

Single Machine Scheduling Problem with Step-deterioration under A Rate-modifying Activity

Byung Soo Kim[†]

Department of Industrial and Management, Incheon National University

단일 복구조정활동 하에 단계적 퇴화를 가지는 단일기계 생산일정계획

김 병 수[†]

인천대학교 산업경영공학과

In this paper, we deal with a single machine scheduling problems integrating with step deterioration effect and a rate-modifying activity (RMA). The scheduling problem assumes that the machine may have a single RMA and each job has the processing time of a job with deterioration is a step function of the gap between recent RMA and starting time of the job and a deteriorating date that is individual to all jobs. Based on the two scheduling phenomena, we simultaneously determine the schedule of step deteriorating jobs and the position of the RMA to minimize the makespan. To solve the problem, we propose a hybrid typed genetic algorithm compared with conventional GAs.

Keywords : Scheduling, Step-Deterioration, RMA, Genetic Algorithm

1. 서론

생산일정계획은 공급사슬망 내의 생산시스템의 효율적 관리를 통해 성능향상에 상당한 영향을 미친다. 지금까지 상당히 많은 연구들이 전통적인 일정계획으로 연구되어 왔다. 이들 연구들은 대표적으로 작업에 따라 그 처리시간이 그 작업을 처리하는 자원(Resource)의 상황에 관계 없이 항상 일정하다는 점이다. 그러나 현실 상황에서의 작업처리시간은 기계의 가공시간이 경과함에 따라 작업물의 위치이탈, 공구의 마모, 혹은 가공과정에서의 찌꺼기 발생 등에 의해 작업처리시간이 점차 증가하게 된다. 이처럼 기계의 가공시간이 경과함에 따라 작업처리시간이 선형 혹은 단계적으로 증가하는 상황을 고려한 문제를 작업 처리시간의 퇴화(Deterioration)를 고려한 생산일정계획

문제라고 한다. 이러한 기계의 퇴화는 정비(Maintenance) 및 청소(Cleaning) 작업등을 통해 작업처리시간을 원상태로 회복시켜줄 수 있다. 기계와 같은 자원의 작업 처리시간을 원상복귀 또는 향상시키는 행위를 복구조정 활동(Rate Modifying Activity : RMA)라고 한다[21].

퇴화를 고려한 생산일정계획에 관한 최초의 연구로는 Gupta and Gupta[11]와 Browne and Yechiali[5]를 들 수 있다. 이들의 연구 이후 지금까지 관련 연구들이 지속적으로 증가하고 있다[2, 6]. Alidaee and Womer[1]는 작업처리시간의 증가(퇴화)를 작업시작시간의 함수로 가정하고 그 함수를 선형, 단계적 선형, 그리고 비선형으로 분류하여 관련 연구들을 조사하였다. 그들이 조사한 퇴화할 수 유형들은 이후의 퇴화를 고려한 대부분의 단일기계 일정계획에 관한 관련 연구들의 퇴화함수의 형태로 사용되어 왔다. Bachman et al.[2]은 작업처리시간이 작업시작시간 비례하여 선형으로 증가(퇴화)하는 경우, 총 가공

작업완료시간을 최소화 하는 문제의 NP-hardness를 증명하였다. Cheng and Ding[6]은 작업처리속도의 퇴화를 고려한 최대작업시간(Makespan)을 최소화하는 단일기계 일정계획문제의 해결을 위한 동적계획법을 제시하였다. 최근에 Wang et al.[30]은 작업시작시간 중속적인 퇴화를 고려한 총 작업완료시간을 최소화하는 단일기계 일정계획문제에 대해 V형태 최적해 특성[25]을 이용한 두 가지의 휴리스틱을 제안하고 그 성능을 SPT(Shortest processing time) 법칙을 이용한 지역해 탐색 휴리스틱과 비교하였다. 이외에도 최근에는 다수의 연구자들이 작업시작 시간에 비례하는 퇴화를 고려한 연구를 활발히 수행하고 있다[3, 4, 9, 12, 14, 28]. 지금까지의 연구들은 퇴화의 형태가 작업시작시간에 비례하는 형태들이다. 하지만, 많은 생산 및 서비스의 환경이 작업시간이 구분 퇴화 함수(piecewise-deteriorating function) 형태로 증가하고 있다. 따라서, 다수의 연구자들은 현장의 환경을 고려한 단계적 퇴화(step-deteriorating)를 고려한 생산일정계획을 수행하였다. Kunnathur and Gupta[18]는 최초로 작업시간이 두 단계로 퇴화하는 선형함수인 상황에서의 생산일정계획을 수행하였다. Sundararaghavan and Kunnathur[29]는 작업시간이 단계적 퇴화함수인 생산환경에서 단일기계의 총 가중작업완료 시간을 최소화하는 문제에 대해 몇 가지 해결 가능한 경우에 대한 연구를 수행하였다. 한편, Mosheiov[24]는 단계적 퇴화함수의 작업시간을 가지고 총 작업소요시간을 최소화하는 단일기계일정계획이 NP-complete함을 증명하였다. Cheng and Ding[7]은 단일기계의 일정계획문제에서 각 작업들의 작업처리시간이 단일납기를 초과하게 되면 단계적 함수로 퇴화하게 되는 생산환경에서 총 작업소요시간, 흐름시간, 가중흐름시간의 목적함수들에 대해 NP-complete함을 증명하였다. 이외에도 다수의 작업처리시간이 선형 단계적 함수로 퇴화되는 단일 기계생산일정계획에 관한 문제에 대한 복잡도 및 특수한 상황에서 Polynomial time에 해결 가능한 알고리즘에 관한 연구가 수행되었다[16, 17, 23].

한편, 복구조정에 관한 연구들을 살펴보면, Lee and Leon[21]이 최초로 복구조정활동(RMA)을 정의하고 단일기계 생산일정계획문제에 이를 반영한 연구하였다. 그들은 단일 RMA를 고려한 단일기계 일정계획 문제에 관해 4개의 목적함수로 분류하고, 각 문제에 대한 최적해의 성질들을 규명하여 동적계획법을 이용한 알고리즘을 제시하였다. Qi et al.[27]는 배치생산방식의 단일기계 일정계획문제에서 복수의 예방정비를 고려한 연구를 수행하였다. Lee and Lin[19]은 기계의 고장이 확률적 프로세스에 의해 발생하는 상황에서 사후정비와 예방정비를 동시

에 고려하는 단일기계 일정계획 문제에 관한 연구를 수행하였다. Mosheiov and Sarig[26]는 단일 납기가 주어진 작업들의 일정과 RMA 시점을 동시에 결정하는 단일기계 일정계획 문제에 관해 연구하였다. RMA를 고려한 병렬기계 일정계획에 대해서는 Lee and Chen[20]은 계획기간 내에 모든 기계에 한번의 RMA가 수행된다는 제약 조건을 가지고 병렬기계 일정계획문제를 해결하였다. 이상의 연구들에서는 기계 가공시간의 경과에 따른 작업처리속도의 퇴화만을 고려하거나, 작업처리속도의 퇴화는 고려하지 않고 RMA만을 고려하고 있다. 퇴화와 RMA는 생산일정계획에 상호 밀접한 연관성과 중요성을 가지고 있으나, 퇴화와 RMA 두 가지를 동시에 고려하는 생산일정계획 문제에 관한 연구는 많지 않다. Lodree and Geiger[22]는 작업처리시간이 작업시작시간에 비례하여 선형으로 증가(퇴화)하는 단일기계 생산일정계획에 단일 RMA를 고려한 연구를 수행하였다. 이들은 주어진 문제의 특수한 경우들에 대한 최적해의 성질들을 규명하고, 최적 RMA의 위치와 작업의 순서 결정에 이용하였다. Zhao and Tang[34]은 단일 납기창(common due-window)하에 단일 RMA와 작업의 퇴화를 동시에 고려한 단일기계의 생산일정계획을 연구하였다. 이들은 납기창을 위반하는 비용을 최소화하는 일정계획을 위한 최적해의 성질들을 경우(Case)별로 분류하여 규명하였다. Kim and Joo[15]는 복수 개의 RMA와 작업 위치에 따라 비선형으로 퇴화하는 작업시간을 동시에 고려한 단일기계 생산일정계획에 관한 연구를 수행하였다. Iranpoor et al.[13]는 작업 순서에 따라 달라지는 작업준비시간을 갖는 단일기계의 생산일정계획 문제에 단일 RMA를 고려한 문제를 연구하였다. 최근에는 병렬기계 일정계획문제에서 퇴화와 RMA를 동시에 고려한 연구가 수행되고 있다[31, 32, 33].

지금까지의 연구들에서는 현실적으로 퇴화와 복구조정활동을 동시에 고려하는 연구는 많지 않음을 알 수 있고 작업 처리시간이 생산시스템에 많이 사용되는 단계적 퇴화함수로 표현되는 연구는 현재까지 존재하지 않음을 알 수 있다. 따라서, 본 연구에서는 단일복구조정활동하에 단계적 퇴화를 가지는 단일기계 일정계획에 관한 문제의 해결을 위한 효과적인 알고리즘을 제시하고자 한다. 본 논문의 구성은 다음과 같다. 제 2장에서 주어진 단일 RMA와 각 작업들의 단계적 퇴화를 동시에 고려한 단일기계의 생산일정계획 문제에 대한 최적화 혼합정수계획모형을 제시하였다. 제 3장에서 해의 효율적인 도출을 위해 서로 다른 염색체 표현을 갖는 세 가지 유전알고리즘을 제안하고, 제 4장에서는 임의로 생성된 예제들을 통해 제시한 알고리즘들의 성능을 평가하였다.

2. 혼합정수계획모형

<파라미터>

- J : 작업 수
 p_j : 작업 j 의 정규 작업 처리시간
 a_j : 작업 j 의 실제 작업 처리시간
 r_j : 작업 j 의 퇴화(작업시간 증가)율
 d_j : 작업 j 의 퇴화 시작시점
 R : 작업보정활동(RMA) 시간
 k : 작업영역을 나타내는 인덱스
 (1 : RMA 이전 영역, 2 : RMA 이후 영역)

<연속 의사결정변수>

- x_j : 작업 j 의 작업 시작시간
 C_k : 작업영역 k 의 총 작업 완료시간

<0-1 정수 의사결정변수>

- y_{jk} : 작업 j 가 작업영역 k 에서 처리되면 1,
 그렇지 않으면 0.
 z_{ijk} : 작업영역 k 에서 작업 i 가 작업 j 에 선행되면 1,
 그렇지 않으면 0.

<혼합정수계획모형>

$$\text{Min.} \quad \sum_{i=1}^2 C_k + R \quad (1)$$

$$\text{S.T.} \quad a_j = \begin{cases} p_j, & x_j \leq d_j \\ p_j(1+r_j), & \text{otherwise} \end{cases} \text{ for } \forall i, \quad (2)$$

$$x_i + a_i \leq x_j + M \cdot \left(1 - \sum_{k=1}^2 z_{ijk}\right), \quad \text{for } \forall i, j \\ \text{and } i \neq j \quad (3)$$

$$x_i + a_i \leq C_k + M \cdot (1 - y_{ik}), \quad \text{for } \forall i, k \quad (4)$$

$$\sum_{i=1}^J z_{iik} = 1, \quad \text{for } \forall k, \quad (5)$$

$$\sum_{j=1}^J z_{jik} = y_{ik}, \quad \text{for } \forall i, k \quad (6)$$

$$\sum_{\substack{j=1 \\ j \neq i}}^J z_{ijk} \leq y_{ik}, \quad \text{for } \forall i, k \quad (7)$$

$$\sum_{k=1}^2 y_{ik} = 1, \quad \text{for } \forall i, \quad (8)$$

$$X_i \geq 0, \quad \text{for } \forall i, \quad (9)$$

$$C_k \geq 0, \quad \text{for } \forall k, \quad (10)$$

$$y_{ik} = 0 \text{ or } 1, \quad \text{for } \forall i, k, \quad (11)$$

$$z_{ijk} = 0 \text{ or } 1 \quad \text{for } \forall i, j, k \quad (12)$$

제약식 (2)는 단계적 퇴화효과를 고려한 실제 작업처리시간에 대한 제약식이다. 여기서, 단계적 퇴화효과는 각 작업이 해당 작업의 퇴화 시작시점보다 늦게 가공이 시작되면 진행되게 된다. 제약식 (3)은 동일 작업영역상의 작업은 작업의 선후관계를 고려하여 각 작업 시작시간을 제약하며, 제약식 (4)는 각 작업영역에 할당된 모든 작업의 퇴화를 고려한 실제 작업시간을 고려하여 총 작업완료시간을 계산하기 위한 제약식이다. 제약식 (5)는 각 작업영역에 할당된 모든 작업들 중 첫 번째 순서에 반드시 하나의 작업이 존재해야 함을 나타낸다. 제약식 (6)은 특정 작업영역에 할당된 작업이 존재한다면, 그 작업의 선행 작업은 반드시 한 개만 존재해야 함을 제약한다. 제약식 (7)은 특정 작업영역에 할당된 작업이 존재한다면, 그 작업의 후속작업은 1개 이상은 존재하지 않음을 나타낸다. 제약식 (8)은 각 작업은 반드시 작업영역 중 한 곳에만 할당되어야 됨을 제약한다.

3. 유전알고리즘

앞의 선행연구에서 본 연구의 문제는 NP-hard한 문제에 속하고 실제로 본 논문에서 제시한 혼합정수계획모형을 이용하여 RMA 이전과 RMA 이후의 작업영역에 평균적으로 4개의 작업 이상이 할당되는 문제의 경우 LINGO나 CPLEX와 같은 최적화 소프트웨어를 사용하여 제한된 시간(2시간) 내에 최적해를 탐색하기 어렵다. 따라서, 본 연구에서는 규모가 큰 현실적 크기의 문제들을 위한 효과적인 메타휴리스틱 기법 중에 하나인 유전알고리즘(Genetic algorithm) 개발에 초점을 맞추었다.

일반적으로 유전알고리즘은 생물의 진화과정인 적자생존의 원칙을 활용해 해를 탐색해가는 메타휴리스틱 기법의 하나로 본 연구에서 다루는 문제와 같은 조합(Combinatorial) 최적화 문제에 대해 짧은 시간 내에 근사 최적해를 도출하는 것으로 알려져 있다[10]. 일반적인 탐색 기법들과는 달리 유전알고리즘은 염색체(Chromosome)라 불리는 해들로 이루어진 해집단(Population)을 하나의 세대(Generation)로 정의하고 유전적인 변이(Crossover, Mutation)과정을 통해 다음 세대의 다음 세대로의 후보 집단을 형성 후, 적합도(Fitness)라는 평가척도를 통하여 확률적으로 우수한 염색체를 가진 개체들을 다음 세대로의 진화(Evolution)시키며 이를 반복함으로써 해를 탐색하게 된다.

본 연구에서는 효율적인 해의 도출을 위해 본 연구에서는 염색체의 표현 및 염색체-해의 전환(Decoding)절차에 따른 세 가지 종류의 염색체 표현들에 대한 유전알고리즘들의 해의 성능을 비교하고자 한다. 문제의 구조적

형태상 본 연구의 문제는 단종 병렬기계 일정계획문제와 유사성을 보인다. 따라서, 병렬기계일정계획 문제에서 일반적으로 많이 소개되는 두 가지 염색체 표현들을 소개하고 본 논문에서 제안하는 지역해 탐색방법을 이용하는 효율적인 염색체 방법을 제안한다. 먼저, 병렬기계일정계획 문제에서 일반적으로 사용되는 두 가지 방법은 2차원 배열(Double String) 염색체 표현을 이용한 2차원 배열 유전알고리즘(Genetic Algorithm with Double String : GA_DS)과 특수문자(Special Character)를 사용한 염색체 표현 방법을 사용한 특수문자 유전 알고리즘(Genetic Algorithm with Special Character : GA_SC)이라 부른다. 마지막으로 본 논문에서 제시하는 작업의 작업영역 할당순서를 결정하는 1차원 배열과 지역해 탐색기법을 통한 디스패칭 법칙을 사용하여 해당 작업들을 특정 작업영역에 할당하는 하이브리드한 디스패칭 유전알고리즘(Genetic Algorithm with Dispatching Rule : GA_DR)이라 부른다. 우선, 2차원 배열을 이용한 염색체 표현 방법은 첫 번째 배열은 개의 1부터 J 까지의 정수를 이용하여 작업의 순서를 나타내고, 두 번째 배열은 2개의 RMA 전후를 의미하는 이진표현(0 혹은 1)을 이용하여 RMA 전후 할당을 나타낸다. 특수 문자를 이용한 염색체 표현 방법은 RMA 전후를 구분하는 개의 특수문자 *와 J 개의 1부터 J 까지의 정수를 이용한 1차원 배열 방법을 이용하여 해를 표현한다. 여기서, 특수문자*전후의 숫자들은 각각 RMA 전후에 가공되는 작업들의 순서를 의미한다. 따라서, 두 가지 염색체 표현방법들 모두 RMA의 위치 및 RMA 전후에 할당된 작업들의 작업순서를 동시에 결정하게 된다. <Figure 1>은 동일한 해에 대한 두 가지 염색체 표현 대한 <Figure 1>(a)는 2차원 배열을 이용한 염색체 표현으로 첫 번째 배열은 작업의 작업영역 할당 순서 및 선후관계를 결정하고 두 번째 배열은 해당 작업의 RMA 전후 할당을 결정한다. 여기서 0은 작업이 RMA 이전에 할당될 경우를, 1은 해당 작업이 RMA 이후에 할당될 경우를 의미한다. <Figure 1>(b)는 특수문자를 이용한 염색체 표현이다. 두 가지 염색체 표현 모두 작업 7, 9, 2, 4, 그리고 5가 RMA 이전에 작업 8, 3, 6, 10, 그리고 1이 RMA 이후에 순차적으로 할당되는 일정계획을 나타낸다.

8	7	9	3	2	6	4	10	1	5	Sequence
1	0	0	1	0	1	0	1	1	0	Assignment

(a) Double String

7	9	2	4	5	*	8	3	6	10	1
---	---	---	---	---	---	---	---	---	----	---

(b) Special Character

<Figure 1> Representation of Two Chromosomes

마지막으로 디스패칭 법칙을 이용한 염색체 표현방법은 (0, 1)구간의 임의의 실수를 작업 J 개수만큼 생성하여 실수로 표현된 작업들을 오름차순으로 정렬하여 표현되는 순서를 특정 작업영역에 할당되는 작업순서로 표현하는 방법이다. 일단 작업의 작업영역 할당순서가 정해지면 작업의 작업영역의 할당 및 해당 작업영역 작업순서 결정은 지역해 탐색기법을 통한 디스패칭 법칙을 사용하여 결정하게 된다. 세부적인 디코딩 절차는 아래에 설명된다.

DECODE_GA_DR Algorithm

- Step 1 : 생성된 실수를 오름차순으로 정렬하여작업 번호를 입력하여 해당 작업의 작업영역 할당순서를 결정한다.
- Step 2 : 작업할당 순서대로 해당 작업을 각각의 작업영역에 할당하여 작업영역의 현재 총 완료시간을 구한다.
- Step 3 : 총 완료시간이 가장 짧은 작업영역에 해당 작업을 할당한다.
- Step 4 : 할당작업이 남아있다면 Step 2로 돌아가고 그렇지 않으면 종료한다.

초기세대의 해집단(Population)을 형성하기 위한 초기 해들은 임의(Random)로 생성된다. 해집단 내의 염색체들은 적합도(Fitness) 함수로써 총 작업완료시간을 갖는다. 일반적으로 유전알고리즘은 교차변이연산(Crossover), 돌연변이연산(Mutation), 재생산(Reproduction)의 세 가지 유전 작업들을 이용하여 유전형질을 다음세대로 계승한다. 교차변이연산으로 하나의 교차점을 임의로 선택하고 두 부모의 유전자를 교차하여 자식을 생성하는 한점교차(One cut-point crossover)방법을 사용하였고 돌연변이연산으로는 교차변이(Swap mutation)방법을 세 가지의 염색체 표현들에 동일하게 적용하였다. 다음 세대를 구성하기 위한 염색체 재생산을 위한 재생산과정에서는 각 세대의 최우수 염색체가 후속세대로 넘어가지 않는 경우가 발생 수 있으므로 본 연구에서는 각 세대의 최우수 염색체 1개를 선정하여 다음세대로 바로 복사하는 정책(elitist policy)을 사용하였고 또한 부모 해는 현 세대 해집단의 적합도 평균보다 좋은(높은) 적합도를 갖는 해들의 집단과 유전변이과정이 시행된 집단의 풀(Pool)에서 룰렛휠 방법에 의해 다음 세대를 선택한다. 이와 같은 방법을 사용하면 자식 해는 기본적으로 선정된 부모 해들을 임의로 짝을 지어 교차변이나 돌연변이 연산으로 생성하나, 가장 우수한 해가 일정비율 이상 부모가 되도록 조정하게 된다. 재생산과정을 정리하면 아래의 EPLVOLE 알고리즘과 같다.

EPVOLVE Algorithm

- Step 1 : 현 세대 해집단의 염색체들의 적합도를 계산한다.
 Step 2 : 최우수 염색체를 다음 세대로 그대로 사용한다.
 Step 3 : 현 세대 해집단에서 적합도가 평균보다 우수한 해집단과 유전변이 과정을 시행 해집단을 다음 세대를 형성하기 위한 해집단 풀(Pool)로 형성한다.
 Step 4 : Step 3에 의해 생성된 해집단 풀로부터 룰렛 휠 방법에 의해 다음 세대의 해집단을 형성하기 위한 나머지 염색체들을 선택한다.

4. 알고리즘 성능 실험

앞서 제시한 휴리스틱 알고리즘들의 성능을 평가하기 위해 본 절에서는 임의로 생성한 많은 예제들을 사용하여 알고리즘의 성능을 비교한다. 본 문제의 알고리즘의 성능은 문제의 복잡도와 퇴화의 정도에 따라 영향을 많이 받을 것으로 예측된다. 따라서, 문제의 복잡도는 총 작업 수(J)에 따라 가장 크게 영향을 받고 퇴화의 정도에 영향을 미치는 작업 퇴화 시작시점에 생성에 따라 많은 영향을 받으므로 이 두 모수들을 몇 개의 수준으로 나누어서 실험을 실시한다. 평균적인 작업의 수는 20, 30, 40, 50, 60 등의 5개의 값 중에 하나로 하고, $\omega = \sum_{i=1}^J a_i/2$ 이라 놓으면 각 작업퇴화 시작시점은 $U[1, 0.5 \times \omega]$, $U[0.5 \times \omega, \omega]$, $U[1, \omega]$ 등의 3개의 분포 중 하나로 임의의 값을 생성한다. 여기서, $U[a, b]$ 는 a 와 b 사이에서 임의로 생성된 정수라는 의미이다. 작업퇴화 시작시점을 구분하여 생성한 이유는 알고리즘의 성능이 작업퇴화 시작시점의 생성에 영향을 미치는지에 대한 여부를 판별하기 위해 다음과 같이 작업퇴화 시작시점이 빠른 그룹, 늦은 그룹, 보통인 그룹으로 나누어 생성하였다. 따라서, 총 15개의 조합의 문제를 임의로 생성하여 각 문제당 10번 반복하여 생성한다. 문제를 생성하기 위해서는 a_i , $r_k (i=1, 2, \dots, J)$, R 값을 생성하여야 한다. 정규 작업 처리시간(a_i)은 하루 총 작업시간이 480분(8시간)이라고 가정하고 해당 문제의 총 작업 수(J)가 하루에 작업이 처리될 수 있는 평균 작업처리 시간에서 [-20%, +20%] 사이에 임의로 생성한다. 즉, $p_i = U[-0.2 \times 480/J, 0.2 \times 480/J]$ 이다. 또한, 퇴화율은 $r_i = U[0.01, 0.1]$ 로 가정한다. 마지막으로 RMA시간은 작업영역당 평균적인 총 퇴화시간의 10%인 $R = \sum_{i=1}^J a_i/2$ 으로 가정한다. 생성된 문제들은 앞서 제시한 3가지의 알고리즘(GA_DS, GA_SC, GA_DR)을 사용하여 해결한다. 혼합정수모형에 의한 최적해는 ILOG CPLEX 12.4을 적용하여 구하였고 두 가지 유

전알고리즘은 C#을 이용하여 구현하였으며, 모든 실험은 1.86GHz Intel Core 2 Processor, 2GB RAM의 PC를 이용하여 수행되었다. 유전알고리즘 파라미터 설정은 제시된 GA_DS, GA_SC, GA_DR 모두 해집단의 수를 $2 \cdot J$ 으로, 총 세대수를 1,000으로 하였고, 교차변이율과 돌연변이율은 예비 실험결과에서 가장 좋은 성능을 보였던 0.8과 0.2로 수행하였다.

작은 규모의 문제들에 관한 수치실험들의 실험결과는 <Table 1>에 큰 규모의 수치실험들의 실험결과는 <Table 2>에 각각 정리되어 있다. 이 표들의 세 가지의 알고리즘의 성능과 관련된 척도가 사용된다. 첫째로, 해의 절대적 혹은 상대적 성능 평가를 위해 분자를 해당 알고리즘으로 구한 총 완료시간(makespan)으로 하고 분모를 각 값들은 작은 규모의 문제에서는 CPLEX로 구한 최적해를 큰 규모의 문제에서는 세 가지의 알고리즘들 중 가장 우수한 해 값(최선해)으로 사용하여 이 비율 값을 100으로 곱한 상대 비율 편차(Relative Percentage Deviation : RPD)를 사용한다. RPD는 최선해 혹은 최적해에 대한 알고리즘에 의한 해의 편차의 비율을 의미하고 아래의 식 (13)으로 표현된다.

$$RPD(\%) = \frac{\text{알고리즘의 해} - \text{최선해(최적해)}}{\text{최선해(최적해)}} \times 100, \quad (13)$$

둘째로, 제안된 알고리즘들의 성능을 평가할 또한, 한 가지 척도로 평균 절대편차(Mean Absolute Deviation : MAD)를 사용한다. MAD는 반복실험을 통해 나온 해들의 절대적인 편차를 의미하고 식 (14)로 표현된다. 마지막으로, 해의 탐색시간의 평균값을 사용한다.

$$MAD(\%) = \frac{[\text{해} - \text{해평균}]}{\text{해평균}} \times 100, \quad (14)$$

<Table 1>의 실험결과를 보면 최적해는 작업수가 늘어나면 해의 탐색시간이 급격히 늘어남을 알 수 있다. 본 연구에서 제안한 세 가지 유전알고리즘들인 GA_DS, GA_SC, 그리고 GA_DR 모두 대다수의 경우 최적해를 탐색하였으며, RPD 전체 평균도 0.15%, 0.43%, 0.09%로 최적해 대비 아주 작아 그 성능이 모두 우수하다고 판단된다. MAD 역시 세 가지 알고리즘 모두 0.09%, 0.06%, 0.05%로 동일한 실험 데이터로 반복 실험하여도 큰 변화 없이 유사한 해를 탐색함을 알 수 있다. 해의 탐색시간에 있어서는 세 가지 알고리즘 모두 평균적으로 0.02, 0.01, 0.01초로 0.02초 이내에 해를 탐색하는 것으로 나타났다.

<Table 1> Results of Numerical Experiment for Small Sized Problems

Data		CPLEX		GA_DS			GA_SC			GA_DR		
Job	Deterioration	Opt. Sol	Time (Sec.)	RPD (%)	MAD (%)	Time (Sec.)	RPD (%)	MAD (%)	Time (Sec.)	RPD (%)	MAD (%)	Time (Sec.)
7	U(1, 0.5w)	560.12	2.892	0.00	0.00	0.01	0.00	0.00	0.01	0.00	0.00	0.01
7	U(1, w)	573.23	2.545	0.00	0.00	0.01	0.00	0.00	0.01	0.00	0.00	0.01
7	U(0.5w, w)	593.12	2.145	0.00	0.00	0.01	0.00	0.01	0.01	0.00	0.00	0.01
8	U(1, 0.5w)	612.34	14.892	0.15	0.04	0.01	0.35	0.03	0.01	0.05	0.00	0.01
8	U(1, w)	633.18	14.545	0.07	0.02	0.02	0.13	0.02	0.01	0.03	0.00	0.01
8	U(0.5w, w)	653.58	14.145	0.01	0.00	0.01	0.11	0.01	0.01	0.01	0.00	0.01
9	U(1, 0.5w)	783.15	413.892	0.29	0.21	0.02	0.79	0.17	0.01	0.09	0.05	0.01
9	U(1, w)	794.17	422.545	0.17	0.10	0.02	0.67	0.10	0.01	0.07	0.10	0.01
9	U(0.5w, w)	805.39	402.145	0.05	0.02	0.02	0.55	0.02	0.02	0.09	0.14	0.01
10	U(1, 0.5w)	900.17	3002.892	0.55	0.45	0.02	0.95	0.15	0.01	0.35	0.05	0.01
10	U(1, w)	913.69	2792.545	0.34	0.15	0.02	0.84	0.15	0.02	0.24	0.05	0.01
10	U(0.5w, w)	933.25	2732.145	0.12	0.05	0.02	0.74	0.05	0.02	0.12	0.15	0.02
Average				0.15	0.09	0.02	0.43	0.06	0.01	0.09	0.05	0.01

<Table 2> Results of numerical experiment for large sized problems

Data		Best Sol.	GA_DS			GA_SC			GA_DR		
Job	Deterioration		RPD (%)	MAD (%)	Time (Sec.)	RPD (%)	MAD (%)	Time (Sec.)	RPD (%)	MAD (%)	Time (Sec.)
20	U(1, 0.5w)	5527.1	29.24	13.53	0.61	19.45	8.91	0.63	1.12	10.00	0.63
20	U(1, w)	506	0.00	0.00	0.50	0.00	0.00	0.53	0.00	0.00	0.51
20	U(0.5w, w)	560.5	11.98	14.98	0.56	19.15	6.97	0.59	1.87	8.10	0.58
30	U(1, 0.5w)	6180.59	20.44	8.99	1.45	37.52	10.19	1.58	1.47	6.60	1.54
30	U(1, w)	516	1.10	1.74	1.23	2.80	4.91	1.32	0.00	0.00	1.29
30	U(0.5w, w)	1040.78	73.76	16.22	1.40	95.25	15.31	1.51	2.41	19.88	1.48
40	U(1, 0.5w)	4384.95	33.33	9.39	2.79	55.54	11.40	3.07	2.58	7.70	3.03
40	U(1, w)	495	0.00	0.00	2.32	1.32	2.08	2.63	0.00	0.00	2.50
40	U(0.5w, w)	629.67	57.70	13.11	2.58	67.67	13.44	2.88	2.95	9.80	2.86
50	U(1, 0.5w)	17494.72	72.33	6.00	4.82	96.35	26.56	5.35	2.83	12.41	5.27
50	U(1, w)	506	1.00	1.79	3.88	12.37	7.33	4.51	0.00	0.00	4.32
50	U(0.5w, w)	1021.28	81.90	15.26	4.45	102.52	16.81	5.05	5.53	22.13	4.98
60	U(1, 0.5w)	8134.9	27.37	13.14	7.24	67.95	14.36	8.21	6.71	7.83	8.18
60	U(1, w)	486	10.21	5.64	5.98	25.96	10.18	7.04	1.51	1.79	6.73
60	U(0.5w, w)	1320.24	67.89	24.66	6.87	109.50	16.20	7.87	9.03	21.56	7.82
Average			32.55	9.63	3.112	47.56	10.98	3.519	2.53	8.52	3.449

큰 규모의 문제들에 관한 실험 결과인 <Table 2>에서는 GA_DR의 RPD값이 2.53%로 일반적인 염색체 표현에 의한 GA_DS와 GA_SC의 RPD값 32.55%, 47.56%에 비해 월등히 우수함을 알 수 있다. 또한, MAD 역시 GA_DR의 MAD값이 8.52%로 GA_DS와 GA_SC의 MAD값 9.63%, 10.98%에 비해 상대적으로 적은 편차를 보이고 있다. 세 알고리즘 모두 퇴화의 시작시간 분포가 총 작업처리시간 전반에 골고루 퍼져있는 경우에는 상대적으로 좋은 성능을 보이거나 그렇지 않은 경우 즉, 상대적

으로 총 작업처리간의 전반부 혹은 후반부 분포하는 경우에는 좋은 결과를 보이고 있지 않다. 그 원인은 단계적 퇴화가 너무 많이 발생해도 혹은 너무 적게 발생해도 GA가 최적해를 찾지 못하고 지역해에 수렴하기 때문이라고 추론된다. 마지막으로 GA_DR이 상대적으로 GA_DS보다는 계산시간이 많이 소요되지만, 그 차이가 매우 적고, 세 알고리즘 모두 수초 이내에 현실적 규모의 문제에 대해 효율적인(Efficient) 일정계획을 수립할 수 있다고 판단된다.

5. 결론

본 연구는 작업 처리시간이 일반적인 생산시스템에 많이 사용되는 단계적 퇴화함수로 표현되고 단일 작업보정 활동(RMA)을 고려하는 단일기계 생산일정계획문제를 다루었다. 우선 문제의 해결을 위해 문제의 상황을 잘 표현해주는 수리계획 모형을 제시하였고 이 최적화 모형은 NP-hard한 문제이므로 현실적 크기의 문제를 해결하기 위해 염색체-해의 전환(Decoding)절차에 따른 세 가지 종류의 염색체 표현들에 대한 유전알고리즘들의 해의 성능을 임의로 생성된 다양한 수치실험을 통해 그 성능을 비교·분석하였다. 실험의 결과 현실적 규모의 문제에서 디스패칭 유전알고리즘(GA_DR)의 성능이 2차원 배열의 염색체 표현의 유전알고리즘(GA_DS)과 특수문자를 사용한 염색체 표현 방법을 사용한 유전알고리즘(GA_SC)에 비해 월등히 우수함을 알 수 있었다. 해의 탐색시간 측면에서는 GA_DR이 GA_DS와 GA_SC에 비해 상대적으로 약간 우수하였으나 모두 큰 규모의 문제에서도 수초 이내에 해를 탐색하는 효율성을 입증하였다.

본 연구의 후속으로 앞으로 다음과 같은 내용을 연구할 계획이다. 우선, 본 연구는 복수의 작업보정활동들이 고려되는 상황 하에서 본 문제를 해결할 필요성을 제시하고 있으며 추후에 작업보정활동들이 고려되는 상황 하의 연구가 수행될 필요가 있다고 판단된다. 또한, 문제의 목적함수인 총 작업 완료시간(makespan)에 있어서 문제의 RMA의 수 역시 영향을 미치게 되므로 이를 고려하여야 할 것이다. 마지막으로 현재의 문제를 병렬기계의 상황으로 확장할 필요가 있다고 판단된다.

Acknowledgement

이 논문은 2013년도 정부(미래창조과학부)의 재원으로 한국연구재단의 기초연구사업지원을 받아 수행된 것임(과제번호 : NRF-2013R1A1A1012454).

References

- [1] Alidaee, B. and Womer, N.K., Scheduling with time dependent processing times : review and extensions. *Journal of the Operational Research Society*, 1999, Vol. 50, No. 7, p 711-721.
- [2] Bachman, A., Janiak, A., and Kovalyov, M.Y., Minimizing the total weighted completion time of deteriorating jobs. *Information Processing Letters*, 2002, Vol. 81, No. 2, p 81-84.
- [3] Bahalke, U., Yolmech, A.M., and Shahanaghi, K. Meta-heuristics to solve single-machine scheduling problem with sequence-dependent setup time and deteriorating jobs. *International Journal of Advanced Manufacturing Technology*, 2010, Vol. 50, No. 5-8, p 749-759.
- [4] Bank, M., Fatemi Ghomi, S.M.T., Jolai, F., and Behnamian, J., Application of particle swarm optimization and simulated annealing algorithms in flow shop scheduling problem under linear deterioration. *Advances in Engineering Software*, 2012, Vol. 47, No. 1, p 1-6.
- [5] Browne, S. and Yechiali, U., Scheduling deteriorating jobs on a single processor. *Operations Research*, 1999, Vol. 38, p 495-498.
- [6] Cheng, T.C.E. and Ding, Q., Single machine scheduling with deadlines and increasing rates of processing times. *Acta Informatica*, 2000, Vol. 36, p 673-692.
- [7] Cheng, T.C.E. and Ding, Q., Single machine scheduling with step-deteriorating processing times. *European Journal of Operational Research*, 2001, Vol. 134, No. 3, p 623-630.
- [8] Cheng, T.C.E., Ding, Q., and Lin, B.M.T., A concise survey of scheduling with time-dependent processing times. *European Journal of Operational Research*, 2004, Vol. 152, p 1-13.
- [9] Cheng, T.C.E., Hsu, C. J., Huang, Y.C., and Lee, W.C., Single-machine scheduling with deteriorating jobs and setup times to minimize the maximum tardiness. *Computers and Operations Research*, 2011, Vol. 38, No. 12, p 1760-1765.
- [10] Gen, M. and Cheng, R., *Genetic Algorithms and Engineering Optimization*. New York : Wiley, 2000.
- [11] Gupta, J.N.D. and Gupta, S.K., Single facility scheduling with nonlinear processing times. *Computers and Industrial Engineering*, 1988, Vol. 14, p 387-393.
- [12] Huang, X. and Wang, M.Z., Parallel identical machines scheduling with deteriorating jobs and total absolute differences penalties. *Applied Mathematical Modelling*, 2011, Vol. 35, No. 3, p 1349-1353.
- [13] Iranpoor, M., Ghomi, S.M.T., and Zandieh, M., Machine scheduling in the presence of sequence-dependent setup times and a rate-modifying activity. *International Journal of Production Research*, 2012, Vol. 50, No. 24 p 7401-7414.
- [14] Jafari, A. and Moslehi, G., Scheduling linear deteriorating jobs to minimize the number of tardy jobs. *Journal of Global Optimization*, 2012, Vol. 54, No. 2, p 389-404.

- [15] Kim, B.S. and Joo, C.M., Single-machine total completion time scheduling with position-based deterioration and multiple rate-modifying activities. *Industrial Engineering and Management Systems*, 2011, Vol. 10, No. 4, p 247-254.
- [16] Kovalyov, M.Y. and Kubiak, W., A fully polynomial approximation scheme for minimizing makespan of deteriorating jobs. *Journal of Heuristics*, 1988, Vol. 3, No. 4, p 287-297.
- [17] Kubiak, W. and van de Velde, S., Scheduling deteriorating jobs to minimize makespan. *Naval Research Logistics*, 1998, Vol. 45, No. 5, p 511-523.
- [18] Kunnathur, A.S. and Gupta, S.K., Minimizing the makespan with late start penalties added to processing times in a single facility scheduling problem, *European Journal of Operational Research*, 1990, Vol. 47, No. 1, p 56-64.
- [19] Lee, C.Y. and Lin, C.S., Single-machine scheduling with maintenance and repair rate-modifying activities, *European Journal of Operational Research*, 2001, Vol. 135, p 493-513.
- [20] Lee, C.Y. and Chen, Z.L., Scheduling of jobs and maintenance activities on parallel machines, *Naval Research Logistics*, 2000, Vol. 47, p 61-67.
- [21] Lee, C.Y. and Leon, V.J., Machine scheduling with a rate-modifying activity. *European Journal of Operational Research*, 2001, Vol. 128, p 119-128.
- [22] Lodree, E.J. and Geiger, C.D., A note on the optimal sequence position for a rate-modifying activity under simple linear deterioration, *European Journal of Operational Research*, 2010, Vol. 201, No. 2, p 644-648.
- [23] Mosheiov, G. and Sarig, A., Scheduling a maintenance activity and due-window assignment on a single machine, *Computers and Operations Research*, 2009, Vol. 36, No. 9, p 2541-2545.
- [24] Mosheiov, G., Scheduling jobs with step-deterioration; Minimizing makespan on a single- and multi-machine. *Computers and Industrial Engineering*, 1995, Vol. 28, No. 4, p 869-879.
- [25] Mosheiov, G., V-Shaped polices to scheduling deteriorating jobs. *Operation Research*, 1991, Vol. 39, p 979-991.
- [26] Moslehi, G. and Jafari, A., Minimizing the number of tardy jobs under piecewise-linear deterioration, *Computers and Industrial Engineering*, 2010, Vol. 59, No. 4, p 573-584.
- [27] Qi, X., Chen, T., and Tu, F., Scheduling the maintenance on a single machine. *Journal of the Operational Research Society*, 1990, Vol. 50, p 1071-1078.
- [28] Ruiz-Torres, A-J., Paletta, G., and Perez, E., Parallel machine scheduling to minimize the makespan with sequence dependent deteriorating effects. *Computers and Operations Research*, 2013, Vol. 40, p 2051-2061.
- [29] Sundararaghavan, P.S. and Kunnathur, A.S., Single machine scheduling with start time dependent processing times : some solvable cases, *European Journal of Operational Research*, 1994, Vol. 78, No. 3, p 394-403.
- [30] Wang, J.L., Sun, L., and Sun, L., Single-machine total completion time scheduling with a time-dependent deterioration. *Applied Mathematical Modelling*, 2011, Vol. 35, p 1506-1511.
- [31] Yang, D.-L. and Yang, S-J., Unrelated parallel-machine scheduling problems with multiple rate-modifying activities. *Information Science*, 2013, Vol. 235, p 280-286.
- [32] Yang, D.-L., Cheng, T.C.E., Yang, S.-J., and Hsu, C.-J., Unrelated parallel-machine scheduling with aging effects and multi-maintenance activities. *Computers and Operations Research*, 2012, Vol. 39, p 1458-1464.
- [33] Yang, D.-L., Cheng, T.C.E., and Yang, S-J., Parallel-machine scheduling with controllable processing times and rate-modifying activities to minimise total cost involving total completion time and job compressions. *International Journal of Production Research*, 2013, Vol. 52, No. 4, p 1133-1141.
- [34] Zhao, C. and Tang, H., A note to due-window assignment and single machine scheduling with deteriorating jobs and a rate-modifying activity, *Computers and Operations Research*, 2012, Vol. 39, No. 6, p 1300-1303.