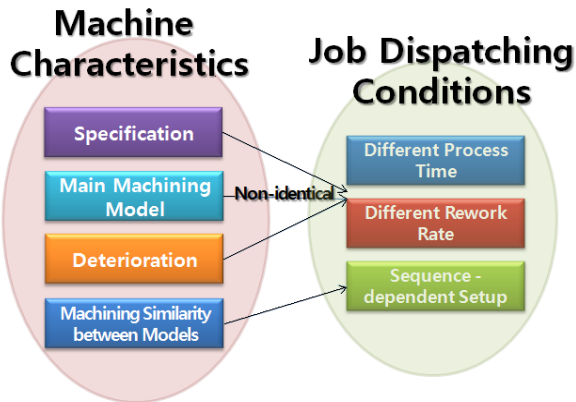


Figure 1. Machine Characteristics -determined Condition of Job Dispatching Environment

<Figure 1>의 공정 환경에서, 작업할당 순서는 투입대비 생산효율을 결정짓는다. 특히, 재작업확률이 높으면 재작업이 자주 발생하고, 재작업에 소요되는 시간의 증가로 공정소요시간이 증가한다. 반면, 재작업확률이 낮고 작업준비 시간과 가공시간이 짧은 기계에서 가공은 재작업과 같은 낭비시간이 줄어들어 공정소요시간이 단축된다. 따라서 <Figure 1>의 공정 특성에서 생산효율 향상은 공정소요시간 단축과 상통하며, 공

정소요시간 단축은 재작업과 가공시간 및 작업준비 시간을 고려한 효과적인 작업할당을 통해 이루어져야 한다. 그러나 재작업확률과 가공시간은 측정단위가 다르므로 두 가지 요소에 대한 직관적 비교가 어렵다. 즉, 재작업확률이 낮으나 가공시간이 긴 설비 또는 재작업확률이 높지만 가공시간이 짧은 설비처럼 두 가지 요소가 상충관계에 놓인다면 작업할당의 기준이 모호해진다. 특히, 순서의존적 작업준비 시간은 기계에 이전 할당된 작업 유형에 영향을 받으므로 할당시점마다 효과적인 투입이 될 수 있도록 확인해야만 한다. 따라서 작업할당이 생산효율과 직결되지만, 할당시점에서 다양한 제품유형과 다수 기계가 존재할 때 공정 환경을 고려한 생산 효율적 작업배분은 쉽지 않다. 그러나 성능의 차이가 존재하는 다수 기계 공정 환경에서 작업배분의 기준에 따라 생산능력은 증대되거나 감소한다. 따라서 생산효율 측면에서 작업할당을 재조명하고, 공정 환경 및 공정소요시간의 상관관계와 상충 관계를 충분히 고려하여 공정에서 불필요한 시간을 감소할 수 있는 효과적인 작업할당에 대한 연구가 필요하다.

기존의 일반적인 작업할당 문제는 납기 준수와 공정완료시간 최소화를 목적으로 규칙기반 할당과 탐색기반 할당을 사용하여 주로 연구되었다. 규칙기반 할당은 특정 조건들을 수치적으로 조합한 규칙을 통해 가중치를 산정하여 가중치 중요도 따라 작업 할당을 수행한다. 규칙기반 할당 연구는 다음과 같다.

Chung *et al.*(2000)은 생산능력이 다른 이중병렬기계 생산체제에서 일정계획 상황의 동적 변화를 감당할 수 있는 일정계획 지원시스템을 연구했다. 할당해법은 최소비용할당법이다. 할당비용은 작업완료시간, 납기, 가공시간, 긴급률 중 하나의 항목으로 결정된다. 할당비용을 다양한 기준에서 평가할 수 있는 장점은 있지만 제시된 항목에 대한 복합적인 평가가 불가능하고 병렬기계의 재작업확률에 대한 고려가 없었다.

최근에는 재작업을 고려하는 작업할당 방법이 제시되고 있다. 하지만 제시된 연구들은 순서의존적인 작업준비시간은 가정하였으나 가공시간 측면의 이중병렬기계 환경은 고려하지 않았다. Kang *et al.*(1998) 이중병렬기계에서 작업의 긴급을 인덱스로 납기지연 작업 수와 납기지연을 최소화하는 일정계획법을 연구하였다. 이어 Kang *et al.*(2007)은 재작업확률이 다른 병렬기계를 이중병렬기계로 정의하고 재작업확률을 고려한 작업의 예상완료시간과 EDDR(Earliest Due Date with Rework rate) 규칙을 이용하여 납기지연과 재작업 빈도를 최소화하는 작업할당방법을 연구했다.

Ko and Baek(2010)도 품질을 고려한 작업할당을 연구하여 MDD(Modified Due Date)에 공정능력지수로부터 산출한 품질인덱스를 포함한 MDDQ(MDD with Quality) 기법을 제시하였다. MDDQ는 품질을 높이고 평균 납기지연 시간을 최소화하는 지수에 의한 작업할당방법이다. Ko(2010)는 MDDQ 규칙을 더욱 체계적으로 심화시켜 MDDQ의 효율을 향상시켰다. Ko and Bek(2010)과 Ko(2010)의 연구는 재작업확률은 동일 작업이라도 기계마다 상이하지만 가공시간은 동일하다고 가정하여 문제

환경을 단순화하였다.

탐색기반 할당은 초기해를 구하고 탐색알고리즘을 통해 최적 또는 최적에 준하는 해를 찾는 방법이다. 작업할당연구에서는 주로 규칙기반 하에 할당대상의 작업들로 초기 작업순서를 찾아내고 탐색기반 할당기법을 통해 가능한 모든 순서조합을 찾아내고 평가하여 최적 목표 값을 갖는 할당순서를 구한다. 따라서 탐색기반 할당 연구는 주로 작업수와 작업발생정보가 할당조건으로 주어져 있는 문제 환경에 기반을 둔다.

Lee and Pinedo(1997)는 순서의존적 작업준비 시간을 고려한 ATCS(Apparent Tardiness Cost With Setup)라는 할당 규칙과 SA(Simulated annealing)를 활용하여 가중된 납기 지연의 합을 최소화하기 위한 세 단계의 휴리스틱 기법을 연구하였다. 최근 Lee et al.(2010)은 최대납기 지연(L_{max})을 최소화하기 위해서 RSA(restricted simulated annealing)라는 제한된 탐색 기법을 순서의존적 작업준비 시간을 고려한 작업환경에서 사용하였다.

Drozowski and Dell'Olmo(2000)는 MWS(Merge by Weight and Size) 알고리즘으로 동종병렬기계에 평균 공정완료시간을 최소화하도록 일체계획을 연구했다. MWS 알고리즘도 작업수와 작업발생정보가 할당조건으로 주어져 있는 환경 하에서 개발되었다.

Kim and Shin(2003)은 Restricted Tabu Search를 사용하여 동종 또는 이종병렬기계 환경에서 최대납기 지연(L_{max})을 최소화하는 작업할당방법을 연구하였다. 이 연구에서는 순서의존적 작업준비 시간이나 이종병렬기계 조건은 있었으나 제작업에 대한 고려는 없었다.

Turkcan et al.(2003)은 이종병렬 CNC(Computerized numerical control)기계 환경에서 Posynomial GP(geometric programming) 모델을 통해 기계의 최적 조건을 찾아내서 기계, Tool, 가공품 할당을 결정하는 방법을 제안하였다. Turkcan et al.이 제안한 방식은 제작업은 고려하지 않았지만 이종병렬기계에서 기계의 현재 상태를 감안한 할당방법이었다.

이상의 연구들에서 제시한 할당방법은 납기준수를 중심으로 연구되었으며 이종병렬기계, 제작업확률 및 순서의존적 작업준비시간을 동시에 고려하지 않았다. 또한 기존 연구에서 제시된 납기만족 위주의 작업할당은 납기까지의 여유시간을

기준으로 우선순위를 결정하였다. 하지만 이종병렬기계는 가장 큰 소요시간을 갖는 작업이 여유시간이 짧아 높은 우선순위를 갖게 되어 생산효율의 저하를 초래하게 된다. 그러나 공정소요시간 단축을 위한 작업할당은 작업의 완료시점을 앞당기고, 작업의 기계점유시간을 전체적으로 단축하여 투입 대비 생산효율을 높일 수 있다. 또한 공정소요시간을 단축하면 기계는 다른 작업을 위한 여유시간 확보가 가능하므로 납기 준수도 유리하게 된다. 따라서 본 연구는 고객의 주문에 의해 실시간 발생하는 작업을 이종병렬기계에 할당할 때 제작업확률과 순서의존적 작업준비를 고려하여 평균 공정소요시간을 최소화하는 작업할당방법을 제안한다.

2. 제작업과 이종병렬기계를 고려한 작업할당

2.1 문제의 정의

순서의존적 작업준비 시간과 제작업이 존재하는 이종병렬기계에서 공정소요시간 최소화를 위한 본 연구의 문제 상황은 <Figure 2>와 같다.

<Figure 2>에서 고객(Customer)의 주문은 시간의 흐름에 따라 실시간 발생된다. 주문은 제품유형(Type)과 납기 정보(Due date)를 포함하는 단위작업(Job)으로 구성되며, 도착 간격은 유니폼 분포를 따른다고 가정한다. 가공 가능한 제품유형은 정해져 있으며, 모든 기계에서 가공이 가능하다. 제품유형은 기계에 따라 서로 다른 작업준비시간, 가공시간, 제작업확률을 갖는다. 특히 동일 제품유형이라도 가공시간, 제작업확률이 기계마다 상이하며, 작업준비 시간은 선행 제품유형에 따라 순서의존적 작업준비 시간을 갖는다. <Figure 2>와 같이 대상공정은 대기공간(Queue), 기계그룹(Machine Group), 검사(Inspection), 초기화(Initialization)로 구성된다. 대기공간은 고객의 주문에 의해 발생한 작업이 기계에 투입되기 전까지 할당을 기다리는 공간이다. 대기공간이 수용할 수 있는 작업의 수는 제한이 없다. 기계그룹은 다수 이종병렬기계로 구성된다. 검사는 기계의 처리가 완료된 작업의 제작업 여부를 판단하는 공정이다. 검사는 기계처리 완료 즉시 제작업 판별을 내린다. 초기화

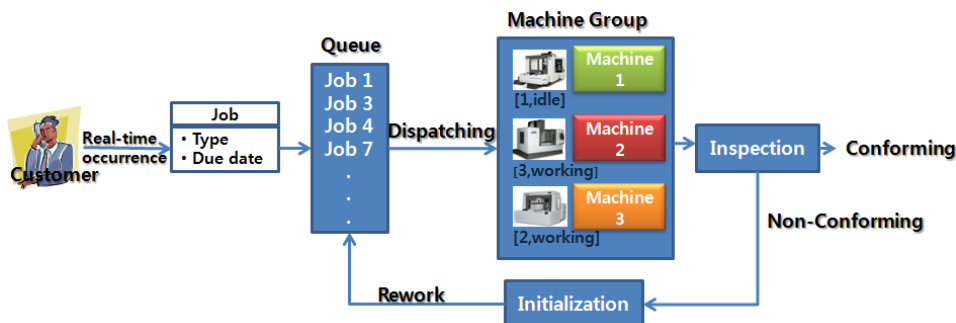


Figure 2. Dispatching Process in Non-identical parallel Machines Environment with Rework and Sequence dependent Setup time

는 검사에서 재작업 판정(Non-Conforming) 받은 작업이 기계 재투입을 위해 준비하는 공정이다. 초기화 시간은 제품유형과 무관하게 동일하다. 초기화된 작업은 대기공간으로 이동한다. 작업은 재작업 판정이 나지 않을 때까지 반복하여 가공된다. 기계의 고장 및 수리에 따른 기계 정지상황은 없고, 공정간 이동거리는 고려하지 않는다.

작업할당은 필요 시점에 미리 정해진 할당규칙에 의해 이루어지며, 하나의 작업이 기계그룹 중 하나의 기계에 할당된다. 작업할당은 할당시점 이후에 발생할 작업에 대한 정보를 참조할 수 없다. 할당 작업은 재작업 판정에 대해 미리 알 수 없다. 재작업 판정 이후 초기화를 진행 중인 작업은 초기화 완료 후 투입가능시점을 알 수 있으나, 동적인 작업발생으로 인해 투입가능시점에서 대기공간의 작업들을 예측할 수 없어 미리 할당계획에 포함 시킬 수 없다. 따라서 작업할당은 해당시점에서 대기공간에 대기하는 작업만을 대상으로 한다.

작업할당은 <Figure 3>과 같은 순서를 갖는다. 기계가 유휴가 되는 시점에 대기공간에 대기 중인 작업이 존재하면 할당기준에 의해 할당이 이루어진다. 기계가 유휴가 되는 시점에 대기공간에 작업이 없다면 할당은 고객주문으로 실시간 발생한 작업이나 초기화된 재작업이 대기공간에 도착할 때까지 유보된다.

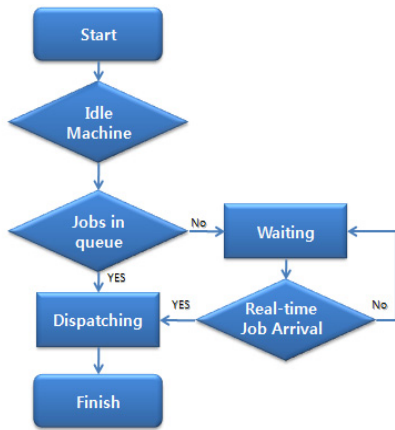


Figure 3. Dispatching Steps in Non-identical parallel Machines Environment with Rework

이상의 공정 환경에 대해, 본 연구는 생산효율을 고려하여 평균 공정소요시간을 최소화하는 작업할당방법을 제안하며, 식 (1)과 같이 목적함수를 표현한다.

$$\lim_{n \rightarrow \infty} Min \frac{1}{n} \sum_{j=1}^n F_j \tag{1}$$

- j : 작업 인덱스
- n : 완료된 작업의 수
- F_j : 작업 j 의 실제 공정소요시간

목적함수에서 공정소요시간 F_j 는 대기시간, 준비시간, 가공시간, 초기화를 비롯한 재작업 시간의 합으로 작업 j 의 실제 소요시간을 의미한다. 본 문제는 작업수의 제한이 없고 시간이 지남에 따라 작업할당이 계속적으로 일어나기 때문에 완료작업 수인 n 은 무한대로 증가한다. 따라서 모든 시점을 대표하는 목적함수 표현이 필요하고, 일정 시점까지 완료된 작업 n 개의 공정소요시간 총합을 n 으로 나눈 평균 공정소요시간을 최소화로 나타낸다. 현재 제조공장들은 원자재 가격 상승과 과열된 경쟁 속에서 원가절감 및 생산량 증대라는 과제에 직면하고 있다. 실제 제조라인에서 작업할당에 의한 평균 공정소요시간의 단축은 공정에 투자를 가하지 않고도 생산능력을 증대할 수 있음을 의미한다. 공정소요시간을 단축하는 작업할당은 동일시간 내 생산량을 증가시키거나, 또는 동일 작업을 처리하는 시간을 단축하여 생산비용 절감을 유도하게 된다.

Pinedo(2002)에 의하면, 본 연구에서 정의한 목적함수와 병렬기계 공정 환경에서 작업할당의 순서결정은 NP-hard이다. 일반적으로 NP-hard의 문제는 휴리스틱 기법인 탐색기반 할당과 규칙기반 할당에 기초한 연구들이 제시되었으며 본 연구도 휴리스틱 기법을 활용하여 작업할당을 제안하고자 한다. 그러나 두 가지 휴리스틱 기법 중 탐색기반 할당은 전체 최적화를 위해 많은 탐색시간이 필요하다. 특히, 대기작업이나 병렬기계의 수가 증가하면 휴리스틱 기법이 탐색해야 하는 조합이 기하급수적으로 증가하여 계산시간도 크게 증가한다. 하지만 규칙기반 할당은 빠른 시간 안에 할당이 가능하고 비교적 간단하여 본 연구와 같이 복잡한 할당 조건에서 실시간 작업할당에 적합하다. 따라서 본 연구는 규칙기반 형태의 작업할당을 제안한다.

2.2 Flow Time Loss Index with Rework Rate

공정소요시간 최소화를 위한 작업할당을 위해 Flow Time Loss Index with Rework Rate(FTLR) 알고리즘을 제안한다. FTLR은 공정 환경을 고려하여 예측한 소요시간을 기준으로 대기작업에 대해서 기계별로 소요시간 단축에 대한 손실률을 상대적인 가중치로 평가한 후 손실을 줄이는 방향으로 작업을 기계에 할당한다. FTLR 알고리즘의 작업할당 방법은 <Figure 4>와 같이 요약할 수 있다. 먼저, FTLR은 대기 중인 작업에 대해서 이중병렬기계에서의 가공시간, 순서의존적 작업준비시간, 재작업확률을 기반으로 작업의 공정소요시간을 기계별로 예측한다. 예측된 공정소요시간으로 각 대기작업별 예상 공정소요시간의 평균과 표준편차를 구하고, 평균과 표준편차는 다시 예상 공정소요시간을 0~1사이로 정규화(Normalizing)하여 FTLR 가중치를 구하는데 기준으로 이용된다. FTLR은 유휴기계에서의 FTLR 가중치를 비교하여 가중치가 가장 큰 작업을 기계에 할당한다. 만약 가중치가 동일하다면, FTLR은 납기순수를 고려하여 납기가 이른 작업을 우선 선택한다.

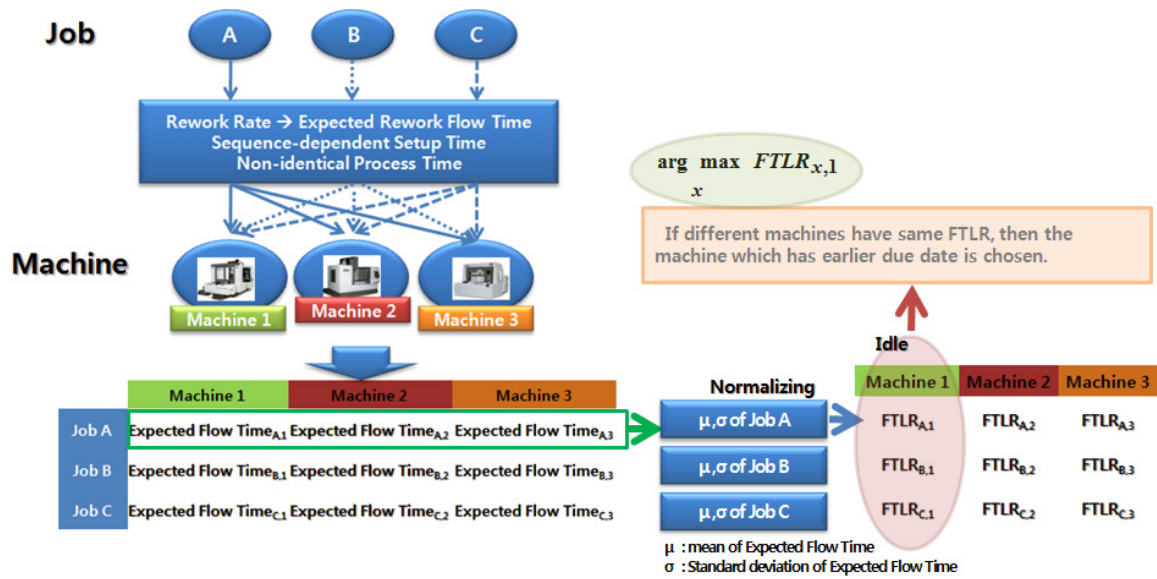


Figure 4. Calculating FTLR and Dispatching Rule

FTLR에서 작업할당 순서는 시간을 기준으로 이루어져야 하므로 재작업확률은 시간으로 나타내는 단위 환산작업이 필요하다. FTLR은 재작업 발생 시 처리시간과 재작업확률을 곱하여 재작업확률을 시간으로 예측한다. 재작업 발생 시 처리시간은 초기화 시간과 현 시점에서의 작업준비 시간과 가공시간으로 하여, 현 시점의 할당 상황을 충분히 고려하도록 한다. 따라서 FTLR 작업할당의 기준은 동적 할당에 근거하여 실제 공정소요시간이 아닌 재작업확률을 환산한 시간에 작업준비시간과 가공시간을 합한 예상 공정소요시간으로 식 (2)와 같이 산출한다.

$$P_{j_i,m} = (s_{j_i k_l} + p_{j_i,m}) + r_{j_i,m} \times (W + s_{j_i k_l} + p_{j_i,m}) \quad (2)$$

$P_{j_i,m}$: 기계 m 에서 제품유형이 i 인 작업 j 의 예상 공정소요 시간

j : 작업 인덱스 ($j = 1, 2, \dots, N$)

i : 제품유형 인덱스 ($i = 1, 2, \dots, J$)

m : 기계 인덱스 ($m = 1, 2, \dots, M$)

k : 기계에서 생산 중이거나 최후 생산한 작업 인덱스

($k = 1, 2, \dots, N-1$)

l : 기계에서 생산 중이거나 최후 생산한 작업의 제품유형 인덱스 ($l = 1, 2, \dots, J$)

k_l : 기계에서 생산 중이거나 최후 생산한 제품유형이 l 인 선행 작업 k

$s_{j_i k_l}$: k_l 에서 j_i 로 작업 변경 시 소요되는 작업준비시간

$p_{j_i,m}$: 기계 m 에서 j_i 의 가공시간

W : 재작업을 위한 작업 초기화시간을 나타내는 단일값

$r_{j_i,m}$: 기계 m 에서의 작업 j_i 의 재작업확률

작업 j_i 를 기계 m 에 할당하여 가공하여 재작업이 발생하면, 초기화 작업시간과 재작업준비 시간 및 재가공시간이 추가적으로 발생한다. 재작업확률은 초기화를 거쳐 기계에 재투입된 시점에서도 동일하므로 재작업은 작업이 정상으로 완료될 때까지 무한반복 된다. 그러나 재투입 시점에서 어느 기계에 할당될 것인지는 할당시점에서 산출된 예상 공정소요시간에 의해 결정되므로, 현 시점에서의 재작업 여부만 판단하여 재작업확률에 의한 예상 공정소요시간을 산출한다.

예상 공정소요시간은 재작업확률이 증가하면 작업준비시간과 가공시간에 비례하여 증가하고 재작업확률이 낮아지면 작업준비 시간과 가공시간에 비례하여 감소한다. 따라서 작업은 공정소요시간 최소화를 위해서 작업준비시간 및 가공시간이 짧고 재작업확률이 낮은 기계에 우선적으로 할당되어야 한다. FTLR은 대기 중인 작업의 기계별 예상 공정소요시간을 비교하여 공정소요시간에 대해 작업이 갖는 기계 선호도를 FTLR 가중치로 표현한다. FTLR 가중치는 기계선호도를 나타내기 위해 작업이 갖는 기계 간 예상 공정소요시간의 차이를 평가한다. 예를 들어, 유희기계에 대해 작업들이 동일 우선순위를 갖더라도 타 기계 대비 소요시간의 단축이 더 유리한 작업이 FTLR 가중치가 더 커서 유희기계에 할당된다. 따라서 FTLR 가중치는 기계에 대한 예상 공정소요시간이 짧은 작업일수록 높다. 또한 FTLR 가중치는 하나의 작업에 대한 기계별 비교도 가능해야 하지만 서로 다른 작업 간 상대적인 비교도 가능해야 하므로 명목 값이 아닌 상대적 크기로 표현하며, 작업비교 시 작업이 갖는 공정소요시간의 절대크기가 고려되지 않도록 0과 1사이로 정규화한 값을 사용한다. 따라서 FTLR 가중치는 식 (3)과 같이 나타낸다.

$$FTLR_{j_i,m} = \frac{\exp(A)}{1 + \exp(A)} \quad (3)$$

$FTLR_{j_i,m}$: 기계 m 에서 작업 j_i 의 $FTLR$ 가중치

A : 예상 공정소요시간을 정규화한 값

$$A = \frac{\overline{P}_{j_i} - P_{j_i,m}}{S_{j_i}} \quad (4)$$

\overline{P}_{j_i} : j_i 의 기계별 예상 공정소요시간의 평균

S_{j_i} : j_i 의 기계별 예상 공정소요시간의 표준편차

$$\overline{P}_{j_i} = \frac{1}{M} \sum_{m=1}^M P_{j_i,m} \quad (5)$$

$$S_{j_i} = \sqrt{\frac{1}{M} \sum_{m=1}^M (P_{j_i,m} - \overline{P}_{j_i})^2} \quad (6)$$

식 (3)에서 보듯 FTLR은 Sigmoid 함수 형태로 FTLR 가중치를 표현한다. 식 (3)의 A는 예상 공정소요시간을 정규화한 값으로 식 (4)와 같이 구한다. 식 (4)는 기계들의 평균 예상 공정소요시간인 식 (5)의 \overline{P}_{j_i} 와 각 기계의 예상 공정소요시간인 $P_{j_i,m}$ 의 차이를 식 (6)의 S_{j_i} 로 나누어 0~1사이의 값으로 정규화한다. 예를 들어, 작업에 대한 특정 기계의 예상 공정소요시간이 이 작업이 투입가능한 모든 기계에서의 예상 공정소요시간의 평균과 같다면, FTLR은 0.5가 된다. 또한 평균 예상 공정소요시간보다 더 작은 공정소요시간이 예상된다면, FTLR 값은 0.5 이상이 되고, 평균을 기준으로 얼마나 더 단축할 수 있는가에 따라 FTLR 값은 더 커지게 된다. 그러나 그 값은 1을 넘지 못한다. 반대로, 평균 예상 공정소요시간보다 더 큰 공정소요시간이 예상된다면 FTLR은 0.5이하가 되고, 더 큰 정도에 따라 FTLR은 낮아져 0에 가까운 값이 된다.

FTLR 가중치가 유도되는 과정을 예를 들어 설명하면 다음과 같다. <Table 1>은 3개의 작업이 3개의 기계에 할당되는 문제 상황으로 각 기계별 작업의 예상 공정완료시간과 평균 예상 공정완료시간, 각 작업이 갖는 예상 공정완료시간의 표준편차를 나타낸다.

Table 1. Expected Flow Time in Case 1

	Machine 1	Machine 2	Machine 3	Average	Standard Deviation
Job 1	100	120	200	140	43.2
Job 2	30	16	5	17	10.2
Job 3	40	20	60	40	16.3

<Table 1>에서 기계 1에 3개의 작업 중 하나의 작업을 할당하는 경우, 공정소요시간을 단축을 기준으로 절대적인 시간을 본다면 예상 공정소요시간이 가장 작은 작업 2를 투입하게 된다. 그러나 작업 2만 놓고 평가하면 3대의 기계 중에서 기계 1

에 할당 될 때 가장 긴 공정소요시간이 예상된다. 따라서 이 경우에는 절대적인 시간은 가장 길지만 상대적인 시간을 기준으로 기계 1에서 가공하였을 때 가장 작은 공정소요시간이 예상되는 작업 1을 할당하는 것이 적합하다. 즉, 작업 1은 기계 1에 할당되지 않는다면, 상대적으로 긴 공정소요시간을 갖는 기계 2, 기계 3에 투입될 확률이 높아지면서 공정소요시간 단축에 있어서 손실이 발생하게 되는 것이다. 따라서 공정소요시간을 최소화하기 위해서는 기계 1에서는 다른 작업보다도 작업 1이 우선적으로 할당되어야 하고 기계 3에서는 다른 작업보다 우선권이 낮아야한다. <Table 1>의 예제에서 FTLR 가중치를 계산한 결과는 <Table 2>와 같다. <Table 2>에서 보듯이 FTLR 가중치는 기계 1에서 작업 1이 0.716으로 가장 크다.

Table 2. FTLR in Case 1

	Machine 1	Machine 2	Machine 3
Job 1	0.716*	0.613	0.199
Job 2	0.219	0.524	0.763
Job 3	0.5	0.773	0.227

정리하면, FTLR은 평균 예상 공정소요시간과 표준편차를 이용한 Sigmoid 함수를 사용하여 기계별 예상 공정소요시간을 기준으로 작업을 선택하되 현재 기계 m 에서 $\overline{P}_{j_i} > P_{j_i,m}$ 이면 다른 작업보다도 우선적으로, $\overline{P}_{j_i} \leq P_{j_i,m}$ 이면 다른 작업을 우선적으로 할당한다. 따라서 FTLR은 절대적 시간 크기와는 상관없이 상대적인 시간을 고려하여 공정소요시간을 단축하는 방향으로 작업을 할당하도록 한다.

<Figure 5>는 3대의 기계에서 3종의 제품유형에 대한 작업이 존재하는 경우, 기계가 유희가 되는 시점에 작업할당의 예제 상황을 보여준다.

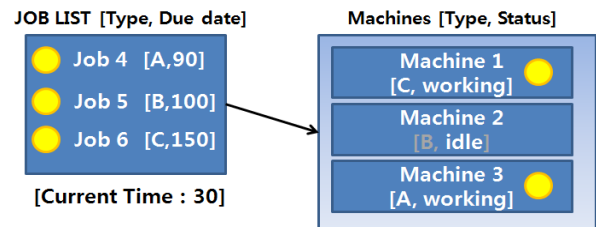


Figure 5. Machine and Job List Status of FTLR Dispatching Example

현재 시각이 30이 되면서 기계 2가 제품유형이 B인 작업을 완료하고 유희 상태가 되었다. 대기공간에 대기작업이 존재하므로, 기계 2는 대기 중인 작업을 대상으로 FTLR 가중치를 계산한다. 대기공간에는 작업 4, 5, 6이 대기하고 있으며, 작업 4는 제품유형이 A이고 납기는 90이다. 작업 5는 제품유형이 B이고, 100이 납기시간이다. 마찬가지로 작업 6은 제품유형이 C, 납기가 150이다. 각 제품유형의 기계별 재작업확률과 가공

시간, 순서의존적인 작업 준비시간이 <Table 3>과 같고, 초기화 작업시간은 50으로 한다.

Table 3. Conditions of FTLR Dispatching Example

		Machine 1	Machine 2	Machine 3	Average
Rework Rate	Type A	0.15	0.05	0.10	0.10
	Type B	0.01	0.10	0.05	0.05
	Type C	0.20	0.10	0.30	0.20
Process Time	Type A	100	150	200	150
	Type B	60	80	100	80
	Type C	70	20	30	40
		Type A	Type B	Type C	Average
Setup time	Type A	0	50	70	40.0
	Type B	50	0	60	36.7
	Type C	70	60	0	43.3

<Table 3>에서 할당기준에 따라 기계 2에 투입되는 작업은 달라진다. 예를 들어, 재작업확률이 기준이면 작업 4를, 가공시간 기준이면 작업 6을, 또는 작업준비시간 기준이면 작업 5를 기계 2에 투입하는 것이 적합하다. <Table 3>과 같은 조건에서는 세 가지 기준 모두에서 우선순위가 높은 작업이 존재하지 않으므로 세 가지 기준을 모두 고려한다면, 작업할당이 쉽지 않다. 반면, FTLR은 <Figure 5>의 상황에서 <Table 3>의 정보를 모두 고려하여 우선작업을 구한다. 먼저 대기작업의 기계에 대한 예상 공정완료시간을 계산한다. 예를 들어, 제품유형이 A인 작업 4의 기계 2에서 예상 공정소요시간은 식 (2)를 통해 $P_{4,2} = (50 + 150) + 0.05 \times (50 + 50 + 150) = 212.5$ 이 된다. <Table 4>는 작업 4, 5, 6에 대한 기계별 예상 공정소요시간을 정리한 표이다.

Table 4. Expected Flow Time of FTLR Dispatching Example

	Machine 1	Machine 2	Machine 3	Average	Standard Deviation
Job 4	203	212.5	225	213.5	9.0
Job 5	121.7	93	160	124.9	27.4
Job 6	94	93	145	110.7	24.3

<Table 4>를 기준으로 기계 2에서 각 작업의 FTLR 가중치를 계산한다. 예를 들어, 작업 4는 기계 2에서 예상 공정소요시간은 212.5이고 작업 4의 평균 예상 공정소요시간이 213.5이므로 기계 2에서 FTLR 가중치는

$$FTLR_{4,2} = \left(\frac{\exp((213.5 - 212.5)/9.0)}{1 + \exp((213.5 - 212.5)/9.0)} \right) = 0.5277 \text{ 이다.}$$

작업 4는 기계 2에서의 예상 공정소요시간이 평균 예상 공정소요시간과 비슷하기 때문에 FTLR은 0.5에 가까운 값을 갖는다. 같은 방법으로 <Table 5>는 할당시점에서 기계별 FTLR 가

중치를 산출한 것이다. 본 예제상황에서 현재 기계 2가 유향이고 기계 2에서 작업 5의 FTLR 가중치가 0.7617로 가장 크므로 FTLR 기준으로는 작업 5가 기계 2에 할당된다.

Table 5. FTLR index of FTLR Dispatching Example

	FTLR		
	Machine 1	Machine 2	Machine 3
Job 4	0.7623	0.5277	0.2182
Job 5	0.5291	0.7617*	0.2177
Job 6	0.6652	0.6743	0.1956

<Table 4>에서 재작업과 순서의존적 작업준비시간을 고려한 예상 공정소요시간은 기계 2에서 작업 5와 작업 6이 93으로 가장 짧았다. 하지만 작업 6이 할당될 경우 작업 5는 기계 1이나 기계 3에 할당될 확률이 높아지면서 공정소요시간 증가가 예상된다. 예상 공정소요시간의 증가는 기계 1에서 28.7, 기계 3에서 67이다. 반면 작업 5를 기계 2에 할당하여, 작업 6이 기계 1에 할당되면 예상 공정소요시간은 1, 기계 3에 할당되면 52만큼 증가된다. 따라서 공정소요시간의 단축기준에서는 동일 우선순위의 동일시간이더라도 작업 5를 선택하는 것이 적합하다. <Table 5>와 같이, FTLR 가중치는 공정소요시간 기준으로 작업에 대한 기계의 선호도를 정규화한 값이므로 타 기계 대비 상대적인 우위를 나타내므로 <Figure 5>의 상황에서 <Table 3>의 조건을 기준으로 FTLR은 작업 5의 할당이 가능하다.

2.3 FTLR 작업할당 절차

본 연구의 작업환경에서는 실시간으로 주문이 발생하므로 작업정보는 발생시점에만 알 수 있다. 따라서 미래에 투입될 작업정보를 현재 시점에서 미리 감안하여 작업을 할당할 수 없으므로 실시간 작업할당을 수행해야 한다. 본 연구는 실시간 작업할당을 위해 규칙기반의 휴리스틱 기법으로 FTLR을 사용한다. 따라서 작업할당은 공정이 진행되고 있을 때 필요시점마다 발생한다.

공정이 진행되고 있을 때 작업이 할당되는 시점은 두 가지 경우다. 첫째, 기계가 유향이 되는 시점에서 할당대기 중인 작업이 있다면 할당절차는 다음과 같이 진행된다.

Step 1 : (기계작업 완료)

기계가 진행 중인 작업을 완료하고 유향상태가 됨

Step 2 : (FTLR 계산)

대기 중인 작업을 대상으로 해당기계에서의 FTLR 가중치 계산

Step 3 : (작업할당)

해당기계에 대한 FTLR 가중치가 가장 큰 작업을 해당 기계에 투입, FTLR 가중치가 동일하면 납기가 빠른 작업 투입

둘째, 기계가 유휴가 되었는데 대기공간에 할당대기 중인 작업이 없다면 할당은 대기공간에 새로운 작업이나 재작업이 도착하는 시점까지 연기되며, 할당절차는 다음과 같이 진행된다.

Step 1 : (기계작업 완료 및 대기)

기계가 진행 중인 작업을 완료하고 유휴상태가 되었으나 할당이 가능한 작업이 없으므로 대기

Step 2 : (대기공간 내 작업 도착)

고객의 주문에 의해 새로운 작업이 발생하거나 초기화를 마친 재작업이 대기공간에 투입

Step 3 : (예상 공정소요시간 계산)

도착한 작업을 대상으로 유휴기계별 해당 작업의 FTLR 가중치를 계산

Step 4 : (기계선택)

FTLR 가중치가 가장 큰 유휴기계에 해당 작업을 할당, FTLR 가중치가 동일하면 유휴기간이 긴 기계 선택

작업할당의 실행시점을 두 시점으로 나누는 것은 기계의 유휴상태를 최소로 허락한다. 따라서 기계가동률이 올라가 작업의 완료시점을 앞당길 수 있다. 또한, FTLR은 가중치가 동일하다면 납기가 빠른 작업을 할당하도록 하여 납기 준수에 좋은 영향을 준다.

3. 실험 및 결과 분석

3.1 비교 알고리즘

본 연구는 동일 제품유형에 대해 재작업확률과 가공시간이 기계별로 상이한 이종병렬기계 환경에서 작업의 평균 공정소요시간을 최소화하는 작업할당인 FTLR을 제안하였다. 목적함수에 대한 FTLR의 공정한 성능평가를 위한 실험은 다양한 비교대안들을 선정하여 진행한다. FTLR과 동일하게 이종병렬기계와 순서의존적 작업준비시간, 재작업확률을 모두 고려하는 기존연구 및 작업할당 방법은 없었다. 따라서 기본적인 작업할당 방식인 EDD(Earliest Due Date)와 MS(Minimum Slack), 그리고 최근 연구된 EDDR과 MDDQ를 비교대안 알고리즘으로 실험을 수행한다. EDDR과 MDDQ는 순서의존적 작업준비시간과 재작업확률을 고려한 작업할당방법이다. 하지만 EDDR, MDDQ는 이종병렬기계는 고려하지 않는다. 그러나 두 알고리즘은 기본적으로 예상 공정완료시간의 기계별 차이에 따라 작업을 할당하므로 공정소요시간에 대한 고려가 존재한다는 것은 FTLR과 동일하다. 따라서 본 연구의 모든 공정조건을 고려한 기존의 알고리즘이 없으므로 FTLR을 FTLR과 유사하게 기계별 공정소요시간의 차이를 기준으로 하는 EDDR과 MDDQ와 비교하여 FTLR의 성능을 평가하고자 한다.

MS는 이종병렬기계 환경에 맞도록 수정된 MMS(Modified Minimum Slack)을 사용한다. MMS 계산식은 식 (7)과 같다. MMS

는 작업가공완료 시간에서 납기까지 여유시간을 산출하여 여유가 가장 없는 작업을 기계에 할당한다.

$$MMS_{j_i,m} = \max(D_{j_i} - s_{j_i k_i} - p_{j_i,m} - t, 0) \tag{7}$$

$MMS_{j_i,m}$: 기계 m 에서의 j_i 의 Minimum Slack

D_{j_i} : j_i 의 납기

t : 할당시점

z 종병렬기계 환경에서는 동일 작업이라도 기계마다 MMS가 다르다. 따라서 작업이 기계를 선택하는 할당시점에서 MMS 알고리즘은 유휴기계 중 MMS가 가장 낮은 기계를 선택한다. MMS가 동일한 경우 FIFO를 기준으로 먼저 발생한 작업을 기계에 할당한다.

EDD는 납기가 가장 이른 순서로 작업을 할당하는 알고리즘이다. 납기는 작업의 특성으로 변하지 않는 값이다. 따라서 납기는 기계가 작업을 선택하는 기준은 되지만 작업이 기계를 선택하는 할당시점에는 기계선택의 기준이 될 수 없다. 이 경우 본 연구의 실험에서는 납기 대신 유휴기간이 가장 길었던 기계에 작업을 할당한다.

EDDR은 Kang *et al.*(2007)이 제시한 작업할당 알고리즘이다. EDDR은 재작업확률을 고려하여 작업을 선호군과 비선호군으로 나누고 각 군 내 EDD로 정렬한다. 기계의 유휴시점에 선호군과 비선호군작업에 대해서 순차적으로 예상 완료시간(ECT : Expected Completion Time)을 계산하여 해당기계에서 가장 빠른 예상 완료시간을 갖는 작업을 기계에 할당한다. 이종병렬기계에서 EDDR의 예상 완료시간은 식 (8)과 같이 산출한다. 식 (8)에서 재가공시간을 예측하는 $r_{j_i,m} \times NR \times (W + \overline{s_{j_i}} + \overline{p_{j_i}})$ 는 문제 상황이 이종병렬기계이므로 작업 j_i 의 평균 가공시간과 평균 작업준비시간을 사용한다.

$$ECT_{j_i,m} = t + s_{j_i k_i} + p_{j_i,m} + r_{j_i,m} \times NR \times (W + \overline{s_{j_i}} + \overline{p_{j_i}}) \tag{8}$$

$ECT_{j_i,m}$: 기계 m 에서의 j_i 의 예상완료시간

NR : 재가공시간 결정모수

$\overline{s_{j_i}}$: j_i 의 제품유형별 작업준비시간 평균

$\overline{p_{j_i}}$: j_i 의 기계별 가공시간의 평균

Kang *et al.*(2007)은 NR 에 따른 EDDR의 성능 분석을 통해 재가공 결정모수 NR 은 1일 때 EDDR의 성능이 좋음을 검증하였다. 본 연구의 실험에서도 NR 을 1로 설정한다. 할당시점에서 작업이 유휴기계 중 하나의 기계에 할당되는 경우에는 예상 완료시간이 가장 짧은 기계를 선택한다.

Ko and Baek(2010)이 제시한 MDDQ는 동적 작업할당알고리즘이다. MDDQ 인덱스는 재작업을 고려한 납기가중치 지표와 품질수준 지표에 조절계수를 곱한 값이다. 특히, 품질수준 지

표로 C_{pk} 를 사용하여 품질차이로 작업분별을 하였으나 본 연구에서는 품질차이를 고려하지 않는다. 따라서 MDDQ의 인덱스 중 재작업을 고려한 납기가중치 지표만을 비교평가에 사용한다. MDDQ의 납기가중치 지표는 납기까지의 여유시간과 예상 가공소요시간에 따른 예상 완료시간을 비교하여 상황에 따라 SPT(Shortest Processing Time)와 MS를 선택적으로 적용하도록 식 (9)와 같은 납기지연시간을 이용한다.

$$T_{j_i,m} = MAX(C_{j_i,m} - t, D_{j_i} - t) \quad (9)$$

$T_{j_i,m}$: 기계 m 에서 j_i 이 갖는 납기지연시간

$C_{j_i,m}$: 기계 m 에서 j_i 의 예상 완료시간

예상 완료시간은 재작업확률을 이용하여 재작업 발생 시 동일기계에서 가공이 계속될 경우 완료시간의 기댓값이다. 식 (9)의 납기지연시간을 기준으로 MDDQ의 납기가중치 지표는 식 (10)과 같다. MDDQ 알고리즘은 MDDQ 가중치가 큰 작업을 기계에 할당한다.

$$MDDQ_{j_i,m} = \exp\left(-\frac{T_{j_i,m}}{s_{j_i} + p_{j_i}}\right) \quad (10)$$

$MDDQ_{j_i,m}$: 기계 m 에서의 j_i 의 MDDQ 가중치 지표

이상의 비교대안 알고리즘을 통해 동일한 실험 조건에서 FTLR을 비교 평가한다. 비교대안 알고리즘은 FTLR의 모든 조건을 고려하고 있지 않지만 MMS와 EDD는 기본 알고리즘으로 FTLR의 기본적인 성능을 평가하고 EDDR과 MDDQ는 품질을 고려한 예상 공정소요시간을 기준으로 하는 알고리즘으로 FTLR의 품질 및 예상 공정소요시간 반영 부분에 대한 평가가 가능하다. 특히, FTLR은 이중병렬기계도 고려하고 있으므로 FTLR이 적합한 할당방법이라면, FTLR은 본 연구의 공정 환경에서 비교대안 알고리즘보다 우수한 성능을 보여야 한다.

3.2 실험설계

실험은 동일 실험환경에서 MMS, EDD, EDDR, MDDQ, FTLR 다섯 가지 알고리즘을 수행한다. 실험환경은 이중병렬기계와 재작업확률, 순서의존적 작업준비시간이 반영되도록 <Table 6>과 같이 설계한다. 성능평가를 위해서 50000시간 동안의 완료작업에 대한 평균 공정소요시간과 평균 납기지연시간을 구한다. 50000은 본 실험설계에서 평균적으로 700개 이상의 작업을 가공할 수 있는 충분히 긴 시간이다.

평균 공정소요시간은 공정소요시간 최소화 목적에 대한 성능을 평가하는 항목이다. 알고리즘은 평균 공정소요시간이 작을수록 공정소요시간 최소화의 목적에 있어 좋은 성능을 가진

다. 또한 본 연구는 공정소요시간 최소화를 통해 납기만족 달성도 고려하므로 납기만족에 대해서도 평가한다. 평균 납기지연시간은 납기만족에 대한 평가척도이다. 평균 납기지연시간이 작을수록 알고리즘은 타 알고리즘에 대비 납기만족도가 높다고 할 수 있다. 하지만 두 가지 성능척도가 좋아도 알고리즘이 작업을 공정하게 할당하지 못하여 최악 조건의 작업이 대기 공간에서 빠져나가지 못해 체류하게 된다면 좋은 알고리즘이라고 할 수 없다. 특히, 실시간 작업할당의 경우 지속적으로 공정이 운용되고 있으므로, 평가시점에서 완료된 작업 외에도 남아있는 작업에 대해서도 평가한다. 잔여작업평가를 위해 실험 완료 후 대기공간의 작업 중 최대 납기지연시간을 측정한다.

<Table 6>과 같이 본 실험에서 사용 중인 제품유형은 10종, 이중병렬기계는 5대이다. 재작업을 위한 초기화시간과 순서의존적 작업준비시간은 가공시간과 기계별 가공시간의 차이를 고려하여 설정한다. 작업의 초기화 시간은 고정된 값 100이고, 작업준비시간은 순서의존적인 작업준비시간에 따라 50~150 사이의 유니폼분포로 랜덤하게 발생한다. 다만, 작업유형이

Table 6. Experiment Data

Experiment Factor	Data	Total
Work Time	50000	1
Job Type	10 where, Random Integer from 1 to 10	1
Number of Machine (M)	5	1
Initialization Time	100	1
Sequence dependent Setup Time	U[50, 150]	1
Basic Process Time	U[200~300]	1
Process Time Difference	Low(U[1~10]) Medium(U[1~30]) High(U[1~50]) Higher(U[1~70])	4
Rework Rate	Low(0.1×U[0.5~1.5]) Medium(0.2×U[0.5~1.5]) High(0.3×U[0.5~1.5])	3
Job Arrival Time (R)	$V \times U[0.8, 1.2]$ where, $V = (1 + \bar{r}) \times (\bar{s} + \bar{p}) / M$ \bar{r} : average rework rate \bar{s} : grand average setup \bar{p} : grand average process time	1
Job Due Date	$R + a \times M \times V$ where, a is random integer from 1 to 4	1
Number of Simulation	AutoMod	100
Number of Experiment		1200

변하지 않을 경우에는 작업변경이 존재하지 않으므로 작업준비시간은 0으로 한다. 재작업확률과 가공시간의 기계별 차이가 연구문제의 주요한 상황이므로 재작업확률과 가공시간의 차이에 변화를 주어 두 요소의 크기 변화가 알고리즘 성능에 미치는 영향을 평가한다. 재작업확률은 일반적인 공정상황을 반영하여 평균을 0.1, 0.2, 0.3으로 하고, 가공시간의 차이는 4가지 경우로 구분하여 작은 차이와 큰 차이를 모두 반영한다. 특히, 가공시간의 경우 작업유형마다 평균 가공시간을 다르게 하여 가공시간의 차이를 더 미세하게 반영한다. 재작업확률과 가공시간 차이 조합의 경우의 수를 구하면 실험 Case는 12가지가 된다. 실험 Case 별 가공시간의 차이를 고려하여 기본 가공시간의 평균은 250으로 한다. 또한 랜덤변수들이 실험에 끼치는 영향을 배제하기 위해 각 Case의 실험결과는 100회 실험 후 평균을 사용하여 표현한다.

FTLR과 비교 알고리즘의 성능을 평가하기 위해서는 각 알고리즘의 할당특성이 잘 반영되도록 대기공간에 평균적으로 적정 수의 작업이 존재해야 한다. 따라서 작업발생간격은 평균 작업준비시간과 평균 가공시간에 재작업확률을 반영하여 $V = (1 + \bar{r}) \times (\bar{s} + \bar{p}) / M$ 과 같이 설정한다. V 는 발생간격 조정실험을 통하여 구한 값으로 12가지 실험 Case에 대해 대기공간에 기다리는 작업이 최소한 5~10이상이 되도록 하는 발생간격이다. 여기서 \bar{r} 는 실험 Case별 평균 재작업확률로 0.1, 0.2, 0.3이고, \bar{s} 는 제품유형의 총 평균 작업준비시간인 100으로, \bar{p} 는 제품유형의 총 평균 가공시간인 250으로 설정하고, M 은 병렬 기계수 5이다. 따라서 실험 Case별로 재작업확률이 변동되면, 작업 발생간격도 조정된다. 납기는 작업발생 시점에서 작업발생간격 V 를 기계 1대당 발생간격으로 환산한 시간에 1~4배로 가정한다. 1~4배는 1에서 4까지의 정수를 무작위 발생시켜서 결정한다.

3.3 실험결과

FTLR의 성능을 평가하기 위해서 기계간 가공시간 차이와 재작업확률을 증가시키면서 MMS, EDD, EDDR, MDDQ, FTLR

알고리즘의 평균 공정소요시간, 평균 납기지연시간, 대기공간 내 최대 납기지연을 측정하였다. 결과 중 상대적인 지표는 Lee et al.(2003)의 Relative Deviation Index(RDI) 산출식을 참고하였다. 다만, 0과 1값을 배제하기 위해서 본 연구에서는 RDI를 $(S_{ab} - 0.9 \times S_B) / (1.1 \times S_W - 0.9 \times S_B)$ 로 산출하였다. S_{ab} 는 a의 실험조건에서 algorithm b의 결과 값, S_B 는 최소 결과 값, S_W 는 최대 결과 값이다. 성능이 좋은 알고리즘은 평균 공정소요시간, 평균 납기지연시간, 대기공간 내 최대 납기지연이 짧으므로 RDI 값이 작은 알고리즘이 성능이 좋다고 할 수 있다.

실험결과는 실험조건별로 재작업확률을 Low, Medium, High로 구분하고, 기계간 가공시간 차이를 Low, Medium, High, Higher로 구분하였다. 납기를 만족하면서 공정소요시간을 최소화하는 목적을 달성하려면 알고리즘은 평균 공정소요시간, 평균 납기지연이 작아야 한다. 또한 동적 작업할당에서 공평한 작업할당이 이루어지려면 알고리즘은 대기공간 내 최대 납기지연이 짧아야 한다.

<Figure 6>은 알고리즘별 평균 공정소요시간 결과를 나타낸다. 평균 공정소요시간은 FTLR이 가장 짧고, MDDQ, EDDR, MMS, EDD 순서로 짧게 나타났다. 비교 알고리즘은 모두 재작업확률이 커질수록 평균 공정소요시간이 증가하는데 재작업확률이 증가하면서 재작업의 빈도가 많아지기 때문이다. 반면, MMS와 EDD의 평균 공정소요시간은 다른 세 알고리즘과 확실하게 구분되는 큰 값을 가진다. 특히, 평균 공정소요시간의 차이는 재작업확률이 커질수록 가공시간의 차이가 커질수록 더 크게 나타났다.

<Table 7>~<Table 9>는 재작업확률이 각각 Low, Medium, High일 때, 평균 공정소요시간을 알고리즘별로 상대 평가한 값(RDI)과 평균 공정소요시간의 산포를 Box Plot으로 정리한 것이다. 재작업확률이 Low, Medium, High일 때 평균 공정소요시간에 대한 알고리즘의 성능은 대체적으로 FTLR, MDDQ, EDDR, MMS, EDD 순으로 좋다. 재작업확률이 Low인 <Table 7>(a)에서 FTLR은 가공시간 차이에 상관없이 4가지 Case에서 모두 가장 짧은 평균 공정소요시간을 보인다. 특히, FTLR은 두 번째로 성능이 좋은 MDDQ 보다 3~10배로 성능이 좋다. <Table 7>(b)를

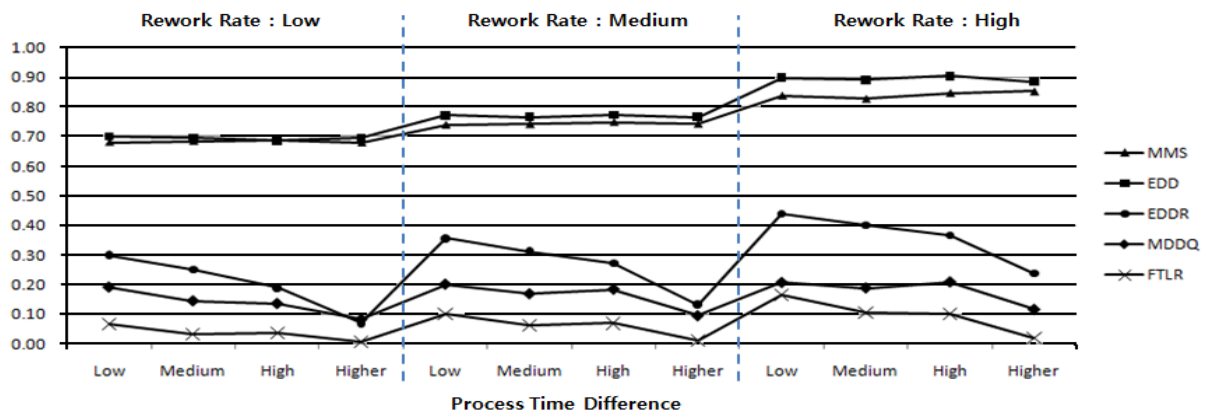


Figure 6. RDI of Average Flow Time in 12 experiment cases

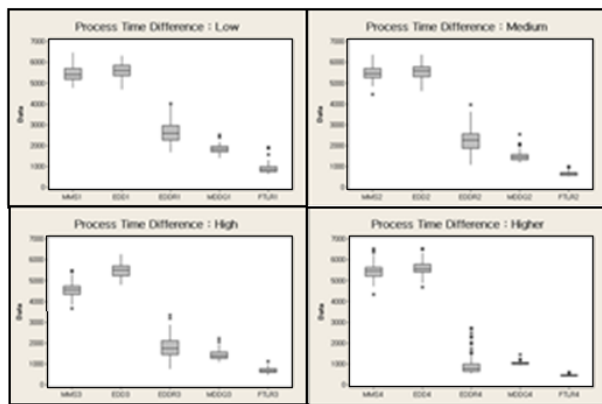
Table 7. Experiment Result-Average Flow Time
(Rework Rate : Low)

(a) RDI of Average Flow Time

Index	Process Time Difference	MMS	EDD	EDDR	MDDQ	FTLR
RDI	Low	0.88	0.90	0.39	0.25	0.09*
	Medium	0.88	0.90	0.32	0.19	0.04*
	High	0.88	0.88	0.24	0.18	0.05*
	Higher	0.87	0.90	0.09	0.11	0.01*

RDI : Relative Deviation Index.

(b) Box Plot of Average Flow Time



보면, 가공시간의 차이가 커질수록 FTLR, MDDQ, EDDR의 산포는 대체적으로 작아지며, FTLR이 산포가 가장 작다. 또한 FTLR이 다른 알고리즘과 유의한 차이를 보이면서, 짧은 평균 공정소요시간을 가짐을 알 수 있다. 재작업확률이 Medium인 <Table 8> 역시, <Table 7>과 비슷한 양상을 보인다. 재작업확률이 Medium일 때, <Table 8>(a)와 같이 FTLR은 가공시간의 차이에 관계없이 가장 짧은 평균 공정소요시간을 가지며, 평균 공정소요시간의 차이는 MDDQ의 2~10배이다. <Table 8>(b)에서 확인할 수 있듯이, FTLR은 타 알고리즘과 유의한 차이를 보이며 성능이 좋음을 확인할 수 있다. 특히, 가공시간 차이가 Higher일 때, FTLR이 가장 작은 산포로 안정적인 성능을 보인다. 재작업확률이 High로 가장 높은 <Table 9>(a)와 <Table 9>(b)에서도, FTLR이 타 알고리즘 대비 가장 짧은 평균 공정소요시간을 가진다. <Table 9>(b)에서 가공시간 차이가 Low인 경우, 재작업확률에 의한 예상 공정소요시간의 차이가 할당의 기준에 큰 영향을 미친 반면, 가공시간의 차이가 할당에 미치는 영향이 작으므로, 재작업확률을 고려한 MDDQ가 FTLR과 비슷한 양상을 나타낸다. EDDR도 재작업확률을 고려하지만 예상 공정소요시간에 따른 예상 완료시간의 비교가 이중병렬기계 내 제한적으로 이루어지기 때문에 MDDQ 대비 상대적으로 성능이 좋지 못하다.

MMS는 납기와의 여유시간이 가장 작은 작업을 우선 할당하도록 한다. MMS 작업할당 특성 때문에 이중병렬기계에서

MMS는 공정소요시간이 큰 기계에 작업을 할당한다. 따라서 MMS는 이중병렬기계에서 공정소요시간 최소화에는 적합하지 않았다. EDD는 작업할당의 기준이 납기순서이므로 기계마다 달라지는 가공시간을 고려하지 못했다. 납기순서로만 기계에 작업을 할당하게 되면 작업은 가공시간이 오래 걸리는 기계에 배치되는 경우도 발생하여 오히려 공정소요시간의 증가를 초래한다.

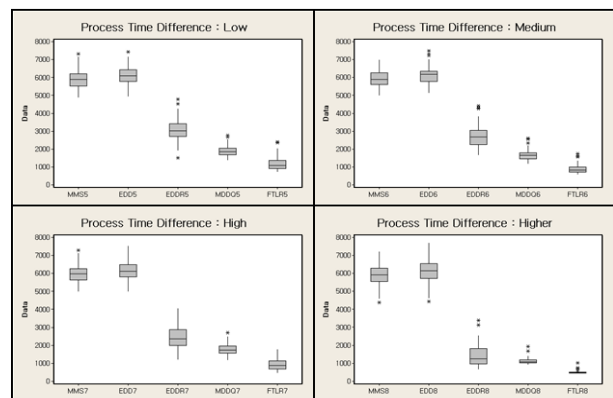
Table 8. Experiment Result-Average Flow Time
(Rework Rate : Medium)

(a) RDI of Average Flow Time

Index	Process Time Difference	MMS	EDD	EDDR	MDDQ	FTLR
RDI	Low	0.86	0.90	0.41	0.23	0.11*
	Medium	0.87	0.89	0.36	0.19	0.07*
	High	0.87	0.90	0.31	0.21	0.08*
	Higher	0.87	0.89	0.15	0.11	0.01*

RDI : Relative Deviation Index.

(b) Box Plot of Average Flow Time



EDDR과 MDDQ, FTLR은 재작업확률을 고려한 예상 공정소요시간을 기준으로 작업을 할당하므로 MMS와 EDD에 비해 좋은 성능을 나타냈다. 또한 재작업확률의 증가 대비 평균 공정소요시간의 증가도 MMS와 EDD만큼 급격하지 않았다. 세 알고리즘이 MMS와 EDD와는 달리 재작업확률을 고려하는 것을 반영하는 결과이다. 또한 가공시간의 차이가 커질수록 평균 공정소요시간이 감소하는 경향을 보였다. 특히 FTLR은 12가지 실험 Case에서 모두 타 알고리즘 대비 평균 42에서 76만 큼 차이로 가장 최소의 평균 공정소요시간을 보였다.

EDDR은 가공시간과 순서의존적 작업준비 시간을 포함하지 않은 재작업확률만으로 우위를 점하는 기계에 작업을 1차 배분하므로 공정소요시간 단축에 비효율적이었다. EDDR이 유휴기계에서 작업을 선택하는 최종기준은 가장 빠른 예상 공정완료시간이다. 따라서 EDDR 알고리즘 하에서는 기계그룹 전반에 걸쳐 평균적인 가공시간이 가장 작은 작업이 할당 우위를 점하게 되므로 공정한 작업할당이 이루어지지 않았다.

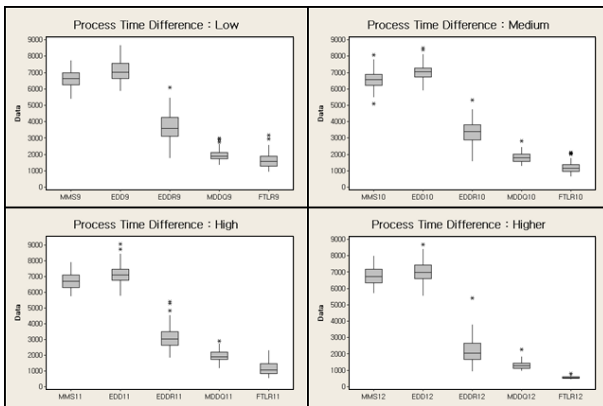
Table 9. Experiment Result-Average Flow Time
(Rework Rate : High)

(a) RDI of Average Flow Time

Index	Process Time Difference	MMS	EDD	EDDR	MDDQ	FTLR
RDI	Low	0.84	0.90	0.43	0.20	0.16*
	Medium	0.82	0.89	0.39	0.18	0.09*
	High	0.84	0.90	0.36	0.20	0.09*
	Higher	0.85	0.88	0.23	0.11	0.01*

RDI : Relative Deviation Index.

(b) Box Plot of Average Flow Time



MDDQ는 납기를 어긴 대기작업이나 여유시간이 공정소요시간보다 작은 대기작업에는 SPT 규칙을 반영하므로 공정소요시간 단축의 효과가 일부 있었다. 그러나 MDDQ의 SPT 규칙은 FTLR과는 달라서 이중병렬기계에서 작업에 대한 기계의 상대적인 우선순위가 기준이 아니므로 공정소요시간 단축에 그 효과가 제약적이었다. 또한 MDDQ는 납기까지의 여유가 예상 공정소요시간의 크기보다 큰 대기작업에 대해서는 MMS와 동일하게 여유시간의 길이로 작업을 할당하므로 MMS와 같이 공정소요시간이 큰 작업을 선택하는 문제가 발생하였다.

반면, FTLR은 이중병렬기계 환경까지도 고려한 알고리즘으로 MMS와 EDD는 물론이고 재작업확률과 순서의존적 작업준비시간만을 고려한 EDDR와 MDDQ 보다 공정소요시간 최소화에 더 좋은 성능을 보였다. 특히, 재작업확률이 커질수록 타 알고리즘과의 평균 공정소요시간의 차이가 큰 것으로 보아, 재작업확률이 높을수록 FTLR이 공정소요시간 최소화에서 더 효율적으로 작업을 할당하는 것을 알 수 있다. 또한 FTLR의 평균 공정소요시간의 표준 편차가 타 알고리즘 대비 작았으므로 동일 환경과 동일 작업 발생에서 FTLR이 타 알고리즘보다 더 안정적으로 작업을 할당할 것을 확인했다.

평균 납기 지연시간은 재작업확률별로 정리하면 <Table 10> ~<Table 12>와 같다. 평균 납기 지연시간은 평균 공정소요시간과 비슷한 결과를 보였다. 평균 납기 지연시간에서도 MMS와 EDD의 평균 납기 지연시간이 EDDR, MDDQ, FTLR과는 큰

차이로 가장 큰 값을 보였다. <Table 10> ~, <Table 11>, <Table 12>(a)에서 기계간 가공시간의 차이가 증가함에 따라 EDDR, MDDQ, FTLR의 평균 납기 지연은 줄어들는데 반해, MMS와 EDD의 납기 지연은 변동이 없었다. EDD와 MMS는 본래 납기 위주의 작업할당이지만 본 연구의 공정 환경에서는 오히려 공정소요시간 증가로 납기에서도 납기 지연을 초래했다. MMS와 EDD가 납기를 우선함에도 납기 만족을 이룰 수 없었던 것은 알고리즘이 재작업확률과 이중병렬기계를 고려하지 않고 납기를 기준으로 작업을 기계에 투입하기 때문이다. 즉, 재작업에 의한 공정소요시간의 증가와 작업에 유리한 기계를 선별하지 못하여 발생하는 공정소요시간의 증가에 기인하여, 선행 작업의 기계 점유율이 높아지고, 상대적으로 대기공간에서의 대기시간이 증가하게 되므로 두 알고리즘은 납기 만족을 이룰 수 없는 것이다. MMS와 EDD가 이중병렬기계와 재작업확률이 존재하는 작업 환경에 있어서 효율적이지 못한 알고리즘이라는 것을 다시 한 번 확인할 수 있다.

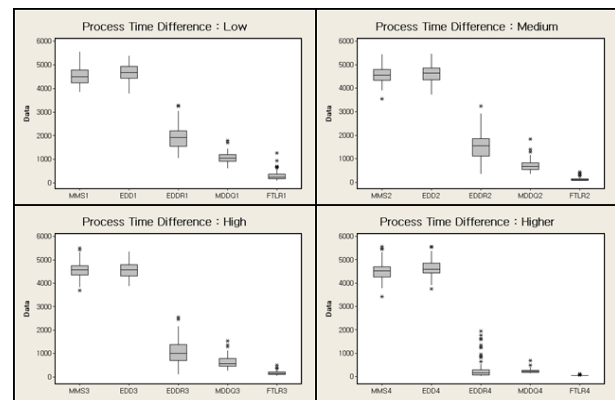
Table 10. Experiment Result-Average Tardiness(Rework Rate : Low)

(a) RDI of Average Tardiness

Index	Process Time Difference	MMS	EDD	EDDR	MDDQ	FTLR
RDI	Low	0.881	0.908	0.367	0.201	0.054*
	Medium	0.887	0.900	0.296	0.131	0.019*
	High	0.889	0.887	0.206	0.118	0.025*
	Higher	0.878	0.900	0.056	0.038	0.001*

RDI : Relative Deviation Index

(b) Box Plot of Average Tardiness



반면, EDDR과 MDDQ, FTLR은 평균 납기 지연 시간에서도 MMS와 EDD에 비해 좋은 성능을 가진다. 납기 지연에 대한 세 알고리즘의 성능은 세 알고리즘의 평균 공정소요시간이 MMS와 EDD 보다 짧았던 데서 기인한다. 특히, FTLR의 평균 납기 지연 시간은 타 알고리즘에 비해 월등히 낮아서 성능이 좋았던

Table 11. Experiment Result-Average Tardiness (Rework Rate : Medium)

(a) RDI of Average Tardiness

Index	Process Time Difference	MMS	EDD	EDDR	MDDQ	FTLR
RDI	Low	0.864	0.904	0.391	0.219	0.102*
	Medium	0.867	0.898	0.332	0.172	0.052*
	High	0.876	0.908	0.273	0.197	0.063*
	Higher	0.868	0.898	0.107	0.061	0.002*

RDI : Relative Deviation Index.

(b) Box Plot of Average Tardiness

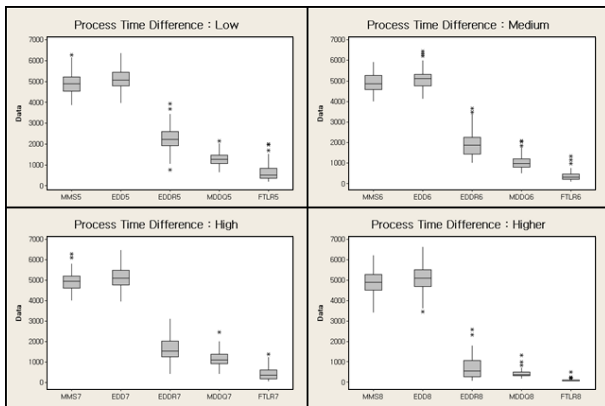


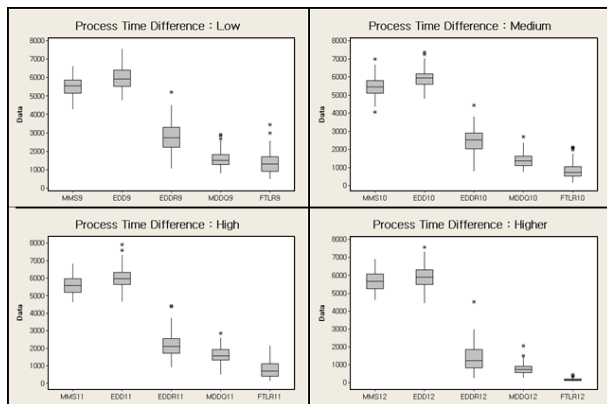
Table 12. Experiment Result-Average Tardiness(Rework Rate : High)

(a) RDI of Average Tardiness

Index	Process Time Difference	MMS	EDD	EDDR	MDDQ	FTLR
RDI	Low	0.833	0.899	0.406	0.221	0.190*
	Medium	0.820	0.892	0.364	0.195	0.104*
	High	0.843	0.907	0.319	0.226	0.099*
	Higher	0.849	0.886	0.184	0.092	0.003*

RDI : Relative Deviation Index.

(b) Box Plot of Average Tardiness



MDDQ 보다 대체적으로 절반 이하로 가장 짧았다. 재작업 확률이 High인 <Table 9>(a)와 <Table 9>(b)에서 가공시간 차이가 Low일 때 MDDQ의 평균 공정소요시간이 FTLR과 비슷하였으므로 납기지연에 있어서도 재작업확률이 High인 <Table 12>(a)와 <Table 12>(b)에서 가공시간 차이가 Low인 경우 MDDQ는 FTLR과 거의 같은 성능을 보이는 것을 확인할 수 있다. 그러나 재작업확률이 높고 가공시간 차이가 가장 작은 경우를 제외하고 FTLR은 타 알고리즘 대비 평균 공정소요시간이 짧아서 동일 작업 수라도 빠르게 처리하므로 납기지연도 상대적으로 적었다.

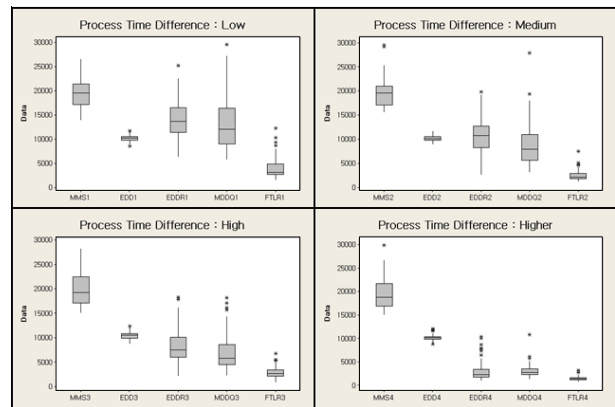
Table 13. Experiment Result-Max Tardiness in queue(Rework Rate : Low)

(a) RDI of Max Tardiness in queue

Index	Process Time Difference	MMS	EDD	EDDR	MDDQ	FTLR
RDI	Low	0.903	0.441	0.630	0.488	0.077*
	Medium	0.909	0.438	0.472	0.268	0.025*
	High	0.895	0.430	0.306	0.192	0.020*
	Higher	0.892	0.441	0.062	0.020	0.0001*

RDI : Relative Deviation Index.

(b) Box Plot of Max Tardiness in queue



<Table 13> ~ <Table 15>는 실험 완료시점에서 대기공간 내 잔류작업의 최대 납기지연시간의 평균을 각 Case별로 재작업 확률 구분에 따라 정리한 것이다. 평균 공정소요시간과 납기지연에서 MMS와 EDD가 비슷한 성능을 지녔던 반면, <Table 13>, <Table 14>, <Table 15>(a)와 <Table 15>(b)에서 확인할 수 있듯이, 대기작업의 최대 납기지연시간에서는 EDD가 MMS보다 더 좋은 성능을 보였다. 가공시간을 포함하여 납기까지의 여유시간을 고려한 MMS와는 달리, 납기 우선의 EDD는 다른 요소는 고려하지 않고 가장 빠른 납기만을 기준으로 작업을 기계에 할당하므로 대기작업의 최대 납기지연시간은 짧고 안정적이었다. <Table 13>, <Table 14>, <Table 15>(b)에서 보듯이 EDD는 타 알고리즘보다 산포가 작은 안정적인 성능을 보이지

Table 14. Experiment Result-Max Tardiness in queue(Rework Rate : Medium)

(a) RDI of Max Tardiness in queue

Index	Process Time Difference	MMS	EDD	EDDR	MDDQ	FTLR
RDI	Low	0.884	0.377	0.626	0.642	0.157*
	Medium	0.892	0.378	0.486	0.466	0.067*
	High	0.909	0.371	0.371	0.447	0.063*
	Higher	0.893	0.378	0.127	0.073	0.001*

RDI : Relative Deviation Index.

(b) Box Plot of Max Tardiness in queue

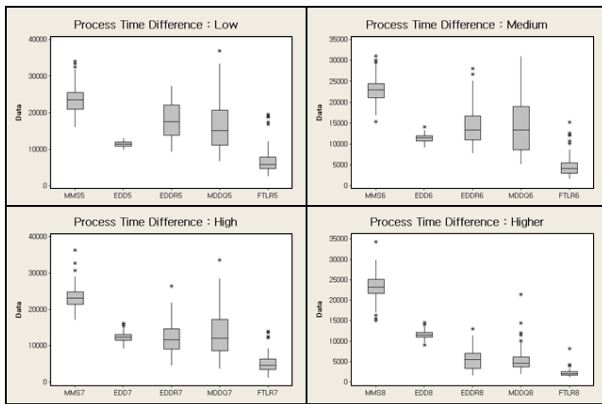


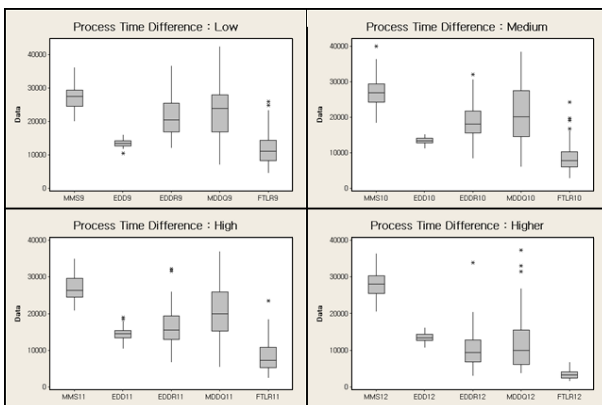
Table 15. Experiment Result-Max Tardiness in queue(Rework Rate : High)

(a) RDI of Max Tardiness in queue

Index	Process Time Difference	MMS	EDD	EDDR	MDDQ	FTLR
RDI	Low	0.908	0.363	0.640	0.829	0.283*
	Medium	0.892	0.361	0.539	0.692	0.156*
	High	0.888	0.357	0.448	0.631	0.125*
	Higher	0.906	0.363	0.243	0.254	0.001*

RDI : Relative Deviation Index.

(b) Box Plot of Max Tardiness in queue



만, FTLR, MDDQ, EDDR에 비하여 평균 공정소요시간의 증가로 인해 대기작업의 최대 납기지연 시간을 단축에는 한계가 있었다. EDDR도 작업을 1차 분류하는 시점에 납기순서로 작업을 정렬하여 납기가 빠른 작업을 우선하므로 대기작업의 최대 납기지연 시간에 있어서는 비교적 좋은 성능을 보였다. MMS는 납기를 어긴 대기작업은 FIFO 기준으로 작업할당을 하므로 납기순서가 고려되지 않아 성능이 좋지 않았다. MDDQ는 납기를 어긴 작업에 대해서는 SPT를 적용해 할당하므로 대기작업의 납기만족에 있어서는 성능이 좋지 않았다. 또한 여유시간을 기준으로 MDDQ가 작동할 때에는 기계가 재작업확률과 가공시간이 큰 작업을 선호하는 경향이 생겨서 재작업 발생빈도가 높아지고 공정소요시간이 증가한다. 따라서 공정을 신속히 빠져 나가지 못하는 작업들이 계속 발생하여 납기내 완료되지 못하고 대기공간에서 대기하는 작업도 많아지므로 납기준수 성능이 떨어진다. 특히, 재작업확률이 높아지는 <Table 14>와 <Table 15>에서 보면, MDDQ는 EDD와 EDDR보다도 성능이 떨어졌다.

반면, FTLR은 대기작업의 최대 납기지연 시간이 현저히 짧아서 대기작업의 최대 납기지연 시간에 있어서는 다른 알고리즘보다 성능이 우수함을 확인하였다. 특히, 가공시간의 차이가 가장 큰 3가지 Case의 경우, 납기지연 시간이 월등히 짧고, 산포가 상당히 좁아서 매우 우수하고 안정적인 성능을 보인다. 가공시간의 차이가 가장 큰 3가지 Case를 제외하고 재작업확률에 따라 FTLR의 성능을 두 번째로 성능이 좋은 알고리즘과 비교하면, FTLR의 최대납기지연은 재작업확률이 Low인 <Table 13>에서 평균 8.8배, 재작업확률이 Medium인 <Table 14>에서 평균 4.6배, 재작업확률이 High인 <Table 15>에서 평균 2.2배 짧았다. 특히, 가장 성능이 안 좋은 MMS 보다는 모든 실험 Case에서 최소 3.2배. 최대 10000배 이상의 좋은 성능을 보였다. FTLR은 본 연구의 공정 환경을 모두 반영한 예상 공정소요시간을 상대적 우위정도로 평가하기 때문에 일부 환경에 편향적인 작업할당을 지양하게 된다. 따라서 FTLR은 잔류작업과 할당작업에 대한 공평한 작업할당을 통해 타 알고리즘 대비 잔류작업의 최대 납기지연이 최소화였다.

이상의 실험결과와 분석을 통해 이중병렬기계와 재작업확률을 고려할 때 공정소요시간을 단축하는데 있어 FTLR이 좋은 성능을 보임을 확인하였다. 또한, FTLR은 재작업확률과 가공시간 차이의 변동에 불구하고 실험의 모든 경우에 있어 가장 좋은 성능을 보인 알고리즘이었다. 특히, 실험결과를 통해 FTLR이 달성한 공정소요시간 최소화는 납기만족과 대기작업내 고른 작업분배를 발생시킨다는 점을 확인하였으므로 더욱 의미가 깊다.

4. 결론

기업은 고유가 및 원자재 가격 상승에도 이윤을 창출해야하고 고객의 빠른 요구변화에도 신속한 대응을 통해 경쟁력을 키워

야 한다. 또한 기업은 관리를 통한 생산효율화를 이루기 위해 단기간 저비용으로 성과를 이뤄야 한다. 본 연구는 생산효율을 고려한 FTLR 작업할당 방법을 제안하였다.

FTLR은 이중병렬기계로 구성된 공정에서 재작업확률과 순서의존적 작업준비시간을 고려한 작업할당이다. FTLR은 공정특성을 환산한 예상 공정소요시간을 기준으로 FTLR 가중치를 구한다. FTLR 가중치는 공정소요시간 손실률을 기계별로 산정하여 0과 1사이의 값으로 정규화한 값이다. FTLR 가중치는 정규화를 통해 공정소요시간의 절대적인 크기에 상관없이 기계 간, 작업 간 비교가 가능하다. FTLR 가중치는 손실률이 가장 작은 작업 즉, 평균 공정소요시간을 기준으로 상대적으로 가장 작은 공정소요시간을 갖는 작업이 가장 큰 값을 가진다. FTLR은 FTLR 가중치가 가장 큰 작업을 기계에 할당하여 공정소요시간을 최소화한다. 본 연구의 공정조건을 모두 만족하는 알고리즘이 기존에 없었으므로 FTLR은 성능평가를 위해 재작업확률과 순서의존적 작업준비시간을 고려하였지만 이중병렬기계는 고려하지 않은 EDDR과 MDDQ, 납기지연을 고려한 EDD와 MMS와 동일조건에서 비교평가 되었다. 실험을 통해 이중병렬기계 환경까지 고려한 공정 환경에서는 FTLR이 납기 만족과 공정소요시간 최소화에 효율적임을 확인하였다.

FTLR은 기업이 당면한 내외부적 환경 하에 공정소요시간 최소화를 통해 투입대비 생산효율을 높일 수 있으므로 기업이 관리를 통해 생산효율을 달성하도록 한다. 특히, FTLR은 기계간 성능차이나 기계노후화에 따라 발생하는 이중병렬기계를 고려하였으므로 기계뿐 아니라 이중의 생산능력이 존재하는 다양한 생산체제에 응용이 가능하다. 예를 들어, 1인 완결형 Cell 조립생산에서도 작업자의 숙련도에 따른 조립시간과 재작업확률의 차이가 발생하므로 FTLR 작업할당이 적용가능하다. 따라서 FTLR은 다른 작업할당방법과 비교하여 실제 제조현장의 조건을 더 근접 반영한 작업할당방법이므로 실제 현장에서의 활용도가 높을 것으로 기대된다.

하지만 FTLR은 실시간 작업할당으로 할당시점마다 갱신되는 정보를 고려하지만, 정보가 대기공간의 대기작업과 순서의존적 작업준비시간으로 제한적이다. 그러나 유희기계 뿐 아니라 타 기계의 작업완료정보와 작업이력을 기준으로 할당시점마다 변동되는 정보를 추가적으로 고려한다면 동적 공정 환경에서 더 효율적인 작업할당을 기대할 수 있다. 또한 FTLR이 납기중요도를 가중치에 직접 반영한다면, 생산효율뿐 아니라 납기준수에서도 더욱 좋은 성능이 예상된다. 따라서 FTLR의 효율성을 높이기 위해 문제 환경의 공정조건이 동적으로 변화하는 정보를 추가적으로 고려한 연구와 납기준수와 생산효율을

동시에 고려하는 연구가 필요하다.

참고문헌

- Chung, N. K. and Jeong, M. Y. (2000), A Scheduling Support System for Non-identical Parallel Machine Lines, *Korean management science review*, **17**(2), 67-73.
- Drozdzowski, M. and Dell'Olmo, P. (2000), Scheduling multiprocessor tasks for mean flow time criterion, *Computers and operations research*, **27**(6), 571-585.
- Kang, Y. H., Kim, S. S., Park, J. H., and Sin, H. J. (2007), Scheduling Algorithm for Nonidentical parallel Machines Problem with Rework, *Journal of the Korean Institute of Industrial Engineers*, **33**(3), 329-338.
- Kang, Y. H., Lee, H. C., and Kim, S. S. (1998), Scheduling Jobs with different Due-Date on Nonidentical Parallel machines, *Journal of Korean Institute of Industrial Engineers*, **24**(1), 37-50.
- Kang, Y. H. and Shin, H. J. (2010), An adaptive scheduling algorithm for a parallel machine problem with rework processes, *International journal of production research*, **48**(1), 95-115.
- Kim, C. O. and Shin, H. J. (2003), Scheduling jobs on parallel machines : a restricted tabu search approach, *International journal of advanced manufacturing technology*, **22**(3/4), 278-287.
- Ko, H. H. (2010), *A Study on Dispatching Methods that Simultaneous Assuring Quality and Due Dates for Quality and Productivity with estimated completion time*, The Graduate School Korea University, Seoul, Korea.
- Ko, H. H. and Baek, J. K. (2010), Study on Dispatching for Quality and Productivity with estimated completion time, *The Korea Academia-industrial cooperation Society*, **11**(11), 1095-1100.
- Lee, E-C., Kim, Y-D., Kim, J-G., and Choi, S-H. (2003), A dispatching rule-based approach to production scheduling in a printed circuit board manufacturing system, *Journal of the Operational Resaerch Society*, **54**(10), 1038-1049.
- Lee, Y. H. and Pinedo, M. (1997), Scheduling jobs on parallel machines with sequence-dependent setup times, *European Journal of Operational Research*, **100**(3), 464-474.
- Lee, Z. J., Lin, S. W. and Ying, K. C. (2010), Scheduling jobs on dynamic parallel machines with sequence-dependent setup times, *International Journal of Advanced Manufacturing Technology*, **47**(5/8), 773-781.
- Pinedo, M. (2002), *Scheduling Theory, Algorithms, and Systems*, Second Edition, Prentice-Hall, Upper Saddle River, New Jersey 07458.
- Turkcan, A., Akturk, M. S., and Storer, R. H. (2003), Non-identical parallel CNC machine scheduling, *International Journal of Production Research*, **41**(10), 2143-2168.