

# 이종 병렬설비 공정의 작업완료시간 최소화를 위한 유전 알고리즘\*

최유준\*\* · 송한식\*\* · †이익선\*\*

## A Genetic Algorithm for Minimizing Completion Time with Non-identical Parallel Machines

Yu Jun Choi\*\* · Han Sik Song\*\* · †Ik Sun Lee\*\*

### ■ Abstract ■

This paper considers a parallel-machine scheduling problem with dedicated and common processing machines. Non-identical setup and processing times are assumed for each machine. A genetic algorithm is proposed to minimize the makespan objective measure. In this paper, a lowerbound and some heuristic algorithms are derived and tested through computational experiments.

Keywords : Non-Identical Parallel Machine Scheduling, Makespan, Genetic Algorithm

## 1. 서 론

본 연구는 전용설비와 범용설비가 혼합되어 있는 제조공정에서 생산스케줄 수립과 그 실제 적용

을 위한 것으로 설비별로 작업에 따라 작업준비시간과 작업시간이 상이한 경우를 다루고 있다. 본 논문의 대상문제는 사례기업 면취기 공정(beveling process)의 생산스케줄 수립에서 그 실제 적용의

논문접수일 : 2013년 08월 27일    논문게재확정일 : 2013년 11월 07일

논문수정일 : 1차(2013년 10월 08일), 2차(2013년 11월 04일)

\* 이 논문은 동아대학교 학술연구비 지원에 의하여 연구되었음.

\*\* 동아대학교 경영대학 경영학과

† 교신저자, lis1007@dau.ac.kr

예를 발견하였다. 먼취기 공정은 강관(steel pipes)을 용접을 위해 개선각을 주거나 두 관(pipe)을 조립하기 위해 나선형 곡선으로 관의 양 끝을 그라인딩으로 갈아내는 작업이다. 사례기업에서 작업하는 강관류의 형상에 따라 각각 다른 전용설비에서 작업을 처리할 수 있으며, 철의 두께, 부피에 따라 전용설비가 처리할 수 있는 한계가 있으므로 한계를 벗어나는 제품은 범용설비를 통해 작업을 한다. 하지만 전·범용 설비에서는 철의 두께, 형상에 따라서 금형을 교체해야 함으로 작업준비시간이 필요하게 된다. 따라서 전·범용 설비에 어떠한 순서로 작업을 할당하고 작업량을 계획하는가에 따라 작업준비시간, 작업완료시간, 완제품 납기일에도 큰 영향을 미치기 때문에 주문의 납기를 지킬 수 있는 최적화된 일정계획을 수립해야 할 필요성을 확인하였다.

병렬설비의 일정계획에 대한 기존 연구들은 목적 함수에 따라 설비에 작업(제품)을 할당하여 총 작업완료시간(makespan)을 최소화하는 문제, 작업(제품)의 납기 지연 수를 최소화하는 문제, 하병현[2]의 납기에 대한 조기달성이나 납기 지연(earliness/tardiness)을 최소화하는 문제 등으로 분류할 수 있고, 병렬설비의 일정계획 문제에 대해서 많은 연구가 수행되어 왔다(Baker[4]).

본 연구에서는 사례기업 먼취기 공정상황을 토대로 기계 의존적 작업준비시간을 고려한 이중병렬기계의 작업완료시간 최소화 문제를 유전 알고리즘을 활용하여 최적의 일정을 계획하고자 한다. 먼취기 공정은 다수의 작업을 다수의 이중병렬기계(전용·범용 설비)에 할당하는 문제로서 생산현장에서는 작업(재공품)의 두께, 부피에 따라 전용설비의 전용성 한계가 존재하고, 이 정도를 넘어서는 작업은 범용설비를 통해 생산되어야 한다. 또한 전용설비와 범용설비에서는 작업의 두께, 형상에 따라서 금형 교체와 개선각 조절작업을 수행해야 함으로 작업준비시간이 발생된다. 이러한 작업준비시간, 가공시간(processing time)은 어느 기계에 할

당되는지에 따라 달라지며, 일반적으로 생산일정계획 분야에서의 작업순서 의존적(sequence dependent)과 기계 의존적(machine dependent) 작업준비시간 문제는 여러 공정에서 자주 발생되고, 작업일정에 따라 해당 설비의 효율성과 납기 준수율에 영향을 미치는 것으로 알려져 있다(Behnamian et al.[5]).

병렬설비의 작업완료시간 최소화 연구는 다음과 같다. McNaughton[24]은 선취(선매)방식과 독립적인 작업의 상황 하에서의 동종 병렬설비의 작업완료시간 최소화를 위한 알고리즘을 제안하였고, Hu[18]은 트리선행 제약조건과 동일한 작업처리시간의 작업들에 대하여 선취방식은 적용시키지 않은 작업완료시간 최소화하는 알고리즘을 개발하였다. 최유준[1]의 연구에서는 동종 병렬설비 상황에서 유사 작업을 그룹화하여 설비(열처리로)에 효율적이고 효과적으로 적재하는 방법을 유전 알고리즘을 활용하여 개발하였고, Guinet[14]의 연구에서는 순서 의존적 작업준비시간을 고려하여 동종 병렬설비의 작업완료시간 최소화와 작업 완료시간의 평균을 최소화 하는 알고리즘을 헝가리안 방법론을 확장하여 휴리스틱 알고리즘을 개발하였다. Piersma and Van Dijk[27]는 이중 병렬설비의 작업완료시간 최소화를 위한 일정계획 문제에서 기존의 지역탐색 알고리즘을 개선하여 탐색시간과 더 나은 해를 제공하는 새로운 지역탐색 알고리즘을 제안하였고, Min and Cheng[25]은 복잡도가 높은 문제의 동종 병렬설비 작업완료시간 최소화 문제를 해결하기 위해 유전 알고리즘을 제안하였다. Cheng et al.[7]은 작업의 크기가 각기 다른 작업을 배취 처리하여 동일한 병렬기계에 투입하여 작업완료시간과 총 작업완료시간을 최소화하는 일정계획을 근사 알고리즘을 활용하여 해결책을 제시하였다.

추가적으로 Ghirardi and Potts[13]은 이중 병렬설비의 작업완료시간 최소화를 위하여 RBS(Recovering Beam Search)를 활용한 휴리스틱 알고리즘을 제안하였다. Horowitz and Sahni[17], Ibarra and Kim[19], De and Morton[9], Davis and Jaffe

[8], Potts[28], Lenstra et al.[23], Hariri and Potts [15] 등의 연구에서는 작업완료시간 최소화를 위한 근사 알고리즘을 개발하였으며, Palekar et al.[26]은 근사 알고리즘을 위한 문제의 하한을 제안하였고, Van de Velde[32]은 분지한계 알고리즘과 휴리스틱 알고리즘을 제안하였고, Srivastava[31]는 문제의 해결방법으로 타부서치 알고리즘을 개발하였다.

Kim et al.[20]은 작업준비시간이 필요한 이중 병렬설비의 작업완료시간 최소화에 관한 연구로 설비의 능력이 상이하나 그 차이가 균등한 경우를 대상으로 연구하였고, Gendreau et al.[12], Kim et al.[21]은 동종 병렬설비의 순서 의존적 작업준비시

간을 대상으로 한 휴리스틱 알고리즘을 제안하였다. Franca et al.[10]은 총 작업완료시간을 최소화하기 위하여 타부서치 알고리즘을 제안하였고, Agarwal et al.[3]은 작업완료시간을 최소화 문제를 해결하기 위해 신경망 네트워크 모델을 활용한 휴리스틱 기법을 제안하였고, Chen and Chen[6]은 병목기반 휴리스틱을 활용하여 이중 병렬설비의 유형을 가지는 유연흐름 라인의 작업완료시간을 최소화하는 기법을 제안하였다.

병렬설비의 종류에 따라 연구를 간단히 분류하면 <표 1>과 같다.

본 연구에서는 보다 현실적인 상황을 다루기 위

<표 1> 기존 연구의 분류

병렬설비 종류	연구자	연구내용
동종 병렬설비	McNaughton[24]	<ul style="list-style-type: none"> <li>◦ 선매 선취방식 가능</li> <li>◦ 독립적 작업상황</li> <li>◦ 작업완료시간 최소화 알고리즘</li> </ul>
	Hu[18]	<ul style="list-style-type: none"> <li>◦ 트리선행 제약조건과 동일한 작업처리시간의 작업대상</li> <li>◦ 선취방식 미적용</li> <li>◦ 작업완료시간 최소화 알고리즘</li> </ul>
	최유준[1]	<ul style="list-style-type: none"> <li>◦ 유사 작업을 그룹화하여 동종설비 투입을 가정(batch problem+ bin packing problem)</li> <li>◦ 선취방식 미적용</li> <li>◦ 작업 완료시간 최소화를 위한 유전 알고리즘</li> </ul>
	Guinet[14]	<ul style="list-style-type: none"> <li>◦ 순서 의존적 작업준비시간을 고려</li> <li>◦ 작업완료시간 최소화, 작업완료시간의 평균 최소화 알고리즘(헝가리안 방법론을 확장한 휴리스틱 알고리즘)</li> </ul>
	Min and Cheng[25]	<ul style="list-style-type: none"> <li>◦ 복잡도가 높은 문제의 해결</li> <li>◦ 작업완료시간 최소화 위한 유전 알고리즘 개발</li> </ul>
	Cheng et al.[7]	<ul style="list-style-type: none"> <li>◦ 작업의 크기가 각기 다른 작업을 뺏취 처리</li> <li>◦ 총 작업완료시간을 최소화하는 근사 알고리즘</li> </ul>
	Gendreau et al.[12]	<ul style="list-style-type: none"> <li>◦ 순서 의존적 작업준비시간을 대상</li> <li>◦ 휴리스틱 알고리즘을 제안</li> </ul>
이중 병렬설비	Piersma and Van Dijk[27]	<ul style="list-style-type: none"> <li>◦ 작업완료시간 최소화 위한 지역탐색 알고리즘</li> <li>◦ 기존의 지역탐색 알고리즘을 개선(탐색시간, 더 나은 해를 제공하는 새로운 지역탐색 알고리즘 개발)</li> </ul>
	Ghirardi and Potts[14]	<ul style="list-style-type: none"> <li>◦ 작업완료시간 최소화 위한 RBS(Recovering Beam Search) 휴리스틱 알고리즘 개발</li> </ul>
	Kim et al.[20]	<ul style="list-style-type: none"> <li>◦ 작업준비시간 고려</li> <li>◦ 설비의 능력이 상이하나 차이가 균등한 경우를 대상</li> <li>◦ 작업완료시간 최소화</li> </ul>
	Chen and Chen[6]	<ul style="list-style-type: none"> <li>◦ 병목기반 휴리스틱을 활용</li> <li>◦ 유연흐름 라인의 작업완료시간 최소화 알고리즘 개발</li> </ul>

하여 설비의 성능이 각기 다르고, 이 설비들은 전용과 범용의 설비로 구분되며, 작업의 종류에 따라 지정된 전용설비에서만 작업할 수 있다. 또한 제품의 종류에 따라 각기 다른 전용설비에 투입될 수 있으며, 범용설비는 모든 제품을 작업할 수 있다. 이러한 이중 병렬설비의 일정계획 문제는 모든 병렬설비의 성능이 동일한 경우보다 더 자세하게 현실을 반영한 연구이며, 각 작업이 완료되면 작업교체나 예방정비를 위한 준비시간을 고려하여 작업 완료시간을 최소화 하는 일정계획을 유전 알고리즘을 활용하여 수립하고자 한다.

본 연구의 내용구성은 이와 같다. 제 2장에서는 본 논문에서 고려하고 있는 문제의 정의와 수리모형을 제시하고, 제 3장에서는 문제에 대한 분석과 적용을 통한 유전 알고리즘을 제안하였다. 제 4장에서는 유전알고리즘의 가능해와 성능을 분석하기 위한 하한값(lower bound)이 제시되고, 제 5장에서는 다양한 수치실험을 통하여 해법의 성능을 평가하였다. 마지막으로 제 6장에서는 연구의 결론에 대해 논하였다.

## 2. 문제정의

작업은 병렬로 구성된  $m$ 개의 전용·범용 설비에서 투입되고, 각 작업이 전용·범용에 투입됨에 따라 작업준비시간  $s$ 가 필요하다. 목적함수로써

총 작업완료시간(makespan)  $C_{max}$ 를 최소화하는 것을 목적으로 한다. 문제에서 고려되는 가정들은 이와 같다.

- (1) 투입된 작업은 주문량 완료하기 전까지 중단하거나, 다른 작업이 투입될 수 없다.
- (2) 각 작업은 하나의 설비에만 투입되어야 한다.
- (3) 각 작업은 전용설비와 범용설비의 작업시간, 작업준비시간이 상이하다.
- (4) 작업의 종류에 따라 각각의 전용 설비에만 투입 가능하고, 한계성을 넘어서는 작업은 범용의 설비에 투입되어야 한다(범용설비는 모든 종류의 작업이 가능).
- (5) 설비에 투입된 첫 작업은 작업준비시간을 고려하지 않는다.
- (6) 작업의 전체 완료시점은 설비의 최종작업완료시간 중 가장 긴 종료시점이다.

본 연구의 문제에 관한 이해를 돕기 위하여 연구의 목적에 따라 예제 문제를 활용하여 설명하도록 한다. <표 2>에서와 같이 먼취기 공정의 작업 정보가 있고, 설비의 처리시간이 각기 다른 전용·범용설비가 2대, 2대 있다고 가정한다.

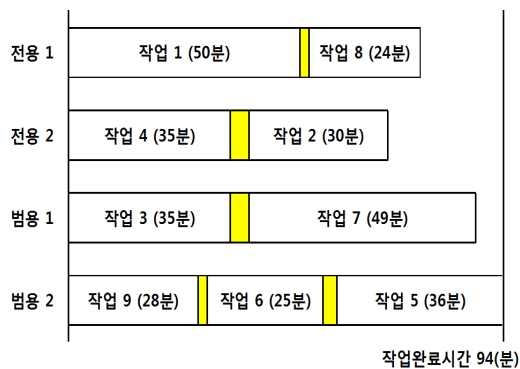
<그림 1>을 통하여 전용설비 2대, 범용설비 2대인 이중 병렬설비의 작업완료시간을 최소화하는 목적을 위한 일정계획 문제의 예제를 참조할 수 있다. 본 연구에서 사용하는 기호들은 다음과 같다.

<표 2> 먼취기 공정의 작업정보

		작업 1	작업 2	작업 3	작업 4	작업 5	작업 6	작업 7	작업 8	작업 9
전용	작업시간(분)	5	6	X	6	4	X	X	4	X
	작업준비 시간(분)	3	4	X	3	2	X	X	2	X
범용 1	작업시간(분)	6	5	5	6	5	3	5	6	5
	작업준비 시간(분)	4	3	5	4	3	2	4	3	5
범용 2	작업시간(분)	5	6	8	3	4	5	4	5	4
	작업준비 시간(분)	4	3	5	4	3	2	4	3	5
주문량		10	5	7	7	7	5	7	6	7
종류		1	2	1	2	2	1	1	1	2

- $j$  : 작업 index
- $i$  : 기계 index
- $a$  : 주문량 index
- $c_j$  : 작업  $j$ 의 작업완료시간
- $a_j$  : 작업  $j$ 의 주문량
- $p_{ij}$  : 작업  $j$ 가  $i$ 번째 설비에서 작업할 때 처리시간
- $s_{ij}$  : 작업  $j$ 가  $i$ 번째 설비에서 작업할 때 준비 시간

그림에서 “작업  $j(C_j)$ ”는 작업  $j$ 의 작업완료시간으로  $C_i$ 는 전용설비  $p_{ij}$ 의 처리시간과 주문량  $a_j$ 의 곱으로 작업시간을 계산할 수 있다. 즉, 전용설비 1의 경우에 첫 번째 “작업 1(50분)”은 작업 1이 설비 1에 투입되고,  $p_{11} = 5$ 이며,  $a_1 = 10$ 이므로 작업시간 50분으로 표현된다. 작업 1과 작업 8의 사이에 표시된 노란색은 작업 8의 작업준비시간  $s_{18} = 2$ 를 의미한다. 그러므로 전용설비 1의 작업완료시간은  $j_1$ 의 작업시간  $p_{11}$ 와 주문량  $a_1$ 로 계산한 50분,  $j_8$ 의 작업시간  $p_{18}$ 와 주문량  $a_8$ 로 계산한 24분과  $j_8$ 의 작업준비시간  $s_{18}$ 인 2분을 더한 것이  $b_1$ 의 작업완료시간 76분이 된다. 작업완료시간 최소화의 목적함수에서는 전체 작업의 완료시점은 설비의 최종 작업완료시간 중에서 가장 긴 종료시점이 되므로, 범용설비 2의 작업완료시간인 94분이 작업완료시간이 된다.



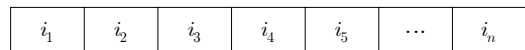
〈그림 1〉 작업완료시간 예제

### 3. 유전 알고리즘 적용

본 연구에서 제안하는 유전 알고리즘은 1859년 찰스다윈의 “종의 기원(the origin of species)”에서 종이 어떻게 변화하고 진화되는지를 반영한 메커니즘으로서 자연도태(natural selection)와 적자생존(survival of fittest)의 과정을 최적화 알고리즘에 적용한 것으로 진화과정을 모방하여 Holland[16]가 처음으로 유전 알고리즘의 개념, 절차, 적용을 소개하였고 유전계획(genetic plan)이라 정의하였다. 유전 알고리즘의 절차는 최유준 등[1]의 연구에서 문제의 상황에 따라서 수정하여 적용하였다.

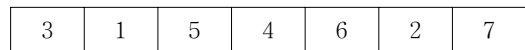
#### 3.1 개체의 표현

유전 알고리즘을 구축하는 첫 단계는 문제의 잠재해를 유전적으로 표현하는 것으로 염색체를 어떻게 표현하는가에 따라 알고리즘의 성능이 달라질 수 있다. 본 연구에서는 <그림 2>에서와 같이 염색체의 비트는  $i_1, i_2, i_3, \dots, i_n$ 이며 각 비트에 작업을 표현하게 된다.



〈그림 2〉 염색체 구조

<그림 3>은 염색체의 예제로 작업이 7개가 있다고 가정한 것이다. 작업 3이 먼취기 공정에 첫 번째로 할당되어지며, 작업 1은 두 번째로, 마지막으로 작업 7순으로 순차적으로 할당되어짐을 의미한다.



〈그림 3〉 염색체 예제

#### 3.2 초기 모집단

유전 알고리즘은 개체들로 구성된 모집단을 운영함으로 초기에 모집단을 생성해야 한다. 본 연구에서는 모집단을 100개로 구성하고, 임의 생성하도

록 하였다.

### 3.3 적합도 평가

적합도(fitness)는 개체의 생존능력을 의미하며, 유전 알고리즘에서는 적합도 함수는 선별과정에 사용된다. 환경에 대한 적응도(적합도)에 의해 현 세대의 모집단에서 다음 세대로 생존할 개체를 선택하는 과정으로 가능해의 평가와 산출에 중요한 역할을 한다. 본 연구에서 적합도 계산을 위해 목적함수( $C_{max}$ )를 사용한다.

### 3.4 선별기법

개체의 적응도에 기초하여 모집단에서 다음 세대로 생존 또는 진화에 필요한 개체를 선별하는 과정이다. 본 연구에서의 선별기법에는 확률적인 과정을 따르며, 일반적으로 보편적으로 널리 사용되는 기법인 roulette wheel selection을 사용하였다.

#### [선별기법 절차]

- step 1 : 모든 개체의 적합도를 계산한다.
- step 2 : 각 개체의 적합도를 합산하여 전체 룰렛의 범위를 결정한다.
- step 3 : 룰렛의 범위 내에서 난수를 발생시킨다.
- step 4 : 발생된 난수가 포함되는 개체를 부모개체로 선택한다.
- step 5 : 모든 부모개체를 선택하였으면 종료, 그렇지 않으면 step 3으로 이동한다.

이는 상대적으로 큰 적합도를 갖는 개체가 생존할 가능성을 높이는 것으로 특정한 염색체가 선택될 가능성이 적합성에 따름을 이용하였다. 계산되어진 적합도를 활용하여 룰렛의 범위를 정하고 난수를 발생시켜 개체의 선택하도록 하였다.

### 3.5 교차연산

유전연산에는 부모개체들의 좋은 유전인자를 자

손개체에 진화시키는 역할을 하는 단계로 roulette wheel selection을 통해 선택된 부모개체 한 쌍을 일련의 교차율에 따라 교차연산절차를 통하여 새로운 한 쌍을 생성하는 것이다. 본 연구에서 사용될 이점교차는 개체에서 두 개의 절단점을 임의로 선택하여 교차하는 것으로 두 절단점에 의해 앞, 중간, 뒤의 세 부분으로 나누어진다. 앞과 뒤 부분의 유전인자는 한 부모로부터 상속받고 중간부분의 유전인자는 상대 부모로부터 상속받도록 하였다.

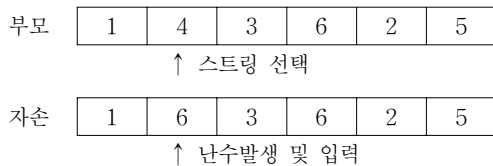
<그림 4> 교차연산의 예제와 같이 본 연구에서는 이점교차연산을 활용하였다. 교차지점을 선정하기 위해 개체의 염색체 수를 반으로 나누고, 난수를 발생시켜 절단점 1(교차지점 1)을 선정하며, 절단점 1의 대칭되는 위치를 절단점 2(교차지점 2)로 선택하였다. 절단지점이 선정되면 부모 1에서 자손 염색체로 유전 될 부분이 결정되며, 부모 1에서 결정된 유전인자(6과 5)는 부모 1에서의 위치와 같게 자손 1의 위치에 유전된다. 그리고 부모 2의 절단점 2 이후의 유전인자(3과 5)가 자손 1의 절단점 1 이전의 위치에 유전되며 부모 2의 절단점 1 이전의 유전인자(1과 6)이 자손 1의 절단점 2 이후의 위치에 유전되며, 같은 방법으로 자손 2 염색체를 생성하게 된다. 하지만 자손 1과 자손 2는 염색체에 동일한 인자가 중복적으로 구성되어 있으므로 실행 불가능 한 염색체로 판명된다. 이 경우 3.7의 교정연산을 통하여 실행가능 염색체로 유효하게 만들 수 있다. 두 염색체의 적합도의 계산을 통하여 모집단에서 가장 열등한 염색체와 비교하여 우월한 자손은 다음 세대로 진화되고, 그렇지 못한 자손개체는 제외시키도록 하였다.

부모 1	3	2	6	5	4	1
부모 2	1	6	4	2	3	5
	절단점 →					← 절단점
자손 1	3	5	6	5	1	6
자손 2	4	1	4	2	3	2

<그림 4> 교차연산의 예제

### 3.6 돌연변이 연산

돌연변이 연산은 염색체를 부분적으로 수정해서 새로운 염색체로 만드는 연산이다. 교차연산에서는 두 개의 부모염색체를 활용하여 새로운 한 쌍의 자손 개체를 생성한다면 돌연변이 연산은 교차연산에서 생성할 수 없는 염색체를 인위적으로 생성하는 연산이다. 본 연구에서 적용된 방법은 roulette wheel selection을 통해 개체를 선택하고, 선택된 개체의 스트링(칸) 수 내에서 난수를 발생시키도록 한다. 난수에 해당되는 비트의 값을 인위적으로 임의의 수로 변화를 주어 새로운 염색체를 만들어 내도록 하였다.



〈그림 5〉 돌연변이연산의 예제

〈그림 5〉는 돌연변이연산의 예제로 선택된 스트링에 난수를 발생하여 그 난수를 입력하여 새로운 염색체를 얻도록 하였다. 하지만 연산 이후 새로운 염색체가 유효하지 않은 염색체라면 3.7의 교정연산을 통하여 실행가능 염색체로 유효하게 만들고, 염색체의 적합도를 구하여 모집단의 열등한 염색체와 비교하여 우월한 자손개체는 다음 세대의 진화에 참여시키고, 그렇지 못한 자손개체는 제외시키도록 하였다.

### 3.7 교정연산

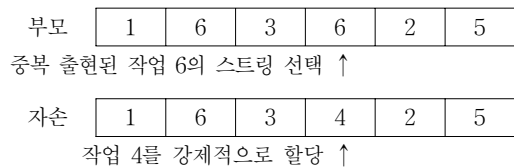
진화과정을 통해 생성되는 염색체의 일부는 연구의 기본적인 가정을 만족하지 못하는 경우가 발생된다. 초기 모집단 구성을 위한 임의생성, 교차, 돌연변이 연산 과정에서 염색체의 작업이 중복적으로 표현되는 경우로 표현되지 않는 작업은 설비에 할당되지 못하는 경우가 대표적인 예이다. 모든 작업이 염색체에 출현할 수 있도록 교정연산으로

조건을 만족하는 염색체로 변형한다.

[교정연산 절차]

- step 1 : 모든 작업의 할당이 완료되었다면 종료한다.
- step 2 : 염색체에 표현되지 않은 작업  $j$ 를 선택한다.
- step 3 : 중복 표현된 작업  $j'$ 를 선택한다.
- step 4 : 작업  $j'$ 를 삭제하고 그 자리에 작업  $j$ 를 할당한 후 step 1으로 이동한다.

위의 ‘교정연산 절차’는 모든 작업이 염색체에 출현할 수 있도록 강제로 작업을 할당하는 방안이다. 〈그림 6〉에서는 작업 4가 어느 염색체 비트에서도 할당되지 않았다. ‘교정연산 절차’를 통해 중복 표현된 작업 6에 강제로 작업 4를 할당하여 염색체를 유효하게 만들어 주었다.



〈그림 6〉 교정연산의 예제

### 3.8 설비할당 절차

염색체는 작업들이 투입되는 순서 정보에 해당된다. 이러한 정보를 바탕으로 전용·범용 설비에 적절히 작업 할당을 결정해야 한다. <표 2>의 전용설비의 작업시간  $X$ 의 의미는 전용설비의 전용성 한계로 인하여 범용설비에만 투입 가능한 작업을 의미하는 것으로 본 연구에서는 다음과 같은 작업 할당 절차를 제시하고자 한다.

이는 염색체의 작업순서 정보에 의거하여 범용설비에서만 작업 가능한 작업들을 우선적으로 할당하는 규칙으로 범용설비를 보다 효율적으로 사용하고자 하는 아이디어에서 출발하는 것으로 다음과 같은 세부절차를 따른다.

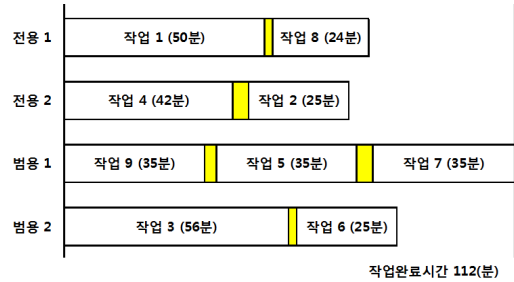
[설비할당 절차]

- step 1 : 전용설비에서 작업시간이 X로 표시된 작업을 선택한다.
- step 2 : 염색체에 표현된 순서대로 나열하고 선택한다.
- step 3 : 범용설비 작업완료시간이 가장 짧은 설비에 작업을 할당한다.
- step 4 : 모든 작업이 할당되었으면 step 5로 이동, 그렇지 않으면 step 2으로 이동한다.
- step 5 : 전용설비의 작업처리시간이 X로 표시된 작업의 염색체를 삭제한다.
- step 6 : 전용설비의 처리시간이 범용설비의 처리시간보다 작은 경우 염색체의 순서에 따라 작업을 전용설비에 할당한다. 그렇지 않다면 step 7로 이동한다.
- step 7 : 범용설비 작업완료시간이 가장 짧은 설비에 작업을 할당한다.
- step 8 : 모든 작업의 할당이 완료되면 종료한다, 그렇지 않으면 step 6으로 이동한다.

1	4	9	3	8	5	6	7	2
---	---	---	---	---	---	---	---	---

<그림 7> 설비할당 절차의 염색체 예제

<표 2>의 작업정보에서 염색체가 <그림 7>과 같다면 작업을 1-4-9-3-8-5-6-7-2의 순으로 설비에 할당하면 된다. 하지만 염색체의 순서로 설비에 할당하게 되면 범용설비의 작업완료시간이 <그림 8>과 같이 전용설비에 비해 상대적으로 크게 나오는 상황이 발생된다. 이러한 단점을 보완하기 위하여 “설비할당 절차”를 활용하여 작업을 할당하면 <그림 9>와 같다. 전용설비의 작업시간이 X로 표시되는 작업 9, 3, 6, 7을 순서대로 나열하고 이 작업들을 범용설비에 우선적으로 할당한다. 이때 설비할당의 규칙은 설비별 처리시간이 작은 곳에 할당하며 이후 1, 4, 8, 5, 2, 작업을 작업의 종류에 따른 전용설비와 범용설비에 각각 투입하게 된다.



<그림 8> 설비할당 절차를 사용하지 않은 경우



<그림 9> 설비할당 절차를 사용한 경우

### 4. 하한값(Lower Bound)

본 연구의 이중 병렬설비의 일정계획 문제는 동일한 성능을 가정한 병렬설비의 문제보다 복잡한 경우로 동종 병렬기계 일정계획 문제의 복잡도가 NP-hard 영역으로 증명된 Garey and Johnson[11], Lenstra and Rinnooy Kan[22], Rinnooy Kan[29], Sethi[30]의 연구보다 최적해를 찾기 어렵다. 그러므로 유전 알고리즘에서 생성된 가능해의 성능을 평가하기 위하여 본 연구에서 두 가지 하한값을 제시하고자 한다. 첫 번째 하한  $LB_1$ 은 다음과 같이 계산된다.

$$LB_1 = \sum_{j=1}^n \left\{ \frac{(mp_j \times a_j) + ms_j}{k+b} \right\}$$

$$mp_j = \min \{ p_{ij} v_{ij} > 0 \}$$

$$ms_j = \min \{ s_{ij} v_{ij} > 0 \}$$

$LB_1$ 은 투입될 작업  $j$ 의 전용설비 처리시간과



범용설비 처리시간 중에서 가장 작은 처리시간을  $mp_j$ 로 선정하며, 전용설비의 작업준비시간과 범용설비 작업준비시간 중 가장 작은 작업준비시간을  $ms_j$ 로 선정한다. 선정된 처리시간  $mp_j$ 는 주문량  $a_j$ 를 곱하여 작업준비시간을 더해주면 작업  $j$ 의 작업완료시간을 구할 수 있다. 이 작업완료시간은 제품의 종류에 따른 전용설비의 전체 수( $b$ )와 범용설비 수( $k$ )의 합으로 나누어 투입될 수 있음을 가정하였다. <표 2>의 작업정보가 주어진다면  $LB_1$  계산식에 대한 예제이다.

$$\begin{aligned} LB_1 &= \frac{\{(5 \times 10) + 3\} + \dots + \{(4 \times 7) + 5\}}{4} \\ &= \frac{53 + 28 + \dots + 26 + 33}{4} \\ &= \frac{283}{4} = 70.75 \end{aligned}$$

두 번째 하한값  $LB_2$ 는 다음과 같이 표현될 수 있다.

$$LB_2 = \sum_{j=1}^n \left\{ \frac{(mp_j \times a_j) + ms_j}{k+1} \right\} - hs(k+1)$$

$$hs = \max\{s_{ij} > 0\}$$

$hs$ 는 전용설비 작업준비시간과 범용설비 작업준비시간 중 가장 큰 작업준비시간을 의미하며,  $LB_2$ 는 작업  $j$ 가 할당 가능한 설비에서만 처리될 수 있고, 해당 설비별로 작업준비시간이 할당되어야 함을 고려한 하한값이다. 이는 첫 작업이 설비에 할당될 때 작업준비시간을 고려하지 않음으로 작업 준비시간 중 가장 큰 작업준비시간  $hs$ 를 전체 설비 대수로 곱하여 빼도록 하였다.  $LB_2$  계산의 예제는 다음을 참조할 수 있다.

$$\begin{aligned} LB_2 &= \left\{ \frac{(5 \times 10) + 3}{3} \right\} + \dots + \left\{ \frac{(4 \times 7) + 5}{3} \right\} - (5 \times 3) \\ &= 17.666 + \dots + 11 - 15 \\ &= 94.666 - 15 = 79.666 \end{aligned}$$

본 연구에서는  $LB_1$ ,  $LB_2$  중 보다 월등한 것을

최종 하한으로 정한다. 최종 하한은 다음과 같다.

$$\begin{aligned} LB &= \max\{LB_1, LB_2\} \\ &= \max\{70.75, 79.666\} \\ &= 79.666 \end{aligned}$$

## 5. 성능 평가

본 장에서는 작업완료시간 최소화 문제에 대하여 제 3장에서 제안된 유전 알고리즘의 성능을 평가하고 분석한다. 유전 알고리즘은 1000세대까지 진화시켰으며, 각 문제당 20가지의 문제를 임의 생성하였고, 10회에 걸쳐 유전 알고리즘으로 문제를 실험하였다. <표 3>은 작업과 설비의 유형, 각 설비 수의 변화에 따라 유전 알고리즘의 성능을 평가한다. 이 실험에서 작업의 수는 30, 50, 70개이고, 설비의 수는 전용설비 3~5대, 범용설비 3~5대이며, 작업  $j$ 의 설비별 작업처리시간을 20~50, 작업준비시간을 20~50, 주문량을 20~50사이에서 임의 생성하였으며, 전용설비에 투입될 수 없는 작업의 생성을 위한 비율은 전체작업의 20%로 가정하고 임의생성 하였다.

위의 문제의 상황을 가정하여 각 문제당 20가지의 문제를 임의생성하고, 각 문제당 10회에 걸쳐 유전 알고리즘의 성능을 실험하였다. 여기서 Gap1(%)은 유전 알고리즘의 가능해와 하한값의 오차를 의미하며, Gap2(%)는 LPT(Longest Processing Time)와 Gap3(%)는 SPT(Shortest Processing Time) 휴리스틱 방법의 해와 하한값의 오차를 의미한다. Average Gap(%)은 Gap(%)의 오차 평균값을 의미한다.

Gap1~3(%)은 다음의 식으로 계산할 수 있다.

$$Gap1(\%) = \frac{GA - LB}{LB} \times 100$$

$$Gap2(\%) = \frac{LPT - LB}{LB} \times 100$$

$$Gap3(\%) = \frac{SPT - LB}{LB} \times 100$$

유전 알고리즘의 가능해와 비교를 위한 LPT, SPT 휴리스틱 방법의 절차를 간단히 설명하면 다음과 같다. LPT 절차를 바탕으로 step 1에서 오름차순으로 정렬하여 설비에 할당하게 되면 SPT 설비할당 절차가 된다.

[LPT 절차]

step 1 : 전용설비와 범용설비의 작업시간 중 긴 작업시간을 선택하여 주문량과 곱하여

내림차순으로 정렬한다.

step 2 : 정렬된 순서대로 작업을 선택한다.

step 3 : 선택된 작업의 투입 가능한 설비의 작업 완료시간을 비교하여 가장 짧은 설비에 할당한다.

step 4 : 모든 작업이 할당되었으면 종료한다. 그렇지 않으면 step 2으로 이동한다.

<표 3>은 작업과 설비의 유형, 각 설비 수의 변

<표 3> 유전 알고리즘의 성능 실험

작업	설비		Average Gap1(%)	Average Gap2(%)	Average Gap3(%)	GA Time(초)
	전용	범용				
30	3	3	6.213	48.084	48.074	0.655
		4	7.919	37.180	47.214	0.735
		5	8.976	38.697	57.616	0.668
	4	3	7.855	31.521	37.299	0.726
		4	8.201	15.107	22.279	0.700
		5	8.395	20.623	37.483	0.690
	5	3	9.094	48.760	55.736	0.717
		4	9.235	26.691	38.949	0.712
		5	13.312	20.585	29.514	0.657
50	3	3	8.716	146.289	159.614	0.956
		4	12.590	121.299	128.236	0.959
		5	13.390	103.963	118.588	0.960
	4	3	8.956	56.008	71.974	1.286
		4	8.962	27.216	23.507	1.438
		5	9.354	29.823	21.108	1.442
	5	3	8.796	35.034	42.703	1.432
		4	13.170	29.678	31.908	1.430
		5	11.388	22.644	24.803	1.440
70	3	3	9.489	126.058	135.751	2.052
		4	12.810	96.517	102.911	2.059
		5	14.014	94.947	106.121	2.077
	4	3	11.981	52.789	66.140	2.648
		4	9.334	20.380	22.779	2.895
		5	9.967	21.837	17.071	2.902
	5	3	13.075	32.910	40.722	2.878
		4	10.982	23.091	19.621	2.882
		5	11.039	25.811	25.377	2.912

화에 따라 유전 알고리즘의 성능을 보여준다. Average Gap1(%)에서 작업의 수와 설비의 수가 증가할수록 이론적 하한값과의 차이가 최대 8% 이내로 비교적 안정적인 차이를 보였으며 실험에 소요되는 시간이 최소 0.6초에서 최대 2.9초로 증가되었다. 이는 임의생성한 문제의 특성에 따라 알고리즘의 성능이 차이는 있겠으나 제안한 유전 알고리즘이 문제의 복잡도가 증가할수록 해의 조기 수렴이나 지역해에 빠지지 않고, 유전연산을 효과적으로 수행하였으므로 재해석 할 수 있다. 또한 작업의 수가 30개, 전용설비가 3대, 범용설비가 3대인 가장 단순한 문제의 알고리즘 성능은 이론적 하한값

과의 차이가 6%이고, 전용설비 6대, 범용설비 6대인 경우의 이론적 하한값과의 차이는 20%로 약 14% 차이를 보인다. 이는 작업의 수에 비하여 설비의 수가 증가하는 경우에 유전 알고리즘의 성능이 저하되는 것으로 본 연구에서는 주문량을 분할하여 할당하지 못하지만 가능해와의 비교를 위한 이론적 하한값에서는 설비에 작업을 쪼개어 투입할 수 있으므로 이론적 하한값이 낮아지게 된다. 이로 인해 가능해와 이론적 하한값의 차이가 많이 발생되어 알고리즘의 성능이 저하되는 것으로 보이나 문제의 복잡도가 증가할수록 가능해와의 차이가 줄어드는 것을 확인할 수 있으므로 가능해가

〈표 4〉 유전 알고리즘의 성능 실험(작업준비시간 증가)

작업준비시간 범위	작업수	Average Gap(%)	GA Time(초)	작업준비시간 범위	작업수	Average Gap(%)	GA Time(초)
[20, 40]	30	12.167	0.602	[20, 50]	30	11.276	0.594
	40	11.233	0.876		40	10.422	0.871
	50	10.690	1.310		50	9.445	1.290
	60	13.034	1.853		60	8.873	1.851
	70	9.810	2.594		70	8.727	2.617
	80	10.894	3.540		80	8.521	3.537
[20, 60]	30	13.799	0.621	[40, 70]	30	14.638	0.593
	40	10.023	0.908		40	14.930	0.885
	50	10.526	1.278		50	15.386	1.337
	60	10.141	1.838		60	14.271	1.940
	70	10.423	2.564		70	14.467	2.610
	80	10.854	3.527		80	13.821	3.577
[20, 80]	30	10.624	0.629	[60, 90]	30	17.335	0.594
	40	10.308	0.892		40	18.955	0.866
	50	11.683	1.288		50	18.256	1.287
	60	10.332	1.853		60	17.721	1.859
	70	9.566	2.598		70	17.697	2.595
	80	8.294	3.615		80	19.653	3.542
[20, 100]	30	10.169	0.600	[80, 110]	30	20.529	0.590
	40	11.650	0.879		40	19.027	0.867
	50	10.220	1.297		50	20.964	1.276
	60	11.089	1.859		60	19.121	1.845
	70	9.138	2.604		70	21.818	2.583
	80	9.746	3.562		80	22.091	3.605

나쁘다고 판단할 수 없다.

Average Gap2(%)는 이론적 하한값과 LPT 휴리스틱 해와의 차이로 설비의 수가 작은 경우에 그리고 작업의 수가 증가될 때 그 차이가 최대 163% 이나 설비의 수가 증가되어 문제의 복잡도가 증가될수록 차이는 최소 15%까지 줄어드는 것을 확인할 수 있었다. 이는 복잡한 문제의 경우 간단한 휴리스틱 방법이라 하더라도 비교적 효율적인 해가 제공됨을 의미한다.

Average Gap3(%)는 이론적 하한값과 SPT 휴리스틱 해와의 차이로 설비의 수가 작은 경우에 그리고 작업의 수가 증가될 때 그 차이가 최대 170%

이나 설비의 수가 증가되어 문제의 복잡도가 증가될수록 차이는 최소 15%까지 줄어드는 것을 확인할 수 있었다. 이는 복잡한 문제의 경우 간단한 휴리스틱 방법이라 하더라도 비교적 효율적인 해가 제공됨을 의미한다.

<표 4>, <표 5>, <표 6>은 작업준비시간, 주문량, 작업처리시간 범위의 변동에 관한 실험으로 Gap(%)은 LB와의 비교를 나타낸다. <표 4>에서 최소 작업준비시간을 20으로 고정된 실험에서는 작업준비시간의 범위가 증가되더라도 알고리즘의 성능은 최소 8%에서 최대 13%로 비교적 안정적인 결과를 보여주며, 준비시간의 범위를 30으로 고정

<표 5> 유전 알고리즘의 성능 실험(주문량 증가)

주문량 범위	작업수	Average Gap(%)	GA Time(초)	주문량 범위	작업수	Average Gap(%)	GA Time(초)
[20, 40]	30	10.458	0.594	[20, 50]	30	13.823	0.599
	40	10.917	0.869		40	11.468	0.872
	50	10.024	1.286		50	12.614	1.291
	60	9.544	1.848		60	11.043	1.861
	70	10.133	2.574		70	12.250	2.598
	80	8.114	3.547		80	11.619	3.543
[20, 60]	30	13.100	0.593	[40, 70]	30	15.272	0.590
	40	11.325	0.868		40	15.263	0.864
	50	12.507	1.287		50	16.426	1.276
	60	11.262	1.847		60	16.297	1.841
	70	12.941	2.589		70	15.417	2.580
	80	13.062	3.535		80	16.380	3.587
[20, 80]	30	15.586	0.595	[60, 90]	30	18.642	0.596
	40	13.884	0.870		40	19.266	0.870
	50	15.900	1.285		50	16.998	1.285
	60	16.637	1.854		60	17.684	1.849
	70	15.430	2.585		70	18.801	2.605
	80	14.845	3.535		80	17.689	3.634
[20, 100]	30	16.748	0.596	[80, 110]	30	19.433	0.603
	40	15.989	0.875		40	19.058	0.881
	50	18.437	1.303		50	19.322	1.289
	60	17.488	1.900		60	19.016	1.857
	70	18.313	2.647		70	20.431	2.609
	80	18.361	3.542		80	18.869	3.552

시킨 실험에서는 작업준비시간이 커질수록 성능이 저하됨을 확인할 수 있다. 이는 작업을 처리하기 위한 시간보다 작업준비에 소요되는 시간이 더 긴 경우를 가정한 실험으로 작업준비시간이 큰 작업들을 처리하는 운영상황에서는 작업 처리시간을 고려하는 해결책 보다는 작업준비시간 최소화 관점에서 해결책을 모색할 필요가 있다.

<표 5>에서 주문량을 20으로 고정한 실험에서는 주문량의 범위가 증가함에 따라 알고리즘의 성능은 최소 8%에서 최대 18%이고, 주문량 범위를 30으로 고정시킨 실험에서는 최소 11%에서 최대 20%의 결과를 보여주고 있다. 공통적으로 작업의 수

가 증가할수록 알고리즘 성능이 좋아짐을 확인할 수 있고, 주문량이 많아지거나 그 범위가 클수록 알고리즘의 성능이 저하됨을 확인할 수 있다. 가능해는 각 작업의 주문량이 커지는 경우 작업이 할당된 설비에서 오랜 시간동안 작업을 처리해야 하고, 이론적 하한값은 긴 처리시간을 분할하여 각 설비에 투입시키기 때문에 가능해와의 차이가 많아져 알고리즘의 성능이 저하되는 것으로 판단된다.

<표 6>에서 작업처리시간의 범위를 20으로 고정한 실험에서는 알고리즘의 성능은 최소 8%에서 최대 22%이고, 작업처리시간의 범위를 30으로 고정한 실험에서는 최소 9%에서 최대 18%의 결과를

<표 6> 유전 알고리즘의 성능 실험(작업처리시간 증가)

작업처리시간 범위	작업수	Average Gap(%)	GA Time(초)	작업처리시간 범위	작업수	Average Gap(%)	GA Time(초)
[20, 40]	30	11.276	0.594	[20, 50]	30	13.088	0.596
	40	10.422	0.871		40	10.947	0.869
	50	9.445	1.290		50	12.552	1.288
	60	8.873	1.851		60	11.547	1.859
	70	8.727	2.617		70	10.809	2.593
	80	8.521	3.537		80	9.287	3.607
[20, 60]	30	14.638	0.593	[40, 70]	30	13.556	0.599
	40	14.930	0.885		40	11.515	0.876
	50	15.386	1.337		50	12.878	1.297
	60	14.271	1.940		60	11.942	1.872
	70	14.467	2.610		70	10.836	2.597
	80	13.821	3.577		80	11.941	3.552
[20, 80]	30	17.335	0.594	[60, 90]	30	14.395	0.594
	40	18.955	0.866		40	15.790	0.871
	50	18.256	1.287		50	13.711	1.294
	60	17.721	1.859		60	15.153	1.856
	70	17.697	2.595		70	13.806	2.594
	80	19.653	3.542		80	13.847	3.549
[20, 100]	30	20.529	0.590	[80, 110]	30	15.621	0.594
	40	19.027	0.867		40	18.312	0.878
	50	20.964	1.276		50	14.941	1.284
	60	19.121	1.845		60	12.642	1.849
	70	21.818	2.583		70	18.475	2.586
	80	22.091	3.605		80	16.235	3.529

보여주고 있다. 주문량 범위의 변동 실험에서와 같이 작업처리시간이 길어질수록 알고리즘의 성능이 저하됨을 확인할 수 있다. 가능해는 각 작업의 처리시간이 커지는 경우 작업이 할당된 설비에서 오랜 시간동안 작업을 처리를 해야 하고, 이론적 하한값은 긴 처리시간을 분할하여 각 설비에 투입시키기 때문에 가능해와의 차이가 많아져 알고리즘의 성능이 저하되는 것으로 판단된다.

<표 7>에서 작업의 수가 증가할수록 알고리즘의 성능이 저하되며, 전용설비 불가 작업의 비율이 0.5에서 가장 좋은 가능해를 제공해 주었다. 이는

<표 7> 유전 알고리즘의 성능 실험(전용설비 투입 불가 작업 비율)

전용설비 작업 불가 비율	작업수	Average Gap(%)	GA Time(초)
0.5	30	8.007	0.661
	40	8.085	0.966
	50	8.744	1.428
	60	8.098	2.068
	70	8.188	2.889
	80	8.206	3.938
0.6	30	8.288	0.661
	40	8.268	0.968
	50	9.228	1.430
	60	9.495	2.062
	70	9.203	2.883
	80	9.183	3.947
0.7	30	8.960	0.734
	40	9.239	0.998
	50	9.571	1.471
	60	9.149	2.097
	70	9.434	2.923
	80	9.887	4.027
0.8	30	8.395	0.690
	40	9.229	0.973
	50	9.354	1.442
	60	8.476	2.083
	70	9.967	2.902
	80	10.073	3.961

전용설비에 처리 불가능 작업이 총 작업의 50%가 되는 것으로 범용설비에서만 처리 가능한 작업이 전체 작업의 20%인 경우보다 비교적 각각의 설비에 균등하게 작업을 할당할 수 있으므로 전용설비의 처리시간과 범용설비의 처리시간의 차이가 줄어드는 것으로 판단된다.

<표 8>는 작업  $j$ 의 규모를 비현실적으로 크게 한 경우의 실험으로 주문량 20~50, 작업준비시간 20~50사이에서 임의 생성하였으며, 전용설비에 투입될 수 없는 작업의 생성을 위한 비율은 전체작업의 20%로 가정하고 임의생성 하였다. 작업의 수를 100, 200, 300, 400개, 설비의 대수는 전용설비 4대, 범용설비 5대인 경우를 대상으로 각 문제당 20가지의 문제를 임의생성하고 10회에 걸쳐 다양한 문제의 상황을 고려하여 유전 알고리즘의 성능을 실험하였다.

<표 8> 유전 알고리즘의 성능 실험(작업수 대규모 증가)

작업수	Average Gap1(%)	Average Gap2(%)	Average Gap3(%)	GA Time(초)
100	12.353	23.586	33.692	6.865
200	12.119	24.040	31.393	45.404
300	13.081	22.892	29.587	146.526
400	13.525	22.250	28.001	336.469

<표 8>에서 작업의 수가 증가함에 따라 알고리즘의 성능은 12%~13%의 결과를 보여주고 있다. 본 연구에서 제안하는 유전 알고리즘이 규모가 큰 문제에 대해서도 안정적인 가능해를 제공해 주는 것으로 판단되며, Average Gap2(%)는 최소 22%에서 최대 23%, Average Gap3(%)은 최소 28%에서 최대33%로 LPT, SPT보다 향상된 가능해를 제공하는 것으로 확인되었다. 하지만 작업의 수가 증가할수록 실험에 소요되는 시간이 증가됨을 확인할 수 있는데, 이는 유전 알고리즘의 염색체 비트가 작업의 수를 의미함으로 진화과정에 있어 염색체 해석과 유전연산에 많은 시간이 소요되어 실험 시간이 길어지는 것으로 해석할 수 있다.

## 6. 결 론

본 논문에서는 전용설비와 범용설비를 혼합하여 제품을 가공하는 사례기업에서 먼취기 공정상황을 토대로 작업준비시간을 고려한 이종병렬기계의 작업완료시간 최소화 문제를 유전 알고리즘을 활용하여 계획하였고, 알고리즘의 성능을 향상시키기 위하여 염색체의 해석을 위한 설비할당 절차와 해의 가정을 만족시킬 수 있는 교정절차를 제시하였다. 가능해의 검증을 위하여 다양한 휴리스틱 알고리즘과 비교를 통해 유전 알고리즘 가능해의 우수성을 분석하였다.

추후 실제 기업에서 발생하는 사례를 접목하는 연구와 다단계 작업장을 대상으로 이종 병렬설비의 연구를 확장할 필요가 있다. 또한 다른 메타 휴리스틱 기법과 다양한 유전 연산을 적용한 유전 알고리즘을 비교하여 평가하는 연구가 필요하다.

## 참 고 문 헌

- [1] 최유준, 유지돈, 송한식, 이익선, "병렬설비를 가지는 열처리공정에서의 배치일정계획 수립 연구", 『한국생산관리학회지』, 제22권, 제4호(2011), pp.451-467.
- [2] 하병현, "자원 제약이 없는 환경에서 부분 우선순위를 고려한 Earliness-Tardiness 최적 일정계획 알고리즘", 『한국경영과학회지』, 제38권, 제2호(2013), pp.141-157.
- [3] Agarwal, A., C. Colak, V. Jacob, and H. Pirkul, "Heuristics and augmented neural networks for task scheduling with non-identical machines," *European Journal of Operational Research*, Vol.175, No.1(2006), pp.296-317.
- [4] Baker, K.R., *Introduction to sequencing and scheduling*, New York : Wiley, Vol.31(1974).
- [5] Behnamian, J., M. Zandieh, and S.M.T. Fatiemi Ghomi, "Parallel-machine scheduling problems with sequence-dependent setup times using an ACO, SA and VNS hybrid algorithm," *Expert Systems with Applications*, Vol.36, No.6(2009), pp.9637-9644.
- [6] Chen, C.L. and C.L. Chen, "A bottleneck-based heuristic for minimizing makespan in a flexible flow line with unrelated parallel machines," *Computers and Operations Research*, Vol.36, No.11(2009), pp.3073-3081.
- [7] Cheng, B., S. Yang, X. Hu, and B. Chen, "Minimizing makespan and total completion time for parallel batch processing machines with non-identical job sizes," *Applied Mathematical Modelling*, Vol.36, No.7(2011), pp.3161-3167.
- [8] Davis, E. and J. M. Jaffe, "Algorithms for scheduling tasks on unrelated processors," *Journal of the ACM(JACM)*, Vol.28, No.4(1981), pp.721-736.
- [9] De, P. and T.E. Morton, "Scheduling to minimize makespan on unequal parallel processors," *Decision Sciences*, Vol.11, No.4(1980), pp.586-602.
- [10] Franca, P.M., M. Gendreau, G. Laporte, and F.M. Müller, "A tabu search heuristic for the multiprocessor scheduling problem with sequence dependent setup times," *International Journal of Production Economics*, Vol.43, No.2(1996), pp.79-89.
- [11] Garey, M.R. and D.S. Johnson, *Computer and intractability : A guide to the theory of NP-completeness*, San Francisco : W.H. Freeman, 1979.
- [12] Gendreau, M., G. Laporte, and E.M. Guimarães, "A divide and merge heuristic for the multiprocessor scheduling problem with sequence dependent setup times," *European*

- Journal of Operational Research*, Vol.133, No.1(2001), pp.183-189.
- [13] Ghirardi, M. and C.N. Potts, "Makespan minimization for scheduling unrelated parallel machines : A recovering beam search approach," *European Journal of Operational Research*, Vol.165, No.2(2005), pp.457-467.
- [14] Guinet, A., "Scheduling sequence-dependent jobs on identical parallel machines to minimize completion time criteria," *The International Journal of Production Research*, Vol.31, No.7(1993), pp.1579-1594.
- [15] Hariri, A.M.A. and C.N. Potts, "Heuristics for scheduling unrelated parallel machines," *Computers and operations research*, Vol.18, No.3(1991), pp.323-331.
- [16] Holland, J.H., *Adaptation in Natural and Artificial System*, University of Michigan Press, Ann Arbor, 1975.
- [17] Horowitz, E. and S. Sahni, "Exact and approximate algorithms for scheduling non-identical processors," *Journal of the ACM (JACM)*, Vol.23, No.2(1976), pp.317-327.
- [18] Hu, T.C., "Parallel sequencing and assembly line problems," *Operations Research*, Vol.9 (1961), pp.841-848.
- [19] Ibarra, O.H. and C.E. Kim, "Heuristic algorithms for scheduling independent tasks on nonidentical processors," *Journal of the ACM(JACM)*, Vol.24, No.2(1977), pp.280-289.
- [20] Kim, D.W., K.H. Kim, W. Jang, and C.F. Frank, "Unrelated parallel machine scheduling with setup times using simulated annealing," *Robotics and Computer-Integrated Manufacturing*, Vol.18, No.3(2002), pp.223-231.
- [21] Kim, J.G., J.S. Kim, and D.H. Lee, "Common Due-Date Assignment and Scheduling with Sequence-Dependent Setup Times : a Case Study on a Paper Remanufacturing System," *Management Science and Financial Engineering*, Vol.18, No.1(2012), pp.1-12.
- [22] Lenstra, J.K. and A.H.G. Rinnooy Kan, "Complexity of scheduling under precedence constraints," *Operations Research*, Vol.26 (1978), pp.22-35.
- [23] Lenstra, J.K., D.B. Shmoys, and É. Tardos, "Approximation algorithms for scheduling unrelated parallel machines," *Mathematical programming*, Vol.46, No.1(1990), pp.259-271.
- [24] McNaughton, R., "Scheduling with deadlines and loss functions," *Management Science*, Vol.6(1959), pp.1-12.
- [25] Min, L. and W. Cheng, "A genetic algorithm for minimizing the makespan in the case of scheduling identical parallel machines," *Artificial Intelligence in Engineering*, Vol.13 (1999), pp.399-403.
- [26] Palekar, U.S., N. Rama, and K. Toaffe, "Duality based relaxations for makespan minimization for unrelated parallel machines," *TIMS/ORSA Bulletin*, Vol.31(1991) p.21.
- [27] Piersma, N. and W. van Dijk, "A local search heuristic for unrelated parallel machine scheduling with efficient neighborhood search," *Mathematical and Computer Modeling*, Vol.24, No.9(1996), pp.11-19.
- [28] Potts, C.N., "Analysis of a linear programming heuristic for scheduling unrelated parallel machines," *Discrete Applied Mathematics*, Vol.10, No.2(1985), pp.155-164.
- [29] Rinnooy Kan, A.H.G., *Machine scheduling problems : Classification, complexity and computations*. The Hague : Martinus Nijhoff, 1976.
- [30] Sethi, R., "On the complexity of mean flow



- time scheduling," *Mathematics of Operations Research*, Vol.2, No.4(1977), pp.320-330.
- [31] Srivastava, B., "An effective heuristic for minimising makespan on unrelated parallel machines," *Journal of the Operational Research Society*, Vol.49, No.8(1998), pp.886-894.
- [32] Van de Velde, S.L., "Duality-based algorithms for scheduling unrelated parallel machines," *ORSA Journal on Computing*, Vol.5, No.2(1993), pp.192-205.