

# 품질과 납기를 동시에 만족하는 작업투입 개선에 관한 연구

윤지명<sup>1</sup>, 고효현<sup>1\*</sup>, 백종관<sup>2</sup>, 김성식<sup>1</sup>

## Improved Dispatching Algorithm for Satisfying both Quality and Due Date

Ji-Myoung Yoon<sup>1</sup>, Hyo-Heon Ko<sup>1\*</sup>, Jong-Kwan Baek<sup>2</sup> and Sung-Shick Kim<sup>1</sup>

**요약** 생산 공정의 효율화는 모든 제조 산업이 추구하고 있는 목표이다. 본 논문에서는 다양한 타입의 제품을 병렬기계에 투입 할 때 목표한 품질을 만족시키고 동시에 평균 납기 지연 시간을 최소화 하는 효율적인 작업투입방법에 대하여 기술하였다. 본 연구에서는 품질을 고려하고 실시간으로 작업을 투입해야 하는 공정에서 Rolling Horizon 타부탐색(Tabu search)방법을 이용하여 평균 납기 지연시간을 최소화 하는 방안을 개발하였다. 시물레이션을 통해 제안하는 방법의 효율성을 비교대안과 비교하였다. 본 방법은 실제 생산 공정에서 품질과 납기를 동시에 만족시킬 수 있는 효율적인 작업투입 방법으로 사용되어 기업의 수익향상과 신뢰성 확보를 이룰 수 있다.

**Abstract** The manufacturing industry seeks for improvements in efficiency at the manufacturing process. This paper presents a method for effective real time dispatching for parallel machines with multi product that minimizes mean tardiness and maximizes the quality of the product. What is shown in this paper is that using the Rolling Horizon Tabu search method in the real time dispatching process, mean tardiness can be reduced to the minimum. The effectiveness of the method presented in this paper has been examined in the simulation and compared with other dispatching methods. In fact, using this method manufacturing companies can increase profits and improve customer satisfaction as well.

**Key Words** : Scheduling, Dispatching, Quality, Tabu search, Rolling horizon tabu search, Parallel machine

### 1. 서론

최근 제조 기업들은 수출향상을 통한 최대 이윤창출에서 고객만족이라는 새로운 패러다임에 맞는 제조전략을 수립하여 실행하고 있다. 고객만족은 우수한 품질의 제품을 생산하여 납기 내에 공급하는 것으로 기업의 생존과 직결된 중요한 관리변화를 가져왔다.(Pinedo, 1999)

기업들은 납기준수와 제품의 품질 간에 상충관계(Trade-Off)에서 어려움을 겪고 있다. 특히, 높은 품질 수준의 제품은 높은 가격으로 거래되어 높은 고객 만족도를 주며 기업경쟁력의 원천이 된다. 기업들은 고객 만족과 높은 수익을 위해 높은 품질의 제품을 생산하려고 한다. 그러나 모든 기계들이 만족할 수 있는 품질을 생산하

지 못한다. 따라서 높은 품질의 제품을 만드는 기계로 생산에서 제약을 수행한다. 생산에서 사용하는 제약은 투입 가능한 기계를 한정시켜 가용기계의 부족으로 작업들의 대기시간을 증가시키게 되고, 대기시간 증가는 총 가공시간을 증가시켜 납기 준수를 어렵게 한다. 즉, 품질을 높이려 오�히려 납기 지연에 따른 경제적손실과 고객만족도 저하로 인해 기업의 시장점유율 상실을 초래하게 된다. 반면, 기업들이 납기 준수를 위해 품질을 고려하지 않고 작업 투입을 지시한다면, 작업이 낮은 품질로 가공되는 기계에 투입되어 전반적인 품질 하락을 피할 수 없게 된다. 낮은 품질은 고객의 요구를 충족시키지 못해 매출감소 및 경쟁력 저하로 이어진다.

따라서 기업이 무한경쟁 속에서 경쟁력을 확보하기 위해서는, 작업투입이 높은 품질과 납기 준수가 동시에 만

본 논문은 교육인적자원부, 산업자원부, 노동부의 출연금으로 수행한 최우수실험실지원사업의 연구결과입니다.

<sup>1</sup>고려대학교 정보경영공학과

교신저자: 고효현(swish17@korea.ac.kr)

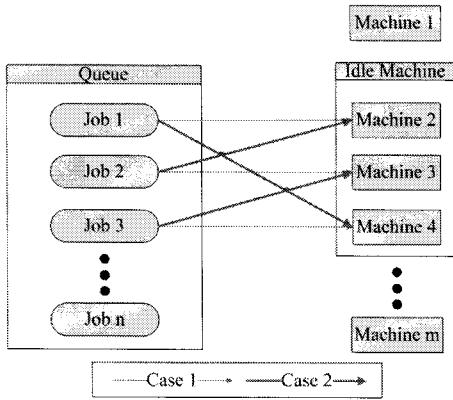
접수일 08년 10월 16일

수정일 08년 11월 14일

<sup>2</sup>서일대학 산업시스템경영과

게재확정일 08년 12월 16일

족될 수 있도록 효율적이고 올바른 지시를 수행해야한다. 품질과 납기 준수를 동시에 고려한 효율적인 작업투입선택을 수행 하는 경우는 [그림 1]과 같이 설명될 수 있다.



[그림 1] 작업투입시 고려사항

[표 1] 작업할당에 따른 납기 지연

Case 1	Machine 2	Machine 3	Machine 4
Job 1			Tardy=1
Job 2		Tardy=1	
Job 3	Tardy=1		
Case 2	Machine 2	Machine 3	Machine 4
Job 1	Tardy=2		
Job 2			Tardy=0
Job 3		Tardy=0	

대기 공간에는 연속적으로 들어오는  $n$ 개의 작업이 있다. 대기공간에 있는  $n$ 개의 작업은  $m$ 개의 기계 중에 유후 기계가 발생하면 대기공간의 작업들의 가중치를 계산하여 그 순서대로 유후기계에 할당을 수행한다. 작업의 가중치는 납기와 품질의 만족을 고려하여 가중치를 산정한다.

만약 [그림1]과 같이 작업 1, 2, 3이 작업장의 기계들에 대해서 품질을 만족하며 투입이 가능할 때, 작업1의 중요도가 가장 높다면 투입 가능한 유후 기계들 중에서 한 기계에 할당되게 된다.

만일 다른 유후기계가 존재하지 않을 경우, 가장 높은 중요도를 갖는 작업의 투입은 품질과 납기에 관련된 좋은 정보를 제공해줄 수 있다. 하지만 다른 유후 기계들이

존재할 경우 가중치 순서대로 작업을 할당하는 것은 전체 작업의 할당을 고려할 때, 효과적이지 않을 수 있다.

예를 들어, [그림1]과 같이 계산된 가중치의 순서에 따라 1번 작업이 기계2로, 2번 작업은 기계3으로, 3번 작업은 기계4로 할당된다고 하자. 이때 각 작업은 납기지연이 최소가 되도록 순서대로 할당되어 [표1]과 같이 각각의 납기지연시간이 1을 갖는다. 그러나 가중치가 가장 높은 1번 작업을 4번 기계로 할당 한다면 작업 1의 납기지연은 2가되지만 작업2와 작업3의 납기지연시간은 0이 되어 평균 납기지연시간은 약 0.67로 낮아진다. 즉, 중요도가 높은 작업을 개별적으로 판단하여 품질 만족여부 및 납기지연 시간을 비교하면 공정의 최적화는 달성할 수 없다. 특히, 연속적으로 투입되는 작업들에 대해 작업투입 방법이 제시한 해를 그대로 사용한다면 납기지연이 점차 누적되어 공정의 효율이 감소하게 된다. 즉, 연속 투입되는 작업들에 대해 품질을 만족하고 동시에 납기지연시간을 최소화 할 수 있는 작업투입방법이 필요하다. 따라서 본 연구는 병렬기계에서 연속적인 작업투입이 이루어 질 때, 작업의 품질을 만족하며 평균 납기지연 시간을 최소화하는 효율적인 작업투입 방법을 제안한다.

본 연구의 문제에서는 두 가지 고려해야할 사항이 존재한다. 첫째는 품질과 납기준수를 고려한 초기 해를 구해야 하고, 둘째는 효율적인 작업투입을 위해 초기 해를 개선해야 한다. 본 연구의 진행에 필요한 두 가지 고려사항에 대해 기존연구를 통해 효율적인 대안이 존재하는가를 확인하였다.

제조회사들은 납기 준수를 위해 FIFO(First In First Out), EDD(Earliest Due Date), SPT(Shortest Process Time) 등과 같은 단순한 작업투입방법이외에도 효율적인 작업투입 방법들에 대해 많은 연구들을 수행해 왔다. Lee(1997)는 작업준비시간이 존재하는 단일기계의 TWT(Total Weighted Tardiness)최소화 문제의 해법으로 ATCS(Apparent Tardiness Cost With Setup) 규칙을 사용하는 3단계 발견적기법 절차를 제안하였고 우수함을 보였다. 또한 Lee and Pinedo (1997)는 병렬기계에서 납기지연의 총합을 최소화 하는 문제를 풀기위해 할당규칙과 시뮬레이티드 어닐링(Simulated Annealing)을 결합한 기법을 제시하였다. Lee and Pinedo(1997)의 병렬기계 문제는 Lee(1997)가 단일 기계 문제에서 제안하였던 ATCS규칙을 병렬 기계로 확장하여 사용하였다.

작업 투입 목적 중에 최대 납기지연시간(Maximum Lateness Tardiness 이하  $L_{max}$ )의 최소화를 목적으로 하는 일반적인 병렬기계문제는 NP-hard이다(Pinedo, 1999).

또한 순서의존적인 작업준비시간이 함께 존재하는 문제는 더욱 어렵다고 알려져 있다(Ovacik, 1995).

Ovacik(1995)은 예측구간(Forecast Window)내에서 작업들의 정보를 이용하여 작업 가능시점과 순서의존적인 작업시간이 존재하는 단일기계에서  $L_{max}$  문제를 풀었다. 그리고 Ovacik(1995)은 알고리즘의 수행속도상의 문제점을 해결하기 위해 같은 문제에 대해(Rolling Horizon Procedure 이하 RHP)알고리즘을 제시하였다.

Ovacik(1995)이 제시한 RHP는 기존의 할당규칙들은 물론 이들 할당규칙에 국지탐색(Local search)을 결합한 방법들보다 우수한 성능을 보인다는 것이 입증하였다.

Schutten(1996)은 동종 병렬기계의 납기지연의 합을 최소화하는 문제를 분지한계법을 이용하여 해결하였다. 그러나 분지한계법을 이용하면 수많은 이웃해를 생성해서 수행 속도상의 문제가 발생한다.

기존의 일정계획 문제나 작업할당문제는 생산효율 향상과 지연시간의 최적화만을 고려하고 품질은 고려되지 않았다. 한편 품질과 납기를 동시에 고려한 연구로는 Kang(2007)의 연구가 있다. Kang(2007)은 다양한 타입의 제품들이 가공되는 기계에서 품질을 Rework확률로 정의하여 품질과 납기를 Rework 확률에 따른 예상완료시간으로 추정하여 전체납기지연 시간을 줄이는 연구를 하였다. Kang(2007)의 일정계획 알고리즘은 대부분의 작업 할당 규칙이 갖는 근시안적인 한계를 지니고 있고 Rework 확률자체가 시간으로 확정되어 해를 개선시킬 수 없는 한계가 있다.

품질과 납기를 동시에 고려한 또 다른 연구로는 Ko(2008)연구가 있다. Ko(2008)의 연구는 본 연구와 같이 연속적으로 발생하는 작업에 대해 실시간으로 작업투입을 수행해야하는 문제 상황을 다룬다.

Ko(2008)는 연속적으로 발생하는 작업에 대해 품질과 납기순수를 위한 작업투입 방법을 제안 하였다. 작업투입 방법에 의해서 생성된 초기 스케줄은 납기지연시간을 단축하기 위해 1단계 국부탐색 방법을 적용하여 개선하였다. 1단계 국부탐색은 해의 개선을 위한 작업 완료시간을 기준으로 탐색하였다. 1단계 국부탐색은 납기를 고려하지 않고 빠른 완료시간을 갖는 작업을 기준으로 기계에 할당하는 방법이다. 1차 국부 탐색은 가장 높은 가중치를 기준으로 순서대로 작업을 선택하여 최적의 완료시간을 산출하여 투입하게 된다. 그러나 여러 기계와 작업이 존재할 경우 먼저 탐색이 완료된 작업으로 인해 남은 작업들이 최적의 완료시간을 갖지 못해 전체 납기 지연시간을 증가시키는 단점을 가지고 있다.

대부분의 기존 연구에서 일정계획수립 문제는 근시안적인 한계를 벗기 위해 효율적인 해의 개선을 요구하고 있다. 효율적인 해의 개선을 위해 기존 연구들은 타부 탐색(Tabu Search), 시뮬레이티드 어닐링(Simulated

annealing), 유전 알고리즘(Genetic Algorithm)과 같은 메타 휴리스틱을 사용하였다. 메타 휴리스틱은 조합 최적화 문제에 많이 적용되어 왔다. 해의 개선에 사용되는 일반적인 메타휴리스틱은 작업의 수가 증가함에 따라 수행속도가 늘어나 실시간으로 해를 개선하여 작업을 투입하는 동적인 상황에서는 한계를 갖는다.

그러나 대상문제의 특성에 맞도록 핵심요소들을 효과적으로 사용하면 메타휴리스틱의 시간적인 한계는 극복할 수 있다. 즉, 실시간적인 해의 개선에 적합하도록 메타휴리스틱의 세부적인 핵심요소를 활용한다면 동적인 작업투입 상황에서 좋은 성능을 보일 수 있다.

기존의 메타휴리스틱 중에서 Glover(1989, 1990)에 의해 현재의 형태로 정립된 타부탐색은 국부 최적 해에 빠질 위험이 적으며, 대상문제 고유한 특징 및 지식을 잘 적용할 수 있다. 타부탐색은 대상문제의 특성에 맞는 핵심요소들을 효과적으로 사용하느냐에 따라 탐색의 성능이 좋아질 수 있다. 타부탐색은 탐색을 위한 이동을 제한을 두어 최적 해를 구하기 위한 탐색을 수행하기 때문에 불필요한 이동을 감소시켜 탐색의 속도를 향상시킬 수 있다(Franca 외, 1996).

타부탐색은 기본적으로 임의적인 이웃해 탐색기법이며 해의 개선을 위해 이동(move)연산을 반복적으로 사용하여 해 영역에서 탐색을 수행한다. 타부탐색의 열망수준(Aspiration)은 탐색을 수행할 때 국부 최적해(Local Optimal)에 머무는 것을 방지하고 좋은 해를 찾기 위해 목적함수 값을 일시적으로 저하 시켜도 이동을 허용한다. 타부탐색은 각 이동정보인 타부속성을 일정기간(Tabu Tenure)동안 타부목록(Tabu List)에 기억함으로써 탐색의 순환을 방지할 수 있다. 타부 속성은 해의 개선을 위한 탐색과정에서 불필요한 이동을 제한하기 때문에 탐색을 위한 이웃해의 수를 줄여 탐색의 시간을 단축시킨다.

타부탐색 알고리즘은 효율적인 작업계획을 위한 연구에서 많이 이용되었다. Armentano(2000)는 병렬기계 일정계획 문제에서 납기지연의 평균을 최소화 하기 위한 타부탐색 기법을 제시하였다. Shin(2002)은 순서 의존적인 작업의 준비시간과 시작가능시간이 존재하는 병렬기계에서  $L_{Max}$  를 갖는 작업을 최소화 하고, 해의 개선에 필요한 탐색의 수행시간을 단축시키기 위해 제한적 타부탐색(Restricted Tabu Search)방법을 제시하였다.

한편, 타부탐색과 함께 동적인 작업 투입상황에서 실시간으로 해를 개선하기 위한 효율적인 탐색방법으로 Rolling Horizon Procedure(이하 RHP)가 있다. RHP는 큰 일정문제를 풀 때 시간 축으로 나타난 일정들을 세부적으로 계산하고 집합으로 모아서 조합하는 체계적인 접근 방법이다. 시간 축으로 나타난 전체 문제를 일정간격으로

구분하고, 구간별로 연산을 수행하기 때문에 불필요한 연산을 줄여 탐색의 속도를 향상시킬 수 있다. 따라서 RHP는 계산에 필요한 시간이 적게 들고, 동적인 상황에서 적용할 수 있다는 장점이 있다.(Pinedo, 2002) 따라서 RHP는 시간 축을 따라 실시간으로 대상 작업들이 발생하고 할당될 때 적합한 방법이 될 수 있다.

따라서 대상 문제의 특성에 따라 효과적인 사용을 할 수 있는 타부탐색과 해를 개선할 때 불필요한 연산을 줄이는 RHP를 접목한 RH타부탐색(Rolling Horizon Tabu search 이하 RHTS)을 통해 신속한 해의 개선을 위한 방안으로 사용할 수 있다.

Ko(2004)는 순서의존적인 작업 준비시간, 시작가능시간, 납기, 및 반복 공정을 가지는 제약적 병렬기계에서의 일정계획 수립의 문제에서 타부탐색을 사용하였다. 또한 Rolling Horizon을 사용하여 타부탐색알고리즘의 탐색속도를 향상 시켰다.

신속한 해의 개선을 수행하는 RHTS의 특징은 작업과 작업의 투입 사이에 모든 작업들에 대한 가상작업할당 스케줄(Dummy Schedule)을 가능하게 할 수 있다는 것이다. 특히, 본 연구의 문제 상황과 같이 연속적으로 발생된 작업을 기계에 할당할 때, 실시간으로 대기공간의 모든 작업을 대상으로 가상작업할당 스케줄을 수행하여 최적의 작업투입 지시를 가능하게 할 수 있다.

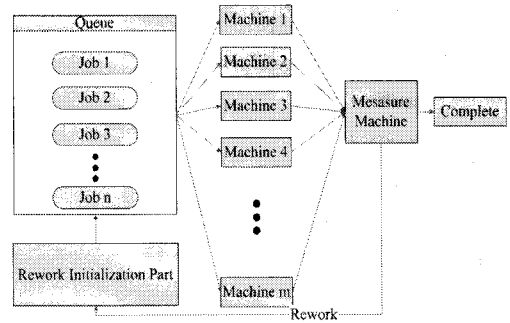
이상의 연구들에서 본 연구와 같이 실시간으로 발생하는 작업들에 대해 품질과 납기준수를 동시에 고려한 연구는 Ko(2008)의 연구를 제외하고 거의 없었다.

그러나 Ko(2008)의 연구도 작업에 대한 개선이 전체 작업 투입에 대한 최적보다 투입자체에 대한 최선책만을 지시하여 올바른 개선을 수행하지 못하였다. 따라서 본 연구는 실시간으로 발생할 작업들에 대해 품질과 납기준수를 동시에 고려하는 작업투입 방법에 타부탐색으로 개선한 효율적인 작업투입을 제안한다. 본 연구를 통한 작업투입은 품질수준과 납기를 동시에 만족하며, 근시안적인 초기해의 한계를 극복하고 짧은 시간에 최적 값에 근사할 수 있는 효율적인 실시간 작업투입방법이다.

## 2. 본론

### 2.1 문제의 정의

본 연구는 품질 만족과 납기준수가 가능한 실시간 작업투입 방법을 제안한다. 본 연구에서 다루는 공정 환경은 [그림 2]와 같다.



[그림 2] 전체 공정 구성도

전체 공정은 작업이 대기하는 대기 공간, 작업을 가공하는 작업장, 결과를 예측하는 검사장, 그리고 Rework을 위한 작업 초기화 영역으로 구성된다. 작업장의 기계는 제품타입에 따른 가공품질, 가공시간 및 작업준비시간을 갖는다. 가공시간은 제품타입별로 다르고, 같은 제품타입에 대해서는 모든 기계에서 동일하다. 작업준비시간은 기계에서 이전에 가공했던 제품의 타입과 현재 투입되는 제품의 타입에 따라 상이한 순서 의존적인 작업준비시간을 갖는다.

작업의 가공품질은 제품타입과 기계에 따라 정규분포  $N(\mu_{nm}, \sigma_{nm}^2)$ 을 따르며, 기계  $m$ 에서 가공된 제품타입  $n$ 의 평균을  $\mu_{nm}$ 이라 하고, 표준편차를  $\sigma_{nm}$ 이라고 정의한다. 가공수준은 기계와 각각의 제품타입 따라 다르다.

작업은 제품타입, 시작가능시간과 납기(Due Date)정보를 갖는다. 본 연구의 문제 상황에서는 Rework을 고려하므로 작업은 가공이 완료되면 Rework여부를 판정받는다. 작업은 양품으로 판정되면 작업 완료시간을 갖는다. 작업 완료시간은 작업이 기계에 투입되는 시간, 준비시간, 및 가공시간의 합으로 표현된다. 만일 작업이 Rework판정을 받으면, 작업 완료시간은 작업 가공에 소요됐던 시간, 초기화시간, 대기 공간까지 이동시간, 대기시간, Rework에 필요한 준비시간, 그리고 가공시간의 합이 된다.

작업발생은 사전에 계획되어 발생하지 않는다. 즉, 작업발생에 대한 정보를 미리 알 수는 없다. 작업투입은 대기공간에 존재하는 작업만을 대상으로 투입을 결정한다. 또한 작업 결과는 가공이 끝나야 알 수 있고 미리 Rework에 대한 판정과 계획도 수립할 수 없다. 기계에 투입될 작업은 유휴 기계가 발생했을 때 대기공간에 존재하는 작업들을 대상으로 정의된 투입방법에 따라 투입된다.

가공을 완료한 작업은 기계를 떠나 검사장에서 작업의 결과를 얻는다. 결과는 목표점과 예측점의 차이로 표현하

고 차이의 정도를 품질이라 한다. 작업의 품질에 따라 공정의 관리규격의 한계를 넘어서 작업은 Rework을 수행한다. 작업 결과는 기계에서 제품과 기계에 따른 품질 가공수준에 의해 결정된다. 기계에서 작업 품질을 결정하는 가공수준의 평균은 공정관리규격 안에 있으며 편차는 0 이상이다. 또한 Rework으로 판정된 작업은 폐기하지 않고 모두 Rework을 수행한다.

[그림 2]에서 전체적인 작업의 흐름을 정리하면 다음과 같다. 발생한 작업은 대기 공간으로 입력되어 대기한다. 만일 기계가 유휴(Idle)하게 되면 대기 중인 작업들을 대상으로 투입방법에 따라 작업을 선택하여 기계에 투입한다. 투입된 작업은 제품타입에 따른 준비시간과 가공시간 만큼 가공을 받고 계속 기계로 이동한다. 계속기계는 완료된 작업의 결과를 기준으로 Rework 여부를 판정한다. 작업결과에 따라 불량판정을 받은 작업은 가공초기화를 거쳐 Rework을 위한 작업대기 공간으로 이동한다. 작업결과가 양품이면 작업은 완료된다.

본 연구는 이상의 제시된 공정 환경에서 품질과 납기를 동시에 만족하는 효율적인 작업투입을 위해, 단순한 작업 투입을 넘어 전체 작업들의 최적 투입이 가능하도록 개선을 수행한 작업투입 방법을 제안한다.

## 2.2 전체 알고리즘 구성

본 연구는 품질수준을 높이며 특히 납기 준수가 동시에 가능한 효율적인 작업투입방법에 대한 연구이다. 본 연구가 수행되기 위해서는 품질과 납기에 대한 성능 평가 지표가 필요하다.

납기의 성능과 관련된 지표 중에, 본 연구의 상황과 같이 작업 수의 제한 없이 연속적으로 발생하는 문제일 경우 평균 납기지연시간이 사용된다. 따라서 본 연구의 성능 평가를 위해 식(1)과 같이 평균 납기지연시간을 사용한다.

$$\text{Min} \frac{1}{n} \sum_{j=1}^n \text{Max}(C_j - d_j, 0) \quad (1)$$

st  
 $Q_{j_i} > \text{Threshold}_i$

$j$  : 작업의 인덱스

$i$  : 제품타입의 인덱스

$C_j$  : 작업  $j$ 의 완료시간

$d_j$  : 작업  $j$ 의 납기

$Q_{j_i}$  : 제품타입  $i$ 인 작업  $j$ 의 품질

$\text{Threshold}_i$  : 제품타입  $i$ 의 최소 품질 임계값

식(1)에서의 작업 투입 목적은 연속해서 발생하는 작업에 대해 품질제약을 만족하며 평균 납기지연시간을 최소화하는 것이다. 식(1)에서  $\text{Max}(C_j - d_j, 0)$ 는 작업의 납기 지연시간으로 작업의 완료시간과 작업의 납기의 차를 표현한다. 또한 작업의 납기지연시간의 총합을 완료된 작업의 수로 나누어 평균 납기지연시간을 표현한다. 제약식에서는 가공이 완료된 작업들의 품질은 최소 한계를  $\text{Threshold}_i$  값 이상이다.  $\text{Threshold}_i$ 는 제조회사에서 규정한 제품타입  $i$ 의 최소 품질 임계값이다. 최소 품질 임계값은 완성된 제품 품질을 최소한 보증하기 위한 것이다. 따라서 품질에 대한 최소 한계를 규정함으로써 평균적인 품질은  $\text{Threshold}_i$  값을 상회하게 된다.

일반적으로 품질은 생산 공정의 목적에 따라 특성화되어 다양하게 사용되고 있다. 따라서 본 연구에서 사용될 품질을 정의할 필요가 있다. 특히 품질에 대한 제약조건 만족여부확인과 투입되는 작업간의 품질의 차이를 비교하기 위해 품질을 정의하고 표현하는 것이 필요하다. 본 연구에서 품질은 목표점과 가공후의 계속값에 대한 오차로 정의 한다. 이를 수식으로 표현하면 식(2)와 같다.

$$\text{Quality}_{j_i} = T_i - O_{j_i} \quad (2)$$

$i$  : 제품타입의 인덱스

$\text{Quality}_{j_i}$  : 제품타입  $i$ 인 작업  $j$ 의 품질

$T_i$  : 제품타입  $i$ 의 공정 목표치

$O_{j_i}$  : 제품타입이  $i$ 인 작업  $j$ 의 계속값

식(2)에서  $\text{Quality}_{j_i}$ 는 제품타입  $i$ 의 가공의 공정 목표값  $T_i$ 와 가공 후의 결과를 측정값  $O_{j_i}$ 의 오차이다. 작업품질은 제품타입별 기계의 가공품질수준으로 결정된다. 가공품질수준은 정규분포로 주어지기 때문에 평균만을 사용한다면 변동성의 영향으로 품질의 차이를 정확하게 구별하기 힘들다. 또한 변동성만으로 평가할 경우 목표치와의 차이를 구분하지 못한다. 따라서 가공품질수준의 평균과 변동성을 모두 이용한 공정을 대표할 수 있는 품질지표가 필요하다. 또한 동일기계에서 제품타입별로 품질의 차이를 표현할 수 있는 품질 지표가 필요하다.

본 연구에서는 공정을 대표할 수 있는 품질 지표로 Ko(2008)의 연구에서 사용한 공정능력지수를 이용한다. 공정능력지수는 설계자가 제품 타입별로 규정한 공정관리규격을 기준으로 공정의 품질능력을 평가한다. 공정능력지수는 결과 품질과 목표품질간의 이탈 도를 고려하여 품질을 표현하므로 비교적 정확하고 표준화된 품질지표를 제시할 수 있다. 식(3)은 공정능력지수에 대한 수식이다.

$$Q_i = Mn\left(\frac{USL_i - \mu_i}{3 \times \sigma_i}, \frac{\mu_i - LSL_i}{3 \times \sigma_i}\right) \quad (3)$$

$$\mu_i = \frac{1}{n_i} \sum_{j=1}^{n_i} q_{ji}, \quad \sigma_i = \sqrt{\frac{1}{n_i - 1} \sum_{j=1}^{n_i} (q_{ji} - \mu_i)^2}$$

- $Q_i$  : 제품타입  $i$ 의 공정능력지수
- $USL_i$  : 제품타입  $i$ 의 공정관리규격의 상한 값
- $LSL_i$  : 제품타입  $i$ 의 공정관리규격의 하한 값
- $\mu_i$  : 제품타입  $i$ 의 품질의 평균값
- $\sigma_i$  : 제품타입  $i$ 의 품질의 표준편차
- $q_{ji}$  : 제품타입  $i$ 인 작업  $j$ 의 개별품질
- $n_i$  : 제품타입  $i$ 인 작업의 전체 완료작업 수

식(3)의 공정능력지수는 공정관리 규격의 상·하한 값 사이에서 공정변동에 대한 규격변동의 양적인 표현과 공정평균과 규격중심의 일치하는 정도를 표현한 것이다. 공정능력지수는 개별 작업의 품질을 표현하지는 않지만 공정전반에 걸쳐 제품타입  $i$ 인 작업들의 목표품질 달성 정도를 표현한다. 따라서 본 연구에서는 공정능력지수를 품질표준화 지표인 품질 인덱스로 정의하여 사용한다. 공정능력지수는 공정의 품질평균이 공정 목표치에 근접하고 변동성이 작으면 높은 값을 갖고, 공정의 품질평균이 목표치와 차이가 많이 나고 변동성이 크면 작은 값을 갖는다.

본 연구에서는 공정능력지수를 사용하여 작업을 투입하는 시점에서 과거에 기계들이 제품타입별로 수행했던 가공의 품질분포와 품질제약 데이터를 기반으로 품질표준지표를 계산하여 비교한다.

이상의 품질과 납기 관련 지표를 이용하여 본 연구에서 제시하는 작업투입 방법은 크게 초기 해를 구하는 단계와 생성된 초기 해를 향상시키는 단계로 구성된다.

초기해를 구성하는 단계는 품질인덱스를 이용하여 품질제약을 만족시키는 작업에 대해 납기, 작업준비시간, 가공시간을 고려하여 효율적인 작업투입이 가능하도록 작업을 선택하게 된다. 본 연구의 목적은 품질제약을 만족하는 작업들에 대해 최종적으로 납기 만족을 수행하는 것이다. 따라서 본 연구에서는 품질과 납기, 및 순서의존적인 작업준비시간을 고려하여 작업의 가중치를 계산하는 Ko(2008)의 ATCSQ(Apparent Tardiness Cost With Setup and Quality)규칙을 이용하여 그 중요도에 따라 작업 투입을 결정한다.

해의 개선단계는 ATCSQ통해 선택된 초기해를 개선하여 목적하는 품질제약을 만족시키고 최적의 평균 납기 지연시간을 갖는 해를 구하도록 한다.

본 연구의 문제 상황에서는 실시간 작업할당을 수행하기 때문에 해의 개선에 많은 수행시간이 필요한 일반적인 메타휴리스틱을 사용할 수 없다. 따라서 연속적으로 작업이 발생하는 상황에서 사용할 수 있는 RHP를 적용한 RHTS를 적용하여 신속하게 최적해에 근사한 작업투입이 되도록 개선한다.

RHTS는 품질제약을 만족하고 목적함수 값인 평균 납기지연시간을 빠르게 최소화 하기위해 기계에 투입가능한 모든 작업들을 대상으로 하여 탐색을 수행한다.

즉, 본 연구는 연속적으로 발생하는 작업에 대해 품질과 납기를 동시에 고려하여 작업을 선택하고 선택된 작업의 개선을 실시간으로 수행하는 효율적인 작업투입 방법을 제시한다.

### 2.3 초기해 생성 알고리즘

초기해 생성방법은 품질과 납기준수를 위한 작업투입 방법 중에서 좋은 성능을 보였던 Ko(2008)의 ATCSQ(Apparent Tardiness Cost With Setup and Quality)규칙을 사용한다. ATCSQ는 대기하는 작업들에 대해 가공시간, 여유시간과 순서의존적인 작업준비시간 및 표준화된 품질 제약을 이용하여 작업가중치를 산출한다. ATCSQ방법으로 계산된 가중치가 높은 작업은 해당 기계에 대해서 대기하는 작업들 중 작업준비시간, 가공시간, 여유시간이 작고 품질이 좋은 작업이다. ATCSQ의 수식은 다음과 같다.

$$I_{jm(t,i)} = \frac{f_{jm}}{p_j} \exp\left(-\frac{Max(d_j - p_j - t, 0)}{k_1 \bar{p}}\right) \exp\left(-\frac{s_{ij}}{k_2 \bar{s}}\right) \quad (4)$$

$$f_{jm} = \begin{cases} \sqrt{(\exp(-Q_{jm}) - 1)^2} & Q_{jm} \leq Threshold_i \\ 0 & Q_{jm} > Threshold_i \end{cases} \quad (5)$$

$$Q_{jm} = Mn\left(\frac{USL_i - \mu_{mi}}{3 \times \sigma_{mi}}, \frac{\mu_{mi} - LSL_i}{3 \times \sigma_{mi}}\right) \quad (6)$$

$I_{jm}(t,i)$  : 시간  $t$ 에서 기계  $m$ 의 가공 작업  $i$ 이후의 제품타입  $i$ 의 작업  $j_i$ 의 작업가중치

$\bar{p}$  : 전체 가공시간의 평균

$p_j$  : 제품타입  $i$ 인 작업  $j_i$ 의 가공시간

$d_j$  : 제품타입  $i$ 인 작업  $j_i$ 의 납기

$\bar{s}$  : 전체 작업준비시간의 평균

$s_{ij}$  : 작업  $i$ 이후 제품타입  $i$ 의 작업  $j_i$ 일 때 작업 준비시간

$f_{j_i}$  : 타입  $i$ 인 작업  $j_i$ 가 기계  $m$ 에서 갖는 정규품질계수

$Q_{jm}$  : 제품타입  $i$  인 작업  $j_i$  가 기계  $m$  에서 품질 인덱스  
 $\mu_{mi}$  : 기계  $m$  에서 작업타입  $i$  의 평균 품질  
 $\sigma_{mi}$  : 기계  $m$  에서 작업타입  $i$  의 품질 표준편차  
 $k_1, k_2$  : 조정모수( $k_1, k_2 > 0$ )

식(4)는 ATCSQ를 통해 제품타입  $i$  인 작업  $j_i$  가 기계  $m$  에서 계산된 가중치를 나타낸다. 식(5)은 계산된 품질 인덱스인  $Q_{jm}$  를 이용하여 품질제약조건을 만족여부를 결정한다. 그리고 품질인덱스를 정규화 하여 식(4)에 사용될 정규 품질계수를 계산한다. 식(6)은 공정능력지수를 이용하여 제품타입  $i$  인 작업  $j_i$  가 기계  $m$  에서 가공되었을 때의 품질 수준을 수치화 하여 품질인덱스를 산출한다.

ATCSQ를 계산하기위한 선행 작업으로 식(6)을 통해 작업들의 품질을 공정능력지수로 계산 하여 품질 인덱스를 구한다.

산출된 품질 인덱스는 식(5)를 이용하여 사전에 정의된 제품타입  $i$  의 품질제약 만족여부를 판단하는데 사용된다. 또한 식(5)는 작업의 품질인덱스가 최소 임계값이상인 작업들을 선별한 후, 임계값 이상의 작업들의 품질 인덱스를 이용하여 작업가중치를 계산하는 식(4)에 반영될 수 있도록 정규 품질계수로 변환한다. 식(5)와 식(6)에 의해 계산된 정규 품질계수는 ATCSQ의 각 항의 표현 범위와 동일한 0과 1사이의 값을 가지며 식(4)의 첫 항에 있는  $f_{j_i}$  에 품질차이로 반영된다.

식(4)의 첫 번째 항은 품질과 SPT(Shortest Process Time)규칙을 반영하여 가공시간이 적고 품질이 높으면 큰 값을 갖게 한다. 첫 번째 항의 정규품질계수  $f_{j_i}$  와 SPT 규칙인  $\frac{1}{p_{j_i}}$  는 0과 1사이의 값을 갖는다. 두 번째 항은 MS(Minimum Slack)규칙을 반영하여 납기까지의 시간적인 여유가 없다면 1에 근접한 값을 갖고, 납기까지의 여유가 많으면 0에 가까운 값을 갖는다. 세 번째 항은 SST(Shortest Setup Time)규칙을 표현하며 순서의존적인 작업준비시간이 작으면 1에 가까운 큰 값을 갖고 작업준비시간이 크다면 0에 가까운 값을 갖는다.  $k_1, k_2$  는 조정모수로서 두 번째 항과 세 번째 항이 식(4)의 인덱스 값에 미치는 영향을 조절해주는 가중치 역할을 한다.

ATCSQ는 공정능력지수를 이용하여 공정의 산포뿐만 아니라 평균의 위치까지 반영하여 품질 인덱스를 계산한다. 따라서 기계와 제품 타입에 따른 품질의 차이를 정확하게 반영한다. 품질 인덱스는 작업의 품질 중요도를 산출하여 품질을 반영한 작업 할당을 지시한다. 또한 품질

의 중요도, 납기, 순서의존적인 작업 준비시간을 고려하여 최적해에 근접한 해에서부터 개선을 시작 할 수 있도록 한다. 이상의 ATCSQ방법을 이용하여 품질과 납기를 동시에 고려하여 최적해로 개선할 수 있는 우수한 초기해를 생성한다.

## 2.4 해의 개선단계

초기해 생성단계에서 ATCSQ는 단순한 할당 규칙보다는 최적에 근사한 작업 할당을 수행한다. 그러나 ATCSQ로 계산된 가중치가 높은 작업을 우선순위에 따라 기계에 투입하는 것은 최적의 평균 납기지연시간을 보장하지 못한다는 한계를 갖는다.

더욱이 ATCSQ는 본 연구의 문제 상황과 같이 실시간으로 작업이 발생하여 빠른 시간 내에 작업을 할당해야 하는 경우에는 신속한 해의 개선을 요구한다. 따라서 해의 개선 단계에서는 초기해의 근사한 한계를 극복하고 신속하게 최적 해에 근접한 해에 도달하기 위한 초기해의 개선을 수행한다.

작업장에서 유희기계가 발생한 시점을  $t$  라고 할 때, 해의 개선은 시점  $t$  에서 한 step 증가하여  $t+1$  시점이 되기 전에 정해진 구간 안의 모든 작업들에 대한 가상 작업 계획(Dummy Scheduling)을 통해 수행된다.  $t$  시점에서 초기 해를 대상으로 하여 해의 개선을 수행한 후 생성된 해의 값들을 비교하여,  $t$  시점에서 품질을 만족하고 납기 지연을 최소화하는 최적의 작업할당을 수행한다.

$t$  시점에서 최적의 작업할당을 수행하는 과정을 마지막 구간까지 연쇄적으로 진행하고, 각 구간마다 최적의 작업 할당 조합하여 최종적으로 전체를 최적으로 하는 작업할당 지시가 가능해 진다

본 연구에서 제안하는 방법은 품질과 납기만족을 동시에 달성하기 위해 구간내의 모든 작업에 대해 각 기계에서 품질제약과 납기지연시간들을 비교하여 최적의 평균 납기지연시간을 찾는다.

### 2.4.1 Rolling Horizon Tabu Search 알고리즘

본 논문에서는 최적 해를 구하기 위한 이웃해 생성에서 탐색에 소요되는 계산시간을 단축시키기 위해 Rolling Horizon Tabu Search(RHTS)알고리즘을 제안하여 사용한다.

제안하는 RHTS는 발생된 전체 작업에 대하여 타부탐색을 수행하지 않고, 작업할당의 시작 시간부터 일정한 간격으로 전체 구간을 나누어 타부탐색을 수행한다.

초기 해를 개선할 때 일반적인 타부탐색과 같이 모든 작업을 타부탐색의 대상으로 포함하는 것은 비효율적이다. 예를 들어 각 기계의 작업순서상 선두에 위치해 있는

작업들은 대부분 납기가 긴급한 반면, 후미에 위치해 있는 작업들은 상대적으로 납기의 여유가 있는 작업들이다.

작업순서가 시간적으로 많은 차이를 보이는 모든 작업들을 동시에 타부탐색의 대상으로 한다면 많은 수의 이웃 해를 증가시킨다. 많은 수의 이웃 해는 탐색의 효율을 저하시켜 해의 개선에 많은 시간이 소요되도록 한다. 따라서 RHTS는 이동할 수 있는 작업의 수를 제한하여 탐색을 수행하므로 일반적인 타부탐색의 긴 시간이 드는 단점을 극복하여 신속하게 실시간적인 작업투입지시를 가능하게 한다.

RHTS는 품질인덱스와 납기지연시간을 비교척도로 사용하여 작업과 기계사이의 품질제약과 납기지연을 동시에 고려하여 목적함수인 평균 납기지연시간을 최소화 한다.

납기 지연시간은 유휴 기계들에서 순서 의존적인 작업 준비시간에 따라 기계별로 차이를 가질 수 있다. 따라서 선택된 중요 작업들에 대해 품질 제약을 만족하는 유휴 기계들의 납기지연시간과 품질을 계산한다. 계산된 값들을 비교하여 유휴기계들 중에서 최적의 납기지연시간을 갖는 기계를 찾아 이동 시키는 것이 해의 개선이다.

해의 개선은 각 작업과 유휴 기계에 따른 납기 지연시간과 품질 수준 값의 조합에 따라 두 종류가 존재한다. 첫 째로 초기해의 작업들은 기본적으로 품질 임계값 이상의 품질수준을 갖는 작업들이다. 따라서 해의 개선은 이동한 후 납기지연시간이 감소했을 때이다.

둘째로 초기 해를 이동한 후 납기지연시간이 같고 품질이 좋아졌다면 해의 개선이 이루어진 것이다. 이동을 했을 때 전체 평균 납기 지연시간은 변화가 없지만 평균 품질 값이 상승한다는 것은 본 연구의 목적 중에 하나인 품질의 최대화를 달성하는 것이 된다.

문제 상황에서 RHTS는 초기해의 첫 작업이 시작하는 시점( $t$ )에서부터 마지막 작업이 끝날 때까지 일정한 간격으로 전체 구간을 구분한다.

탐색의 영역을 일정한 구간으로 한정하여 시간에 따라 전진하며 RHTS를 수행한다. RHTS 알고리즘은 한 구간에서 탐색이 완료되어 작업 할당지시를 수행하면, 그 다음구간으로 이동하여 알고리즘을 수행한다.  $t$ 시점에서부터 시작하여 마지막 구간까지 RHTS를 수행하며 해를 개선한다.

해의 개선작업을 끝내면 RHTS는 전체 목적 식의 값을 비교하여  $t$ 시점에서의 최종 작업투입을 결정하게 된다. 최종 작업투입 결정은 RHTS가 수행되는 동안 구간내의 모든 해당 작업에 대해 품질인덱스와 작업의 납기지연시간을 유휴기계별로 비교하여 효과적으로 수행된다.

RHTS는 기본적으로 이웃해 탐색기법이며 최적 해를 구하기 위해 이동을 반복하며 해 영역을 탐색한다. 그리

고 국부 최적해(Local Optimal)에 머무는 것을 방지하고 좋은 해를 찾기 위해 목적함수 값을 저하 시켜도 이동을 허용한다(Aspiration). RHTS는 각 이동정보인 타부속성을 일정기간(Tabu Tenure)동안 타부목록(Tabu List)에 기억함으로써 탐색의 순환을 방지한다.(Glover 1989, 1990)

### 2.4.2 RHTS의 탐색영역 설정

문제 상황에서 할당 대상 작업들의 시작과 끝을 탐색 기준(Search line)으로 설정한다. 탐색기준은 첫 작업에서부터 마지막으로 할당될 작업의 완료시간( $C_{end}$ )까지 전체 구간을 말한다. 즉, 탐색기준은 초기 해에서 가장 먼저 할당될 작업인 기계  $m$ 의 첫 작업  $\pi(1, m)$ 의 위치에 해당하는  $t=1$ 과 마지막 작업이 완료되는 시점인  $t = C_{end}$ 까지 설정된다.

탐색 기준으로부터 탐색 깊이( $\tau$ : 분할한 각 구간의 크기)안의 작업들을 탐색영역(Target search scope)으로 설정한다. 현재해로부터 생성되는 이웃 해는 제한된  $\tau$  영역에서 탐색을 수행하여 구한다.  $\tau$ 는 생성되는 이웃해의 수에 직접적인 영향을 미치는 결정모수이다.  $\tau$ 는 이웃해의 생성을 일정수준 이하로 제약하여 탐색의 속도를 높인다.

### 2.4.3 이웃해 생성방법

RHTS는 작업을 이동 가능한 시점까지 삽입이동 한 후 재 스케줄을 통해 이웃해를 생성한다.

삽입이동은 한 작업을 선택하여 구간내의 다른 위치에 삽입하여 이웃 해를 생성해내는 방법이다. 이웃 해를 생성하기 위해서는 옮길 작업과 옮겨갈 위치를 결정해주어야 한다. 옮길 작업은 탐색구간 내의 이동조건에 부합하는 모든 작업이 해당되고, 옮겨갈 위치는 이동의 조건을 만족하고 타부조건에 해당되지 않는 모든 위치가 된다. 작업을 삽입이동하고 난후 이동된 작업의 가공시작시간 이후의 작업들에 대해 가중치를 계산하여 이웃해를 구한다. 이웃 해는  $\tau$ 구간 안에서  $L_{max}$  값을 갖는 기계의 작업을 선택하여 투입 가능한 다른 위치에 삽입 이동하여 생성한다.

삽입이동에 의해 생성된 전체 스케줄의 결과는 집합  $\Pi = \{\Pi_1, \Pi_2, \dots, \Pi_m\}$ 으로 표현하고, 다시 각 기계별 스케줄의결과는 집합  $\Pi_m = \{\pi(1, m), \pi(2, m), \dots, \pi(N, m)\}$ 으로 표현된다. 여기서  $\pi(i, m)$ 는  $m$ 번째 기계의  $i$ 번째 위치에 있는 작업이다.  $N$ 은  $m$ 번째 기계에 스케줄 되어 있는 작업의 수를 뜻한다.

RHTS 알고리즘은 현재해  $\Pi_{cur}$ 로부터 이웃해 집합을



생성하기 위해 각  $\tau$ 의 각 구간 내의  $L_{max}$  작업들을 대상으로 삽입이동을 수행한다.

RHTS는  $L_{max}$  작업을 선택하고 그때의 위치를  $L_{max}pos$ 라 설정한다. 그리고 선택된  $L_{max}$  작업을 투입 가능한 위치까지 삽입이동 시킨다.  $L_{max}$  작업을 삽입이동을 한 후, 새로운 위치에서  $L_{max}$  작업이 가공을 시작하게 되는 시점 이후의  $\tau$ 구간의 모든 작업들에 대해 식(4)의 ATCSQ인덱스 값과 식(5)와 식(6)을 이용한 품질인덱스 값을 계산하여 이웃 해집합  $N(\Pi)$ 를 구한다. 그리고 생성된 이웃 해에 대해서 목적함수 값을 계산하여, 최소값을 갖는 해를 현재 최우수해( $\Pi^*$ )로 설정한다.  $\Pi^*$ 와  $\Pi_{Best}$ 의 목적함수 값을 비교하여 작은 값을 갖고, 품질인덱스 값이  $Threshold_i$  이상이라면 현재 최우수해  $\Pi^*$ 를 최우수해  $\Pi_{Best}$ 로 설정한다. 만일  $L_{max}$  작업을 삽입 이동하였을 때, 계산된 목적함수 값이 같고 이동한 작업의 품질이 향상되었다면 현재 최우수해  $\Pi^*$ 를 최우수해  $\Pi_{Best}$ 로 변경한다. 즉, 삽입이동을 한 후에 평균납기 지연 시간은 변화가 없지만, 이동을 통해 평균 품질을 향상시킬 수 있다면 좋은 해가 되는 것이다. 그리고 타부조건에서 허용하는 시점까지 RHTS를 반복 수행하여 최우수해를 구한다.

삽입 이동 연산자는  $\Pi_{cur}$ 의  $L_{max}$  작업  $\pi(i, m)$ 을 새로운 작업순서인 같은 기계  $\pi(b, m)$ , ( $i \neq b$ ) 또는 다른 기계  $\pi(b, u)$ , ( $m \neq u$ )에 삽입한다. 삽입이동의 조건은 다음과 같다.

$$1. \text{Max}((C'_{\pi(i,m)} - d_{\pi(i,m)}), 0) \geq L(\Pi_{cur}) \quad (7)$$

$$2. Q'_{j_i} \leq \text{Threshold}_i \quad (8)$$

3. 이동시킬 작업  $i$ 는 현재  $\tau$ 구간 내의  $L_{max}$  작업

첫 번째 조건에서 식(7)는  $C'_{\pi(i,m)}$ 를  $L_{max}$  작업  $\pi(i, m)$ 가 삽입 이동 된 후의 작업 완료시간이라고 할 때, 삽입 이동이후의 납기 지연시간을 나타낸다. 생성될 이웃해의 작업  $\pi(i, m)$ 가 조건에 해당되면 이동을 허용하지 않는다. 즉,  $L_{max}$  작업  $\pi(i, m)$ 이 삽입 이동된 후의 새로운 납기 지연시간이 현재의 최대 납기 지연시간보다 크다면 작업  $\pi(i, m)$ 을 이동시킬 필요가 없다.

두 번째 조건에서 식(8)은 삽입이동 이후의 품질이 임계값 이하이면 작업의 이동을 허용하지 않는다. 타부목록에 의해 금지되지 않은 삽입 이동을 수행한 이

웃해들의 목적함수 값을 계산하고, 비교하여 가장 작은 값을 갖는 스케줄  $\Pi^*$ 을 선택한다.  $\Pi^*$ 와  $\Pi_{Best}$ 의 목적함수 값을 비교하여 값이 감소하지 않았다면, 이때의 이동을 타부목록에 저장한다. 만약  $\Pi^*$ 와  $\Pi_{Best}$ 의 목적함수 값이 같다면  $\Pi^*$ 와  $\Pi_{Best}$ 의 품질인덱스 값  $Q_{\Pi^*}$ 와  $Q_{\Pi_{Best}}$  값을 비교하여 그 값이 증가 하지 않았다면, 이때의 이동을 품질타부목록에 저장한다.

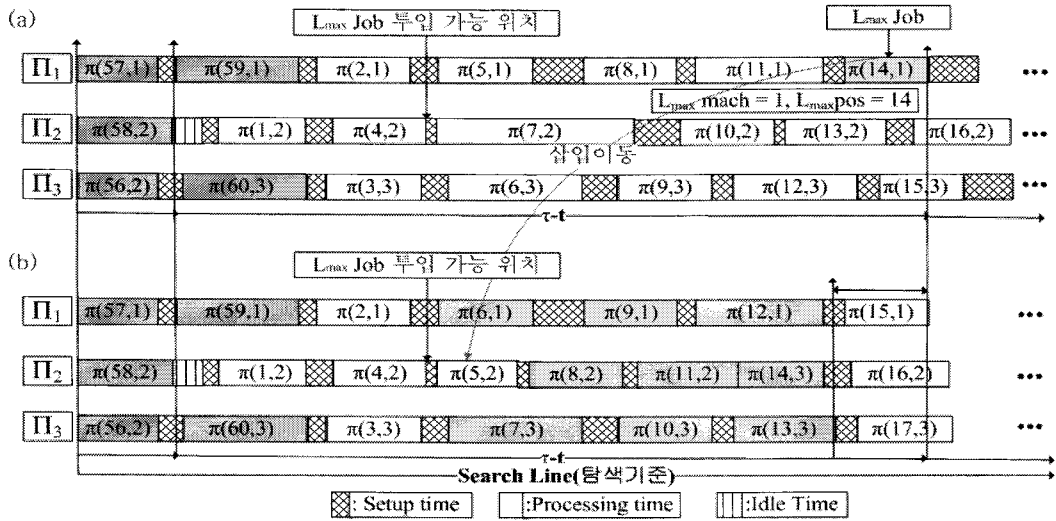
예를 들어 [그림 3]에서처럼 기계  $m=2$ 에서 작업 중인  $\pi(58, 2)$  작업이 완료되어 기계  $m=2$ 가 유휴 해지면 구간 내의  $L_{max}$  작업을 대상으로 삽입 이동이 시작된다. [그림 3(a)]와 같이  $L_{max}$  작업이  $\pi(14, 1)$ 이라고 하면  $L_{max}mach$ 는  $\Pi_1$ ,  $L_{max}pos$ 은 14가 된다. 이때 RHTS의 삽입이동 연산자는  $L_{max}$  작업인  $\pi(14, 1)$ 를 삽입이동시킬 수 있는 투입 가능한 지점인 작업  $\pi(7, 2)$ 의 시작위치로 이동하게 된다. [그림 3(b)]와 같이 이동된 작업과 그 작업의 투입시점 이전의 작업들을 제외한  $\tau$ 구간내의 모든 작업들에 대해서 식(4), 식(5), 식(6)을 이용한 인덱스 값을 계산하여 재 스케줄 한다. [그림 3(b)]와 같이 삽입 이동 후에 해집합  $N(\Pi^a)$ 의 목적함수 값이  $N(\Pi^b)$ 와 같이 감소하였다면 해의 개선이 이루어진 것이다. 만일 작업  $\pi(14, 1)$ 을 [그림 3(b)]의 투입 가능 위치로 이동하였을 때, 목적함수 값은 같지만 삽입이동 후의 작업의 품질이 향상되었다면 해의 개선이 이루어진 것이다. 해의 개선이 이루어진  $\Pi^b$ 를 최우수해  $\Pi_{Best}$ 로 치환하고 타부조건이 이동을 제한할 때까지 삽입이동을 반복한다.

RHTS는  $\Pi^*$ 에서 시작하여 사용자가 지정한 종료조건을 만족시킬 때 까지 현재  $\tau$ 구간 내에서 반복 수행된다.  $\tau$ 구간의 탐색이 완료되면 타부 목록을 초기화하고 다음 구간의 탐색을 시작한다. 모든 구간의 탐색이 완료되면 RHTS는 완료된다.

#### 2.4.4 Tabu 목록 관리

RHTS는 탐색으로 생성된 이동경로를 타부 목록에 일정기간 동안 기억하여 탐색을 수행할 때 지역 최적 해에 머무르는 것을 방지하여 다양한 탐색을 수행한다. 타부목록은 각 단계에서 이동에 제약을 주어 이동이 순환되는 것을 방지한다. 타부목록은 최근에 탐색했던 경로로 이동하는 것을 방지하기 위해 일정시간(tabu tenure)동안 목록을 유지한다.

타부탐색을 사용한 기존연구들에서 타부목록의 크기는 6과 10사이의 값을 사용했다.(Laguna etc 1991). 본 연구는 기존 연구와 실험 경험을 바탕으로 타부목록의 크



[그림 3] 탐색영역 범위 설정

기를 7로 고정하여 사용한다.

본 연구에서 사용하는 삽입이동은 품질과 납기를 동시에 고려하기 위해 두 가지의 속성을 타부 목록을 통해 관리된다. 우선 옮길 작업의 인덱스와 옮길 작업 앞의 인덱스를 타부목록에 기억해 둘 속성으로 정하였다. 즉, 최근의 이동에서 옮길 작업이  $\pi(i,m)$ 이었다면 작업  $\pi(i,m)$ 과 앞 작업의 인덱스들을  $\{\pi(i-1,m), \pi(i,m),\}$ 의 속성으로 기억한다. 이웃해 생성 시에 작업  $\pi(i,m)$ 가 작업순서상으로 작업  $\pi(i-1,m)$ 다음으로 되돌아오는 것을 금지한다. 만약 옮길 작업이 순서상 첫 번째 위치인  $\pi(1,m)$ 라면  $\{null, \pi(1,m)\}$ 의 속성이 타부목록에 저장된다.

본 연구에서 이웃 해를 생성하는 방법은 삽입이동을 수행한 이후의 대상 작업들에 대해 재 스케줄링을 하여 작업투입을 지시하는 것이다. 따라서 이동된 작업과 이동된 위치에서 바로 앞 작업과의 위치 관계를 타부목록으로 사용해야 한다.

그 다음으로 선택 작업이 이동 했을 때, 가공품질 값이  $Threshold_i$  이하 이면 이동할 필요가 없다. 또한 삽입이동 했을 때 같은 납기 지연시간을 갖지만, 이동 후 품질 값이 저하 된다면 이동할 필요가 없다. 따라서 옮겨갈 작업  $\pi(i,m)$ 의 품질수준인  $Q_{j,m}$ 과 새로운 위치로 옮겨 갔을 때의 가공품질 수준인  $Q'_{j,m}$ 을 타부목록에 기록해둘 속성으로 설정한다.

#### 2.4.5 열망수준(Aspiration Level)

열망수준이란 현재해  $\Pi_{cur}$ 로부터 삽입 이동을 통해 생

성된 이웃해  $\Pi^*$ 가 타부목록에 의해 금지된 해일지라도 더 좋은 해를 찾기 위해 이동을 계속 진행 할 수 있는 기준을 의미한다. 즉 열망수준은 좋은 해가 있는 영역으로의 움직임을 타부조건이 방해 할 때 타부조건을 일시적으로 무시하는 조건이다. 본 연구에서는 열망수준을 현 시점까지 구해진 최적해  $\Pi_{Best}$ 가 갖는 목적함수 값  $f(\Pi_{Best})$ 로 설정한다. 이웃해 중에서 타부목록에 의해 금지된 후보해가 있어도 목적함수 값  $f(\Pi_{Best})$ 을 최적화시킬 수 있는 이동이 가능 하다면 이웃해로 채택되도록 하였다.

#### [RHTS 알고리즘 절차]

##### Step 0 : 초기설정

유휴기계의 발생 시점  $t=1$ , 마지막작업의 작업 완료시점인  $C_{end} = l$ 을 탐색기준으로 설정한다.  $\tau$  및 탐색 반복횟수의 크기를 설정한다.

##### Step 1 : 초기해 설정

ATCSQ로 생성한 초기 해를 최우수해  $\Pi_{Best}$  (Best Solution), 그때의 목적함수의 값을 최우수 값  $f(\Pi_{Best})$  (Best Index)로 설정한다.

##### Step 2 : 자연 작업( $L_{max}$ ) 선택

현재 구간에서 최우수해의 성능 저하에 가장 많은 영향을 주는  $L_{max}$  작업을 선택한다. 그때의 기계

$L_{max}mach$ 와 위치  $L_{max}pos$ 의 값을 설정한다.

Step 3 : 이웃해 생성

$\tau$ 구간내의  $L_{max}$ 작업을 이동조건에 맞게 삽입 이동한다. 이동된  $L_{max}$ 작업과  $L_{max}$ 작업의 가공시작시점 이전에 가공 받게 될 작업들을 제외한 나머지 작업들에 대해서 ATCSQ규칙으로 가중치를 계산한다. 계산된 가중치를 이용하여 이웃해의 집합  $N(\Pi)$ 를 생성하고 생성된 이웃해들의 목적함수 값과 품질  $Q_{\Pi}$ 를 계산한다.

Step 4 : 목적함수 값을 비교

$N(\Pi)$ 중에서 목적함수 값이 가장 작은 이웃해  $\Pi^*$ 를 현재해  $\Pi_{cur}$ 로 바꾸고 최우수해  $\Pi_{Best}$  목적함수 값과 품질인덱스 값을 비교한다.

목적함수 값  $I(\Pi_{cur})$  값이  $I(\Pi_{Best})$ 보다 작다면,  $I(\Pi_{Best})$ 와  $\Pi_{Best}$ 를  $\Pi_{cur}$ 의 목적함수 값과  $\Pi_{cur}$ 로 치환한다. 만약 이동이 해를 개선하지 못한다면, 이때의 이동을 타부 목록에 저장한다.

만약  $I(\Pi_{cur})$  값과  $I(\Pi_{Best})$  값이 같을 때, 품질 인덱스 값인  $Q_{\Pi_{cur}}$  값과  $Q_{\Pi_{Best}}$  값을 비교하여  $Q_{\Pi_{cur}}$  값이 더 크다면  $I(\Pi_{Best})$ 와  $\Pi_{Best}$ 를  $\Pi_{cur}$ 의 인덱스 값과  $\Pi_{cur}$ 로 치환한다. 만약 이동이 해를 개선하지 못하거나 품질 값이  $Threshold_i$  값 이하로 변한다면, 이때의 이동을 타부 목록에 저장한다.

새롭게 변경된 현재해  $\Pi_{cur}$ 에서  $L_{max}$ 작업의 위치가 바뀌었다면,  $L_{max}mach$ 와  $L_{max}pos$ 의 값을 변경해 준다.

Step 5 :

RHTS의 탐색반복 횟수 값을 1증가 시키고 Step 2로 이동한다. 만약 증가시킨 탐색 반복횟수 값이 설정된 탐색반복 횟수 값 보다 크거나,  $\pi(i,m)$ 가  $L_{max}mach$ 의  $L_{max}pos$  값 보다 크거나 같으면 탐색 반복횟수 값을 초기화한다. 그리고  $\tau$ 구간을 증가시켜 Step 2로 이동한다. 이때 만약,  $\tau$ 값이 정해진 값 보다 크다면 탐색을 종료한다.

### 3. 실험 및 결과 분석

#### 3.1 실험 대안 설정

본 연구에서는 연속적으로 발생하는 작업들을 실시간으로 병렬기계에 할당할 때 품질만족과 납기 준수를 동

시에 고려하는 효율적인 작업투입 방법을 제안하였다. 본 연구에서 제안한 작업할당 방법에 대한 객관적인 성능 평가를 위한 실험비교대안으로는 기존의 작업투입 연구 중에서 품질과 납기를 동시에 고려하는 작업투입 대안을 대상으로 한다.

먼저 납기 관련대안으로 Lee(1997)의 ATCS를 사용한다. 본 연구의 초기해 생성방법으로 쓰이는 Ko(2008)의 ATCSQ도 ATCS를 개선시킨 모형이므로 ATCS가 적절한 비교대안이 된다.

지금까지의 연구에서는 품질위주의 작업투입을 지시하는 연구가 거의 없었다. 따라서 Ko(2008)의 연구에서 제시한 품질인덱스를 사용하여 품질 우선의 대안을 결정하는 방법을 품질 관련 비교대안으로 사용한다. 품질 관련 대안이란 Cpk를 이용한 품질 지표를 사용하여 작업을 투입할 때 가공품질이 최상인 기계에 우선 투입하는 방법을 대안으로 사용한다.

본 연구와 같이 품질과 납기를 동시에 고려하는 작업 투입 문제에 대한 연구로 Ko(2008)의 연구가 있다.

Ko(2008)의 연구는 품질제약달성과 납기준수를 위해 EATCSQ알고리즘을 사용하여 좋은 결과를 보였다. EATCSQ는 품질과 납기를 동시에 고려하여 작업을 선택하고, 또한 1차 국부 탐색을 통해서 해의 개선을 이루기 때문에 본 연구의 성능을 직접적으로 비교할 수 있는 좋은 대안이다.

본 연구에서 제안한 방법과 대안들의 성능평가는 납기 지연시간과 선택된 작업이 가공을 완료하고 얻은 품질 전체의 평균을 비교적으로 사용한다. 수행도 측정을 위해 목적함수인 평균 납기지연시간과 평균 대기시간, 재가공률(Rework) 및 품질을 측정하여 비교한다. 그리고 품질 제약 조건을 변경 시키며 각 평가 척도들의 변화를 살펴본다.

#### 3.2 실험 설계

본 연구는 품질과 납기 만족을 동시에 고려한 작업투입 방법의 효과를 검증하기 위해 다음과 같은 실험 환경을 설정한다. 대상공정은 단일 작업장에 10대의 기계와 10가지의 작업타입으로 구성된다.

기계들은 제품타입  $i$ 에 따라  $U[40 + 15i-5, 40 + 15i + 5]$ 의 균일분포를 따르는 상이한 가공시간의 분포를 가진다. 또한 작업준비시간은 이전 제품타입  $l$ 과 투입 제품타입  $i$ 에 따라 동일한 타입인 경우는 0으로 설정하고 다를 경우는 [표 2]와 같이 인덱스  $x_{il}$ 을 부여하여 각  $x_{il}$ 값에 따라  $U[15x_{il} - 5, 15x_{il} + 5]$ 인 균일분포를 따르는 상이한 작업 준비시간을 가지며,  $s_{ij} \neq s_{ji}$ 인 순서의존적인

준비시간으로 구성한다.

품질분포는  $N(\mu, \sigma^2)$ 인 정규분포를 따르며 Cpk값에 따라 10개의 기본 분포를 기준으로 제품타입과 기계별로 서로 다른 10가지의 분포로 세분 된다. 따라서 전체 품질 분포는 10×10매트릭스 형태로 구성된다.

본 연구의 문제 상황은 Rework이 존재하므로 Rework 여부를 판단하기위해 사용자가 정의해서 사용할 수 있는 품질규격한계를 사용한다.

Rework은 제품과 기계에 따른 가공품질수준의 분포를 통한 결과품질과 제품타입의 품질규격한계를 비교하여 결정된다. Rework은 제품과 기계에 따른 가공품질수준의 분포를 통한 결과품질과 제품타입의 품질규격한계를 비교하여 결정된다.

[표 2] 순서 의존적인 작업 준비시간을 위한  $x_{il}$  인덱스 행렬

$i \backslash l$	0	1	2	3	4	5	6	7	8	9	10
1	1	0	2	3	4	5	6	7	8	9	10
2	1	10	0	2	3	4	5	6	7	8	9
3	1	9	10	0	2	3	4	5	6	7	8
4	1	8	9	10	0	2	3	4	5	6	7
5	1	7	8	9	10	0	2	3	4	5	6
6	1	6	7	8	9	10	0	2	3	4	5
7	1	5	6	7	8	9	10	0	2	3	4
8	1	4	5	6	7	8	9	10	0	2	3
9	1	3	4	5	6	7	8	9	10	0	2
10	1	2	3	4	5	6	7	8	9	10	0

또한 Rework을 수행 할 때의 작업 이동시간과 재가공 초기화 시간은 제품의 타입에 따라 동일한 고정 값을 사용한다. 이상의 실험설계에서 정의한 실험 조건을 요약하면 [표 3]과 같다.

본 연구의 문제 상황에서 공정의 품질은 기계와 제품에 따라 경향이 다르게 나타날 수 있다. 따라서 공정의 품질을 상, 중, 하로 나누어 품질분포를 구분한 후, 품질 분포를 통해 생성된 데이터를 기준으로 대안들과의 비교 평가를 수행한다. 공정의 품질이 상인 경우 모든 기계들의 공정능력지수를 1이상으로 설정하고, 공정의 품질이 하인 경우는 1 이하로 설정한다. 공정의 품질이 중인 경우는 상과 하의 절충된 품질을 갖는다. [표 4]는 가공품질 수준 분포에 사용된 기본 분포를 공정능력지수와 관련된

여 나타내었다.

본 방법의 성능 비교는 각 공정품질제약을 변화시키며 생성된 결과 값 중에서 경향이 크게 다른 것들을 제외한 평균값을 사용하여 수행된다.

[표 3] 실험 데이터 생성기준

Experiment data List	Data values
Simulation Time	1
Number of Job types	10
Number of Machine in Workstation	10
Processing time of Job types	$U[40 + 15i - 5, 40 + 15i + 5]$
Setup time of Job types	$U[15x_{ii} - 5, 15x_{ii} + 5]$
Inter arrival distribution	$U[1, 150]$
Specification limit	$\pm 10$
Number of simulations Seed	100
Rework entering time ( Moving Time + Initialization Time)	5

[표 4] 각 상황에 따른 기본 품질 분포

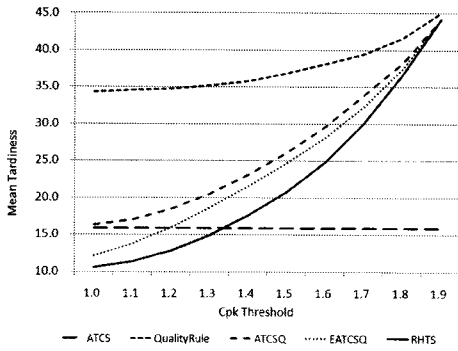
High Quality			Low Quality			Normal Quality		
$C_{pk}$	$\mu$	$\sigma$	$C_{pk}$	$\mu$	$\sigma$	$C_{pk}$	$\mu$	$\sigma$
2.0	0.5	1.6	1.0	0.5	3.2	2.0	0.5	1.6
1.9	1.0	1.6	0.9	1.0	3.3	1.8	1.0	1.7
1.8	1.5	1.6	0.8	1.5	3.5	1.6	1.5	1.8
1.7	2.0	1.6	0.7	2.0	3.8	1.4	2.0	1.9
1.6	2.5	1.6	0.6	2.5	4.2	1.2	2.5	2.1
1.5	3.0	1.6	0.5	3.0	4.7	1.0	3.0	2.3
1.4	3.5	1.5	0.4	3.5	5.4	0.8	3.5	2.7
1.3	4.0	1.5	0.3	4.0	6.7	0.6	4.0	3.3
1.2	4.5	1.5	0.2	4.5	9.2	0.4	4.5	4.6
1.1	5.0	1.5	0.1	5.0	16.7	0.2	5.0	8.3

### 3.3 실험 결과 및 분석

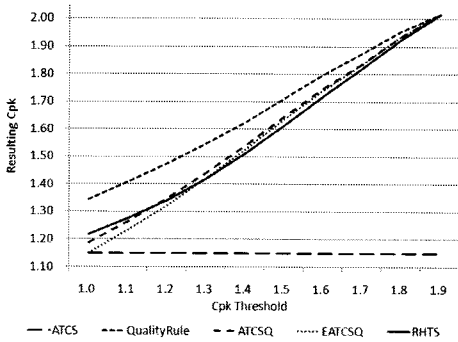
3.2에서 제시한 실험환경을 기반으로 본 연구에서 제안한 RHTS와 대안들과의 성능비교를 수행한 결과를 정리하면 다음과 같다.

[그림 4]와 [그림 5]는 공정내의 품질상황이 제품에 따라 최소  $C_{pk} \geq 1$ 인 가공품질을 갖는 기계들만 존재하는 높은 공정품질에 대한 실험 결과이다.

[그림 4]는 공정품질이 상일 때 품질제약이 증가하며 나타나는 평균 납기 지연시간의 변화를 표현하고, [그림 5]는 Cpk의 변화를 나타낸다. 공정품질을 Cpk로 평가할 때 Cpk가 1 이상이면 Rework Rate이 최대 0.1% 이하로 나타나며 모든 제품은 작업 완료 후에 양품으로 판정된다.



[그림 4] 높은 공정품질 상황에서의 평균 납기 지연시간



[그림 5] 높은 공정품질 상황에서의 평균 품질 값

[그림 4]와 [그림 5]에서 ATCS방법은 품질제약이 반영되지 않으므로 일정한 품질(Resulting Cpk = 1.07)과 일정한 평균 납기 지연시간(Mean Tardiness = 15.23)을 갖는다. Quality\_Rule은 가장 높은 품질 수준과 평균 납기 지연시간을 유지한다. ATCSQ는 ATCS, EATCSQ 및 RHTS 보다는 높고 Quality\_Rule 보다는 낮은 평균 납기 지연시간과 품질 수준을 갖는다. EATCSQ는 특정구간을 제외하고 Quality\_Rule, ATCSQ보다 낮은 평균 납기 지연시간을 갖고 ATCSQ와 Quality\_Rule보다 낮은 품질을 나타낸다.

RHTS는 대안들보다 근소한 차이로 낮은 품질을 보이지만 낮은 공정 품질 제약 상황을 제외한 모든 구간에서 품질 제약을 만족시키며 우수한 낮은 평균 납기 지연시간을 보인다.

Rework이 거의 없는 높은 공정품질 상황에서 낮은 품질 제약을 준다면, RHTS가 EATCSQ보다 약간 높은 평균 납기 지연시간을 갖는다. RHTS는 기계들의 가공 품질 수준이 우수하고 품질의 제약이 거의 없고 해를 개선과정에서 납기 지연시간의 여유가 있을 때, 품질을 고려하여 해를 개선하므로 좋은 품질을 발생시키는 기계로 작업을 투입하게 된다.

RHTS는 품질제약이 증가하면서 품질제약이 약 1.3이후가 되면 ATCS를 제외한 대안 알고리즘보다 낮은 품질을 갖지만 그 차이가 0.1 이하로 근소하다. 그리고 RHTS는 품질제약이 높을 때 제약을 높임에 따라 품질의 증가 폭이 둔화 되는 것을 보였다. 높은 수준의 품질 상황에서도 품질제약의 값을 계속 증가시키면 Rework, 대기작업의 증가로 인한 납기 지연이 증가하게 된다. RHTS의 전체 목적은 평균 납기 지연시간을 줄이는 것이다. 따라서 전체 가공 품질 수준이 높은 상황에서는 전체 최적 값을 찾기 위해 품질보다 납기준수를 많이 고려하는 것이다. 즉, 전체적인 품질이 높기 때문에 품질보다는 최소 납기 지연시간을 얻을 수 있는 이동을 허용하는 품질과 납기 지연시간의 상충 관계를 조절한 결과이다.

본 연구에서 제안한 RHTS는 대안 알고리즘들에 비하여 낮은 평균 납기 지연시간을 갖는다. 품질은 특정구간에서 Quality\_Rule, ATCSQ, EATCSQ 보다 낮지만 제약 조건을 만족시키는 것을 알 수 있다. Rework이 거의 없는 높은 공정품질 상황에서 RHTS가 ATCSQ보다 낮은 평균 납기 지연시간을 갖는다. ATCSQ가 병렬기계문제에서 품질을 고려하여 좋은 초기 해를 제시하지만 최적의 해는 아니기 때문에 해의 개선이 필요하다는 것을 보여준다.

RHTS를 통한 해의 개선은 품질제약이 증가함과 동시에 각 작업들의 납기 지연이 증가할 때 평균 납기 지연시간을 최소화 하는 것을 보여준다. 또한 기존의 연구와 달리 실시간으로 작업이 발생하는 상황에서는 RHTS를 이용한 해의 개선이 가장 좋은 평균 납기 지연시간을 나타낸다.

Quality\_rule은 품질만 고려하여 작업을 투입하기 때문에 납기에 따른 작업의 긴급 여부를 판단하지 못하여 평균 납기 지연시간이 높게 된다.

ATCSQ는 작업의 긴급 여부와 품질의 중요도를 동일하게 계산하기 때문에 목표하는 품질과 납기의 긴급 여부가 혼재할 경우 본 연구의 목적에 해당하는 최적 해를 제시할 수 없게 된다.

해의 개선을 수행하는 EATCSQ는 작업 완료 시간만을 기준으로 하여 1차 국부 탐색을 수행한다. EATCSQ는 ATCSQ 보다는 좋은 결과를 나타내지만 개선과정에서 납기 지연시간을 고려하지 않아 본 연구의 목적에 부합하는 최적 해를 제시하지 못한다. 따라서 본 연구에서 제시한

RHTS가 품질제약 조건과 목적을 만족하는 좋은 대안이라고 할 수 있다.

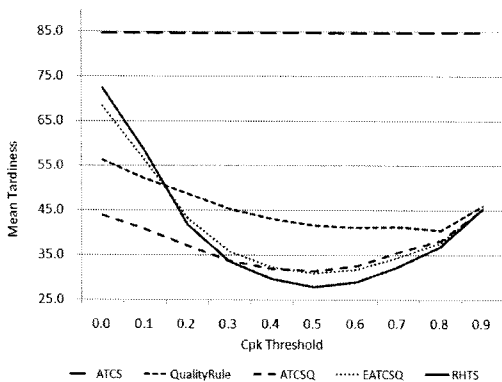
[그림 6]과 [그림 7]은 공정내의 품질상황이 제품에 따라 최대  $Cpk \leq 1$ 인 가공품질 결과를 가지는 기계들이 존재하는 낮은 공정품질에 대한 실험 결과이다.

공정품질을  $Cpk$ 로 평가할 때 1이하인 공정은 Rework Rate을 계산하면 평균 10%정도로 많은 Rework이 발생하는 공정이다. 따라서 작업투입 결정시 품질을 고려해야 하는 공정상황이다. 품질을 고려하지 않고 작업을 투입할 경우 많은 Rework이 발생하여 평균 납기지연시간이 길어지기 때문이다.

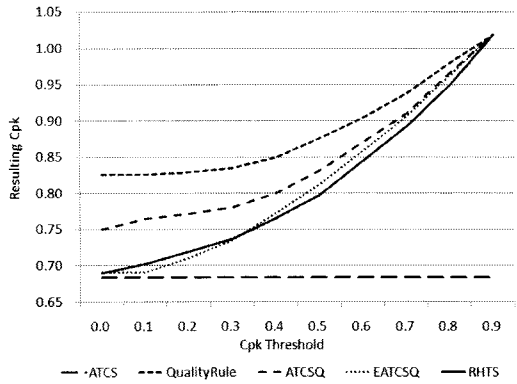
[그림 6]을 보면 ATCS가 평균납기 지연시간이 가장 높다. ATCS는 품질을 고려하지 않아서 많은 Rework을 발생시키고 납기지연을 증가시킨다. 따라서 작업투입에서 납기지연을 줄이기 위해 품질을 반영해야 함을 알 수 있다. 반면 Quality\_Rule은 낮은 공정품질 환경에서는 ATCS와 비교하여 낮은 납기지연시간을 갖는다. 높은 공정품질 상황에서는 납기에 대한 고려가 중요하지만 낮은 공정품질 환경에서는 품질고려를 통한 Rework 감소가 필요하기 때문이다.

그러나 동일 품질을 갖는 작업들 사이에는 납기에 대한 고려가 필요하므로 ATCS보다는 낮은 평균납기지연시간을 갖지만 ATCSQ보다는 그 값이 높다.

낮은 공정품질 상황에서 품질제약이 거의 없다면 ATCSQ는 품질에 대한 고려와 납기에 대한 고려를 동시에 수행하므로 Rework을 줄여 납기지연을 감소시키는 역할을 한다. 또한 품질은 Quality보다 낮지만 EATCSQ와 ATCS보다는 높은 결과를 보여준다. EATCSQ는 낮은 품질제약에서는 높은 평균 납기지연시간을 보이다가 품질제약이 증가하면서 Quality\_Rule, ATCSQ보다 낮은 평균 납기 지연시간을 갖는다.



[그림 6] 낮은 공정품질 상황에서의 평균 납기지연시간



[그림 7] 낮은 공정품질 상황에서의 평균 품질 값

EATCSQ는 낮은 품질에서 ATCSQ로 좋은 초기 해를 구했음에도 품질제약만 만족하면 작업완료시간을 고려하여 기계를 선택하는 탐색상의 문제로 평균 납기지연시간이 증가한다.

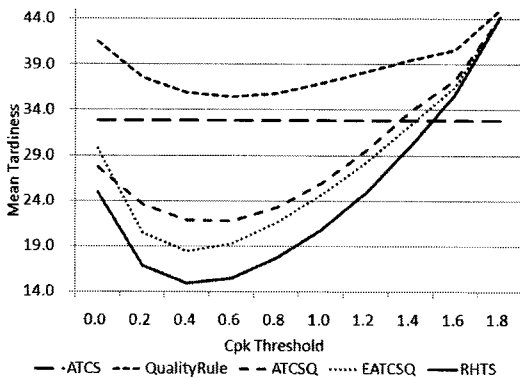
RHTS는 낮은 품질 제약에서는 높은 평균 납기지연시간을 보이다가 품질제약이 증가하면서 대안들보다 가장 낮은 평균 납기 지연시간을 가진다. 이것은 메타휴리스틱이 작업발생에 대한 정보가 없을 때에는 작업투입을 수행할 때는 개선에 많은 시간이 필요로 한다는 특징을 보여주는 것이다. 그러나 품질 제약이 증가하면서 Rework이 발생하고 납기지연이 증가하게 되면, 시간적인 약점을 극복한 RHTS가 Rework으로 인한 손실을 감소시키고 작업투입의 효율을 증가시켜 대안들보다 낮은 평균 납기 지연시간을 갖는다는 것이 확인된다.

또한 [그림 7]에서 보듯이 품질을 고려하지 않고 납기만을 고려한 ATCS는 평균품질이 0.66이다. 이것은 품질을 고려하여 납기지연시간을 최소화 하는 작업투입을 할 때 최소 품질을 0.66 이상으로 하여 제품을 생산할 수 있는 근거가 된다. [그림 6]과 [그림 7]에서 품질 제약이 0.66 이상인 모든 구간에서 RHTS는 품질제약을 만족하며 대안들보다 우수한 최소 납기 지연시간을 보인다. 따라서 적정 품질이상의 제품을 생산하기 위해 ATCS의 품질결과보다 높은  $Threshold_i$  값을 설정하면 RHTS가 품질제약을 충족하고 가장 낮은 평균 납기지연시간을 갖는 작업 투입 방법이 된다. 그리고 [그림 6]과 [그림 7]에서 보듯이 RHTS는 EATCSQ보다 더 효율적으로 해를 개선하기 때문에 요구 품질달성과 납기지연시간의 최소화를 신속하게 만족시키는 가장 좋은 대안이라고 할 수 있다. [그림 8]과 [그림 9]는 공정내의 품질이 일반적인 상황으로 제품에 따라 다양한 품질을 가지는 기계들이 존재하

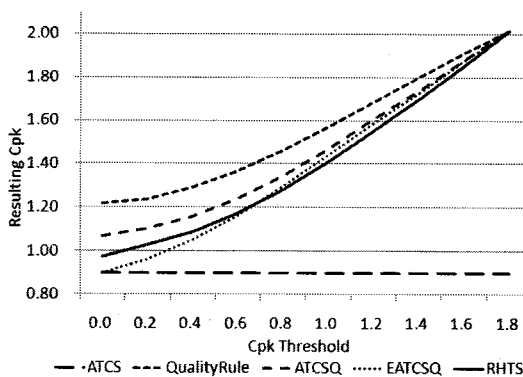
는 일반적인 공정품질의 결과이다.

공정내의 품질 조건이 일반적인 상황으로 다양한 영역의 가공수준을 갖는 기계가 존재한다. 일반적인 공정품질의 경우 본 연구에서 제시한 품질 조건에서는 평균 4.3%의 Rework이 발생한다. 따라서 납기 준수문제를 Rework의 영향을 줄일 수 있는 넓은 품질 제약구간에서 살펴볼 수 있다. [그림 8]과 [그림 9]에서 ATCS를 제외하면 평균 납기 지연시간은 초기에 조금 높다가 품질제약이 증가하면서 감소하는 모습을 보인다. 이것은 ATCS를 제외한 방법들이 품질을 고려하기 때문에 품질제약이 증가하면서 Rework이 줄어드는 결과이다. 하지만 약 0.66의 품질제약 이후 그 값이 증가하면서 평균 납기 지연시간이 증가하게 된다.

품질은 제약 조건에 따라 ATCS는 0.89의 일정한 값을 보이고, 전 영역에서 조건을 만족하며 Quality\_Rule, ATCSQ, RHTS, EATCSQ의 순서로 나타난다.



[그림 8] 일반적인 공정품질 상황에서의 평균 납기 지연시간



[그림 9] 일반적인 공정품질 상황에서의 평균 품질 값

일반적인 공정품질의 경우 본 실험에서 제시한 기계들의 다양한 품질 조건에서는 ATCS가 평균 납기 지연시간

이 가장 높게 나타난다. 그러나 다양한 가공품질수준이 존재하더라도 ATCSQ, EATCSQ 그리고 RHTS는 ATCS나 Quality\_Rule에 비해 평균 납기 지연시간 면에서 좋은 결과를 보인다.

특히 RHTS는 상당히 낮은 제약 상황이 아니라면 전반적으로 대안들보다 가장 낮은 평균 납기 지연시간을 보인다. 즉, 낮은 공정 품질상황에서 품질제약의 최소 기준 값으로 고려되었던 0.66값 이상에서는 RHTS가 품질제약을 만족하며 대안들에 비해 최소의 평균 납기 지연시간을 갖는다.

본 실험에서 다양한 공정 상황을 가정하여 각 대안을 살펴본 결과 높은 품질의 공정상황에서는 RHTS가 본 연구의 목적에 부합하는 최소 평균 납기 지연시간을 보였으며, 낮은 공정품질과 일반적인 공정품질 상황에서도 품질제약에 따라 RHTS가 좋은 성능을 보였다.

낮은 공정품질의 경우 품질을 고려하지 않은 ATCS는 Cpk 0.66수준을 갖는다. 따라서 대안들의 기준품질제약을 Cpk ≥ 0.66 이상으로 설정하여 성능을 비교하면, RHTS가 모든 공정품질환경에서 품질제약을 만족하며 대안들보다 최소 납기 지연시간을 가지며 좋은 성능을 보인다.

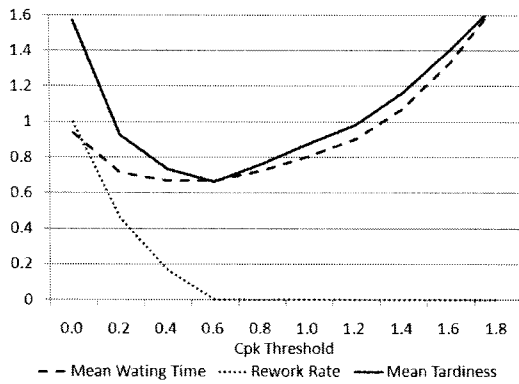
본 실험의 결과는 평균 납기 지연시간의 경우 높은 품질의 공정 상황을 제외하고 품질제약이 변함에 따라 감소하다 증가하는 모습을 보여주고 있다. 평균 납기 지연시간의 변화는 품질제약에 따라 납기에 영향을 주는 여러 요인이 복합적으로 작용한 것이다. 평균납기 지연시간에 영향을 주는 요인으로 평균대기 시간과 Rework Rate는 밀접한 관련을 갖는다.

[그림 10]과 [그림 11]은 RHTS에 대하여 낮은 공정품질 상황과 일반적인 공정품질 상황에서 품질제약에 따른 평균 대기시간과 Rework Rate 및 평균 납기 지연시간을 정규화 하여 표현하였다. [그림 10]과 [그림 11]에서 보듯이 평균 납기 지연시간은 감소하다가 증가하는 모습을 보인다. 평균 납기 지연시간의 감소와 유사한 형태로 평균 대기시간 또한 변한다. 반면 품질제약이 증가하면 작업의 Rework Rate은 줄어드는 것을 볼 수 있다. 또한 하나의 제품타입에 대해 서로 다른 분포를 갖는 기계들이 존재할 경우 품질제약을 수행하면 제약수준에 따라 가용한 기계의 수는 줄어들게 된다.

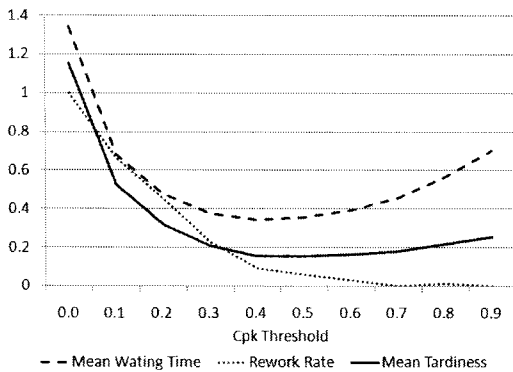
일반적으로 가용기계의 수가 줄어들면 작업들은 투입되지 못하고 대기하게 되어 평균대기 시간은 증가하게 된다. 그러나 [그림 10]과 [그림 11]에서처럼 품질제약에 따라 평균 대기시간은 감소하다가 늘어나는 것을 볼 수 있다.

[그림 10]과 [그림 11]에서 평균 대기시간이 감소하다가 증가하는 것은 Rework Rate의 영향이 반영된 것이다.

낮은 품질제약구간에서는 Rework Rate가 높아 많은 Rework이 발생한다. 따라서 최초 발생된 작업과 많은 Rework으로 인해 대기공간의 작업수가 증가하여 처리되는 작업에 비해 대기하는 작업이 늘어나 평균대기 시간이 증가하게 된다. 그러나 품질제약이 0.66이후 계속증가하면 RHTS 알고리즘에 의해 대기공간으로 돌아오는 Rework이 줄어들고 가용설비의 제약에 따른 처리능력저하보다 Rework Rate의 감소영향이 크게 작용하여 평균 대기시간이 줄어들게 된다.



[그림 10] 일반적인 공정품질 상황에서의 품질제약 증가에 따른 RHTS의 변동성



[그림 11] 낮은 공정품질 상황에서의 품질제약에 따른 RHTS의 품질의 변동성

하지만 품질제약이 계속적으로 증가하면 Rework Rate의 감소 영향보다 가용 설비 부족으로 처리능력감소의 영향이 커져 평균 대기시간은 증가하게 된다. [그림 10]와 [그림 11]에서 보듯이 적절한 제약을 수행하여 Rework Rate와 평균대기 시간의 변화를 조정하면 본 연

구에서 목적하는 평균 납기지연시간을 최소화 할 수 있다.

각 공정품질 환경에서 품질제약을 만족하고 평균납기지연시간을 최소화하는 본 연구의 방법과 대안들이 갖는 최선의 결과를 [표 4]에 표현하였다.

[표 4]는 모든 공정품질 상황에서 RHTS가 대안들과 비교하여 품질제약을 만족하며 평균 납기지연시간을 최소화 하는 것을 보인다. 평균 납기지연시간은 공정의 효율을 측정하는 지표이다. 따라서 좋은 성능을 보이는 RHTS가 효율적인 작업투입방법이 될 수 있다.

[표 4] 품질 제약을 만족하는 범위에서 각 대안들의 평균 납기지연시간의 최적 값

		Mean Tardiness	C <sub>pk</sub>	Threshold
Low Quality	ATCS	84.74	0.68	0.6
	QualityRule	41.07	0.90	0.6
	ATCSQ	31.41	0.83	0.5
	<b>RHTS</b>	<b>27.75</b>	<b>0.80</b>	<b>0.5</b>
Normal Quality	ATCS	32.81	0.89	0.6
	QualityRule	35.42	1.37	0.6
	ATCSQ	21.77	1.24	0.6
	<b>RHTS</b>	<b>14.91</b>	<b>1.09</b>	<b>0.4</b>
High Quality	ATCS	15.24	1.08	1.2
	QualityRule	32.94	1.44	1.2
	ATCSQ	17.65	1.29	1.2
	<b>RHTS</b>	<b>14.61</b>	<b>1.29</b>	<b>1.2</b>

RHTS는 품질제약조건을 만족하지만 품질우선 투입방법에 비해서는 조금 낮다. 납기와 품질을 동시에 만족시키는 것은 실험결과에서 나타나듯이 평균 대기시간과 Rework의 상충관계로 인해 불가피하다. 그러나 RHTS는 모든 구간에서 품질 제약을 만족하고, 그 값이 품질제약을 크게 상회한다. 그러므로 제한한 RHTS가 품질제약을 달성하며 평균 납기지연시간이 가장 우수함을 알 수 있다.

#### 4. 결론 및 추후연구

본 연구는 제품의 품질을 최대화 하고 동시에 납기지연시간을 최소화 하는 효율적인 실시간 작업투입 지시방법을 제시하였다. 또한 기존의 일반적인 타부탐색에 RHP를 적용한 RHTS를 이용하여 최적의 작업투입이 가능하도록 해를 개선하였다.



본 연구에서 제시한 작업할당 지시방법의 성능은 모의 실험을 통해 대안 알고리즘과 비교하였다. 실험 결과에서 본 방법은 품질제약을 만족시키며 평균 납기지연시간을 최소화하는 방법임을 확인하였다. 특히 본 연구의 실험결과에서 제시한 RHTS 알고리즘은 가공을 기다리는 작업들 간의 경쟁이 치열한 환경일수록 비교대안들에 비해 우수한 해를 제공한다는 것이 입증되었다.

문제 환경은 계획되지 않은 작업이 실시간으로 발생하는 상황이다. 본 연구의 작업투입 방법은 실시간으로 발생한 작업을 할당(Job Dispatching)할 때, 작업과 작업의 투입 사이에 가상스케줄(Dummy Schedule)을 통해 지역 최적 해를 구한 후 할당을 지시한다. 또한 지역 최적개선을 통해서 연속적으로 이루어지는 RHTS 알고리즘의 수행이 끝나면 전체 최적값에 도달 하게 되는 것이다.

일반적 메타 휴리스틱을 이용한 타부탐색은 작업의 발생률이 높아질수록 해의 영역이 늘어나 실시간적인 작업 투입에 어려움을 갖는다. 그러나 본 연구에서는 시간적인 약점을 극복하기위해 RHTS를 사용하였다. 또한 일반적인 제조공정의 가용기계들은 발생된 작업을 가공할 수 있는 적정능력(Capacity)을 갖고 있다. 그러므로 일반적으로 작업 대기공간에는 수용가능한 양의 작업들이 존재하므로 RHTS를 이용하여 짧은 시간 안에 평균 납기지연시간을 최소화하는 작업할당이 가능하다. 더욱이 컴퓨터 성능의 발달로 RHTS가 갖는 시간적인 약점은 대부분 극복되어 작업할당 순간에 최적의 작업투입계획을 수립할 수 있다. 따라서 본 연구에서 제안한 방법은 품질과 납기를 동시에 만족 시킬 수 있는 좋은 방법이 될 수 있다. 그러나 일반적인 공정 환경에서 모든 경우의 품질 수준을 알 수 없으므로 최적화된 품질제약의 임계값을 구할 수 없어 향후 이에 대한 연구가 필요하다.

### 참고문헌

- [1] 강용하, 김성식, 박종혁, 신현준, "Rework가 존재하는 이중병렬기계에서의 일정계획 수립", *Journal of the Korean Institute of Industrial Engineers*, 33 329-338, 2007.
- [2] 고효현, 김지현, 백준걸, 김성식, "품질을 고려한 작업투입에 관한 연구", *Journal of the Korean Institute of Industrial Engineers*, 34 108-121, 2008.
- [3] 고효현, 백종관, 강용하, 김성식, "반복 공정을 가지는 제약적 병렬기계에서의 일정 계획 수립", *Journal of the Korean Institute of Industrial Engineers*, 30 107-119, 2004.
- [4] Armentano V. A. and Yamashita, D. S., "Tabu search for scheduling on identical parallel machines to minimize mean tardiness", *Journal of Intelligent manufacturing*, 11(5). 453-460, 2000
- [5] Franca, P. M., Gendreau, M., Laporte, G. and Muller, F. M., "A tabu search heuristic for the multiprocessor scheduling problem with sequence dependent setup times", *International Journal of Production Economics*, 43(2-3), 79-89, 1996.
- [6] Glover, F., Tabu search-part 1, *ORSA Journal on Computing*, 1, 190-206, 1989.
- [7] Glover, F., Tabu search-part 2, *ORSA Journal on Computing*, 2, 4-32, 1990.
- [8] Laguna, M., Barnes, J. W. and Glover, F., "Tabu search methods for single machine scheduling problem", *Journal of Intelligent manufacturing*, 2, 63-74, 1991.
- [9] Lee, Y. H. and Pinedo, M., "Scheduling jobs on parallel machines with sequence-dependent setup times", *European Journal of Operations Research*, 100, 464-474, 1997.
- [10] Ovacik, I. M. and Uzsoy, R., "Rolling Horizon Procedures for dynamic parallel machine scheduling with sequence-dependent setup times", *International Journal of Production Research*, Vol.33, No.11, 31-37, 1995.
- [11] Pinedo M. and Chao X., *Operation Scheduling with Applications in Manufacturing and services*, 1999.
- [12] Pinedo M., "Scheduling Theory, Algorithms, and Systems" Second Edition, Prentice Hall, 2002
- [13] Schutten, J. M. and Leussink, R. A. M. (1996), "Parallel machine scheduling with release dates, due dates and family setup times, *International Journal of Production Economics*, 46, 119-126
- [14] Shin, H. J. and Kim, S. S. (2002) "A Restricted Neighborhood Generation Scheme for Parallel Machine Scheduling", *IE Interfaces* Vol. 15, No. 4, 338-348, 2002

윤 지 명(Ji-Myoung Yoon)

[준회원]



- 2007년 : 고려대학교 공과대학 산업시스템정보공학과 학사
- 2007년 ~ : 현재 고려대학교 정보경영공학 전문대학원 정보경영공학과 산업시스템 석사과정

<관심분야>

SCM, 생산관리, System Optimization

백 중 관(Jong-Kwan Baek)

[정회원]



- 1994년 : 고려대학교 공과대학 산업공학과 학사
- 1996년 : 고려대학교 일반대학원 산업공학과 석사
- 2002년 : 고려대학교 일반대학원 산업공학과 박사
- 현재 : 서일대학 산업시스템경영과 교수

<관심분야>

APC, Modeling, System Optimization

고 효 현(Hyo-Hyun Ko)

[정회원]



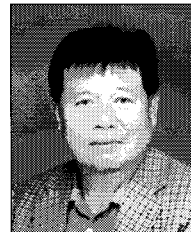
- 2002년 : 명지대학교 산업공학과 학사
- 2004년 : 고려대학교 공과대학 산업공학과 석사
- 현재 : 고려대학교 산업시스템정보공학과 박사과정

<관심분야>

MIS, 생산관리, APC

김 성 식(Sung-Shick Kim)

[정회원]



- 1972년 : 고려대학교 공과대학 기계공학과 학사
- 1974년 : 고려대학교 공과대학 산업공학과 석사
- 1976년: 미국 Southern Methodist University 산업공학과 석사
- 1979년: 미국 Southern Methodist University 산업공학과 박사
- 1979년 ~ : 현재 고려대학교 정보경영공학과 교수

<관심분야>

APC, Modeling, System Optimization