

다양성유지를 기반으로 한 Job-shop Scheduling Problem의 진화적 해법

Genetic Algorithms based on maintaining a diversity of the population for Job-shop Scheduling Problem

권창근* · 오갑석**

Chang-keun Kwon* and Kab-suk Oh**

*투어인포 대표 ** 동명대학 정보통신계열

요 약

유전자알고리즘(Genetic Algorithm)은 확률적인 집단 탐색법이고 적응도함수의 형태에 관계없는 직접 탐색법이기 때문에 최근 최적화 방법으로 주목을 받고 있다. 본 논문에서는 Job-shop Schedule Problem에 대하여 교배방법으로 JOX를 사용하되, 효율적인 탐색을 위하여 탐색범위를 축소시키는 강제조작으로 형질유전을 고려한 형질유전GT법을 제안하고, 세대교체에 있어 모집단의 다양성을 유지하기 위하여 집단 내에 동일한 개체를 배제하는 방법을 제안한다. 제안 알고리즘을 Fisher & Thompson의 FT10×10 및 FT20×5 문제에 적용하여 유효성을 실험적으로 검증한다.

Abstract

This paper presents a new genetic algorithm for job-shop scheduling problems. When we design a genetic algorithm for difficult ordering problems such as job-shop scheduling problems, it is important to design encoding/crossover that is excellent in characteristic preservation and to maintain a diversity of population. We used Job-based order crossover(JOX). Since the schedules generated by JOX are not always active-schedule, we proposed a method to transform them into active schedules by using the GT method with characteristic preservation. We introduce strategies for maintaining a diversity of the population by eliminating same individuals in the population. Furthermore, we are not used mutation. Experiments have been done on two examples: Fisher's and Thompson's 10x10 and 20x5 benchmark problem.

Key Words : GA, 형질유전 GT법, 다양성유지, Job shop Scheduling Problem, Job-based Order Crossover

1. 서 론

유전자알고리즘(Genetic Algorithm: GA)은 확률적인 집단 탐색법이고 적응도함수의 형태에 관계없는 직접 탐색법이기 때문에 최근 최적화 방법으로 주목을 받고 있다. Job-shop Schedule Problem는 각 기계간의 의존관계가 매우 밀접하여, 가장 풀기 어려운 순서결정문제의 하나로서 유전자알고리즘을 적용하기에 적합한 문제이다. 그러나 GA를 최적화문제, 특히 Job-shop Scheduling Problem와 같은 풀기 어려운 순서결정문제에 적용할 경우 형질유전성이 우수한 교배방법의 설계가 요구된다. 여기서 개체의 특성을 나타내는 형질은 개체의 적응도를 결정짓는 것으로 형질유전에 실패하면 GA는 부작위 탐색과 비슷한 성능 밖에 발휘하지 못한다. 비록 형질유전성이 우수한 교배를 설계했다라도 탐색의 효율을 증대시키기 위해 강제조작을 사용할 경우 개체의 유전형질을 고려하지 않으면 교배에 의해 생성된 개체의 유

전형질이 보존되지 않는 문제점이 있다. 또한 형질유전성이 우수한 교배방법과 형질유전을 고려한 강제조작을 설계했고 하더라도 모집단의 다양성이 유지되지 못하면국소해로 수렴하여 교배와 강제조작의 우수 기능이 발휘될 수 없으므로 모집단의 다양성유지를 위한 연구도 필요하다.

최근 Job-shop Scheduling Problem에 유전자알고리즘으로 접근한 논문은, Davis, Nakano, Yamada, Fang, Mattfeld, Kobayashi, Shi, Ono 등이 있다[1]-[8]. Yamada[3], Mattfeld[5]에서는 GA 특유의 오퍼레이터인 교배가 단지 힐 등 반법의 초기값 생성을 위해 사용되었고, Davis[1], Nakano[2], Fang[4], Yamada[10]은 교배를 주 오퍼레이터로 이용하고 있으나, 좋은 성능을 발휘하지 못하므로 형질유전성이 우수하다고는 말할 수 없다. Kobayashi[6]에서는 교배를 주 오퍼레이터로 사용하고 탐색의 효율을 향상시키기 위하여 탐색공간을 좁히는 강제조작법으로 Giffler & Thompson법(GT법)을 도입하였다. 그러나 GT법은 교배에 의해 생성된 개체의 유전형질을 고려하지 않으므로 최적해로 수렴하지 못하고 발산할 수 있다. Ono[8, 15]에서는 최적해 탐색능력을 향상시키기 위하여 형질유전을 고려한 교배 방법인 JOX(Job-based Order Crossover)법을 제안하고, 강제조작법으로는 GT법[8]과 힐등반법을 적용[15]하고, 집단의 다양성유지를

접수일자 : 2000년 6월 20일

완료일자 : 2001년 5월 2일

본 논문은 2000년도 동명대학 학술연구 조성비로 연구되었음.

위하여 돌연변이[8, 15]를 적용하였다. 그러나 GT법은 경합 집합 중에서 무작위로 작업을 선택함으로써 유전형질과 무관한 액티브 스케줄이 생성되며, 힐 등반법은 국소해로 수렴할 가능성을 갖고 있으며, 개체의 유전인자를 인위적으로 조작하는 돌연변이의 적용은 돌연변이 고유의 분체와 최적해로의 수렴에 한계가 있다.

이와 같은 문제점을 해결하기 위하여 본 논문에서는 Job-shop Scheduling Problem에 형질유전을 고려한 교배 및 강제조작을 적용하고, 집단의 다양성유지를 위하여 집단 내에 동일한 개체를 배제하는 방법을 제안한다. 제안 알고리즘의 코드화는 각 작업을 Gantt chart로 표시할 경우, Gantt chart에 투입되는 작업의 순서(작업순열)로 하며, 교배방법은 JOX를 적용하며, 강제조작은 형질유전GT법을 적용하며, 모집단의 다양성유지를 위하여 동일개체 배제를 적용한 세대교체로 구성된다.

본 논문의 2장에서는 Job-shop Scheduling Problem와 JOX법에 대하여 설명하고, 형질유전GT법 및 다양성유지를 적용한 Job-shop Scheduling Problem의 진화적 해법을 제안한다. 3장에서는 제안한 방법을 Fisher & Thompson의 FT10×10 및 FT20×5 문제[9]에 적용하여 그 유효성을 실험적 고찰을 통하여 성능을 평가하고, 4장에서는 결론을 논 의한다.

2. Job-shop Scheduling Problem의 진화적 해법

2.1 Job-shop Scheduling Problem와 JOX

Job-shop Scheduling Problem는 n 개의 업무(job, item) $J_i(i=1, 2, \dots, n)$ 와 m 대의 기계 $M_q(q=1, 2, \dots, m)$ 에 대하여 각 업무를 가공하는 기계의 순서와 각 업무가 각 기계에서 처리되는 시간이 주어졌을 때, 주어진 목적함수가 최적이 되도록 각 업무가 각 기계에서 가공이 시작되는 시각을 결정해야 할 때, 순서를 부여하는 문제가 발생한다. 이때, 모든 업무가 m 대의 기계에서 각각 한번씩 가공되는 것으로 할 경우, 각 업무 J_i 를 가공하는 기계의 순서 $M_q(r=1, 2, \dots, m)$ 을 기술적순서(technological ordering)라고 하며, $T_i=(iq_1, iq_2, \dots, iq_m)$ 로 표현하고, $T_i(i=1, 2, \dots, n)$ 를 제행으로 하는 $n \times m$ 행렬 T 를 기술적순서 행렬이라 한다. 또한 각 업무 J_i 가 기계 M_q 에서 처리되는 시간을 가공시간(processing time)라 하고, $P_{i,q}$ 로 표현한다. 이때, 각 업무 i 에 가공시간렬 $P_i=(p_{i,1}, p_{i,2}, \dots, p_{i,m})$ 가 대응하고 P_i 를 제 i 행으로하는 $n \times m$ 행렬 P 를 가공시간 행렬이라 한다. 각 업무 J_i 가 각 기계에서 처리되는 것을 작업(operation)이라 하고, 작업 iq (Jiq)로 표현한다. 기술적순서와 가공시간이 주어졌을 때, 모든 가공이 끝날 때까지 걸리는 총 소요시간을 makespan 이라 한다. 따라서 Job-shop Scheduling Problem은 n 개의 업무와 m 대의 기계에 대하여, 기술적 순서행렬과 가공시간행렬이 주어졌을 때 목적함수인 makespan이 최소가 되도록 각 기계의 작업순서를 결정하여, 각 작업의 시작시각을 결정하는 것이다. 단, 각 기계의 종류는 모두 다르며, 동시에 두가지 이상의 업무를 가공할 수 없다. 또한 각 기계상의 작업은 주어진 가공시간 동안 각 기계상에서 중단없이 가공되는 것으로 한다. 본 논문에서는 여기서 결정된 각 기계의 작업순서를 작업열이라 한다. Fisher & Thompson의 6개의 업무를

6대의 기계에 가공하는 FT6x6문제를 기술적순서 행렬 T 와 가공시간 행렬 P 로 나타내면 각각 식(1)과 식(2)와 같다.

$$T = \begin{pmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \\ T_5 \\ T_6 \end{pmatrix} = \begin{pmatrix} J_{11} & J_{12} & J_{13} & J_{14} & J_{15} & J_{16} \\ J_{21} & J_{22} & J_{23} & J_{24} & J_{25} & J_{26} \\ J_{31} & J_{32} & J_{33} & J_{34} & J_{35} & J_{36} \\ J_{41} & J_{42} & J_{43} & J_{44} & J_{45} & J_{46} \\ J_{51} & J_{52} & J_{53} & J_{54} & J_{55} & J_{56} \\ J_{61} & J_{62} & J_{63} & J_{64} & J_{65} & J_{66} \end{pmatrix} = \begin{pmatrix} M_3 & M_1 & M_2 & M_4 & M_6 & M_5 \\ M_2 & M_3 & M_5 & M_6 & M_1 & M_4 \\ M_1 & M_4 & M_6 & M_1 & M_2 & M_5 \\ M_2 & M_1 & M_3 & M_4 & M_5 & M_6 \\ M_3 & M_2 & M_5 & M_6 & M_1 & M_4 \\ M_2 & M_4 & M_6 & M_1 & M_5 & M_3 \end{pmatrix} \quad (1)$$

$$P = \begin{pmatrix} P_{1,3} & P_{1,1} & P_{1,2} & P_{1,4} & P_{1,6} & P_{1,5} \\ P_{2,2} & P_{2,3} & P_{2,5} & P_{2,6} & P_{2,1} & P_{2,4} \\ P_{3,3} & P_{3,4} & P_{3,6} & P_{3,1} & P_{3,2} & P_{3,5} \\ P_{4,2} & P_{4,1} & P_{4,3} & P_{4,4} & P_{4,5} & P_{4,6} \\ P_{5,3} & P_{5,2} & P_{5,5} & P_{5,6} & P_{5,1} & P_{5,4} \\ P_{6,2} & P_{6,4} & P_{6,6} & P_{6,1} & P_{6,5} & P_{6,3} \end{pmatrix} = \begin{pmatrix} 1 & 3 & 6 & 7 & 3 & 6 \\ 8 & 5 & 10 & 10 & 10 & 4 \\ 5 & 4 & 8 & 9 & 1 & 7 \\ 5 & 5 & 5 & 3 & 8 & 9 \\ 9 & 3 & 5 & 4 & 3 & 1 \\ 3 & 3 & 9 & 10 & 4 & 1 \end{pmatrix} \quad (2)$$

예를 들면, 업무1(J_1)의 기술적인 작업순서(T_1)는 기계 $M_3, M_1, M_2, M_4, M_6, M_5$ 이며, 각 기계에서 가공하는데 소요되는 시간은 각각 1, 3, 6, 7, 3, 6 단위시간임을 의미한다.

여기에 각 작업의 가공시간을 부여하여 각 작업의 가공시작시각과 가공완료시각을 결정된 것을 스케줄이라 하며, 이 스케줄을 그림으로 나타내는 방법으로는 작업열을 기계별로 표시하는 Gantt chart가 이용된다. 그림 1은 6개의 업무를 6대의 기계에서 가공할 경우, 기계 M_q 에 업무 J_i 가 가공되는 최적의 작업열을 Gantt chart로 표현한 예이다. y축은 기계의 종류를 나타내며, x축은 각 업무의 가공시간(종료시간-시작시간)을 나타낸다.

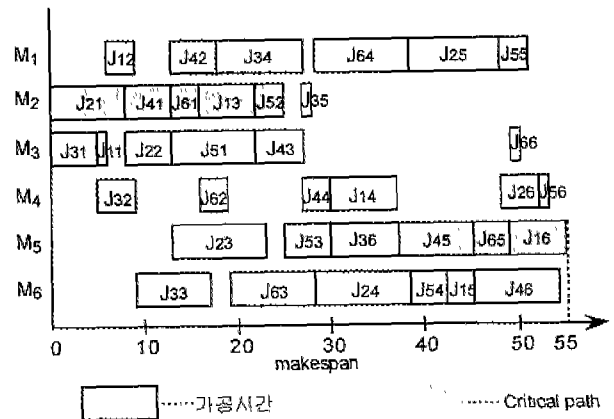


그림 1. 6개의 부품을 6대의 기계에서 가공할 경우의 Gantt chart 예

그림 1에서 각 작업열은 각 업무 사이에 여유시간(다음 업무의 시작시간 - 현재 업무의 종료시간)이 존재한다. 모든 업무를 대상으로, 가장 먼저 시작되는 작업으로부터 가장 마지막으로 끝나는 작업까지의 경로들 중에 여유시간의 합이 최소인 경로가 존재하는데 이를 Critical path라 한다.

Job-shop Scheduling Problem의 코드화는 기계명과 작업명을 한 쌍으로 하여 Gantt chart에 할당하는 순서(작업순열)로 나타낸다. 예를 들어 3개의 업무를 3대의 기계에서 가공할 경우의 코드화는 그림 2와 같이 작업 J_{iq} 와 기계 M_q 의 쌍으로 표시할 수 있다. 그림 2에서 작업을 기준으로 볼 때, 작업 J_{11} 은 기계 M_1 에서 가공하고, 작업 J_{32} 은 기계 M_2 에서 가

포함됨 의미한다. 기계 M_1 은 작업 J_{11} 을 가공한 다음 작업 J_{31} 을 가공해야하고, 다음으로 작업 J_{21} 을 가공해야 한다는 것을 나타낸다. 같은 방법으로 기계 M_2 , 기계 M_3 에서 모든 작업이 이루어진다. 또한 기계를 기준으로 보면, 기계 M_1 은 반드시 J_{11} , J_{31} , J_{21} 순으로 가공해야 함을 의미하고, 같은 방법으로 기계 M_2 , 기계 M_3 의 각 작업의 가공순서가 결정된다.

(J_{11}, M_1) (J_{32}, M_2) (J_{31}, M_1) (J_{23}, M_3) (J_{33}, M_3) (J_{13}, M_3) (J_{22}, M_2) (J_{12}, M_2) (J_{21}, M_1)

그림 2. 3개의 부품을 3대의 기계에서 가공할 경우의 코드화 예

이렇게 코드화된 개체를 교배방법인 JOX법에 적용하면 다음과 같다[11].

- step 1. 부모1과 부모2에서 임의의 작업을 포함하는 형질을 무작위로 선택한다.
- step 2. step 1에서 선택한 형질을 부모1에서 자식1로, 부모2에서 자식2로 유전자자리를 완전히 보존하여 이동시킨다.
- step 3. step 2에서 유전되지 않은 나머지 형질은 부모2에서 자식1로, 부모1에서 자식2로 할당순서에 따라 왼쪽에서 오른쪽으로 빈 유전자자리에 이동시킨다.

JOX법을 도식화하면 그림 3과 같다.

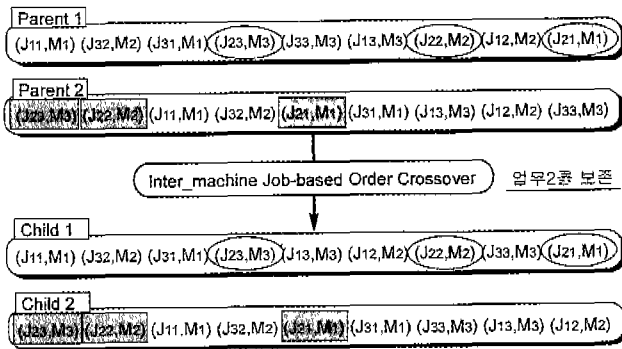


그림 3. Job_based Order Crossover(JOX)

2.2 형질유전을 고려한 GT법

Job-shop Scheduling Problem에서 스케줄은 실행 가능한 스케줄을 의미하며, 각 작업의 가공시작시각과 가공완료시각을 결정하는 것이다. 그러므로 일단기술적 순서와 작업열에 의하여 실행 가능한 스케줄이 결정되면, 직접 후속하는 2개의 작업간에는 기계의 휴식시간이 발생할 수 있다. 이러한 기계의 휴식시간을 차례차례 임의로 증가시켜도 실행 가능한 스케줄이 되므로, 무수히 많은 스케줄을 생성할 수가 있다. 본 논문에서는 각 작업을 가능하면 빨리 가공하는 스케줄을 구하기 위하여 목적함수로 Makespan을 적용한다.

Job-shop Scheduling Problem의 스케줄은 “실행 가능한 스케줄” ≡ “세미액티브 스케줄” ≡ “액티브 스케줄” ≡ “최적 스케줄”로 구분할 수 있다[12]. 세미액티브 스케줄(semiactive schedule)이란 1개의 실행 가능한 스케줄에서 주어진 기술적 순서를 만족하며, 작업열을 변경시키지 않고 가능한 한 왼쪽 방향으로 밀착시켜 가공시작 시각을 빨리 한 스케줄을 의미한다. 액티브 스케줄(active schedule)이란 기

술적인 순서가 보장되는 범위 내에서 작업열의 변경을 허용하고, 가능한 한 왼쪽 방향으로 밀착시켜 가공시작 시각을 빨리 한 스케줄을 의미한다. 최적 스케줄(optimal schedule)은 액티브 스케줄공간 내에 포함되어 있으므로 액티브 스케줄공간을 탐색하는 것이 탐색효율의 증대가 기대된다. 액티브 스케줄을 생성하는 강제조작법으로는 Giffler & Thompson법(GT법)이 널리 알려져 있다[13]. GT법을 적용하여 액티브 스케줄을 생성하는 알고리즘은 다음과 같다.

- step 1. 어떤 기계 M^* 에 가공될 순번인 작업 중에 가공시작 시각이 가장 빠른 작업 O^* 와 처리시간이 중복되는 비스케줄 작업으로 구성된 경합집합 C를 만든다. 여기서 경합집합의 요소는 기계 M^* 에서 가공 완료된 작업 및 전 처리가 끝나지 않은 작업을 제외한 모든 작업으로 구성된다.
- step 2. 경합집합 C 중에서 무작위로 한 개의 작업을 선택하여 스케줄링 한다.
- step 3. 비스케줄 작업이 없어질 때까지 step 1에서 step 2를 반복한다.

이때 비액티브 스케줄에 GT법을 적용하면 step 2에서와 같이 경합집합 C에서 무작위로 한 개의 작업이 선택되므로 교배에 의해 생성된 개체의 유전형질과 무관하게 액티브 스케줄을 생성하게 된다. 즉, GT법을 적용하면 경합집합에 포함된 복수의 작업 중에서 무작위로 1개의 작업을 선택하게 되어 임의성이 존재한다. 임의성이란 계통적인 작업을 선택함으로써 모든 액티브 스케줄을 생성할 수 있고, 또한 무작위로 작업을 선택함으로써 임의의 액티브 스케줄을 생성할 수 있음을 의미한다. 이는 형질유전성이 우수한 코드화/교배를 설계하여 스케줄을 생성하였다하더라도 강제조작을 도입함으로써 유전형질이 보존되지 못하고 초기수축이 일어나 코드화/교배의 기본 성능이 저하되는 결과를 초래한다.

따라서 본 논문에서는 탐색의 효율을 높이기 위해 액티브 스케줄 이외의 개체는 치사유전자로 보고, JOX법에 의해 생성된 실행 가능한 스케줄공간의 개체를 유전형질이 보존되면서 액티브 스케줄공간의 개체로 강제하는 형질유전을 고려한 GT법을 제안한다. 그림 4는 교배방법인 JOX법에 의해 생성된 개체에 형질유전을 고려한 GT법을 적용하는 방법을 보인다.

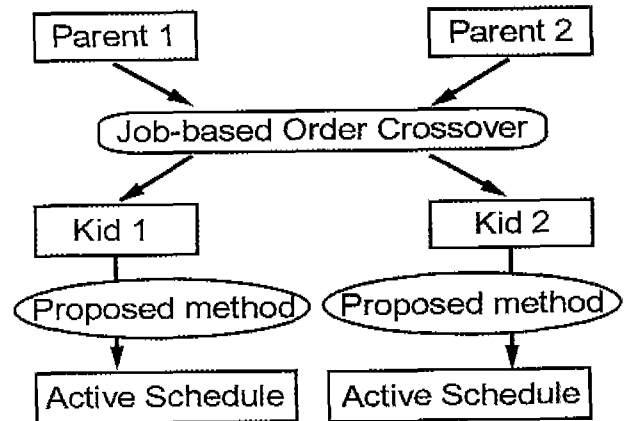


그림 4. 형질유전GT법에 의한 강제조작 개념도

제안 방법인 형질유전을 고려한 GT법은 다음과 같다.

- step 1. 어떤 기계 M^* 에 가공될 순번인 작업 중에 가공시작

시간이 가장 빠른 작업 O^* 와 처리시간이 중복되는 비스케줄 작업으로 구성된 경합집합 C 를 만든다.

- step 2.** 교배에 의해 생성된 기체의 작업열에서 기계 M^* 상의 최초 비스케줄 작업을 Jm^* 이라 하자. 경합집합 C 중에 Jm^* 이 존재할 경우 작업 $O(M^*, Jm^*)$ 를 스케줄링 한다. 만약경합집합 C 중에 Jm^* 가 존재하지 않으면, 기계 M^* 상의 다음 비스케줄 작업을 Jm^* 라하고 경합집합 C 중에 Jm^* 가 존재할 때까지 이를 반복한다.
- step 3.** 비스케줄 작업이 없어질 때까지 step 1에서 step 2를 반복한다.

이와 같이 형질유전GT법을 적용하면 step 2에서 경합집합 중 하나의 작업이 선택될 때 무작위로 선택하지 않고, 교배에 의해 생성된 스케줄의 유전형질인 작업열의 우선순위에 따라 액티브 스케줄로 강제되므로 교배에 의해 생성된 유전형질이 보존된다. 교배에 의해 생성된 스케줄을 GT법과 제안방법인 형질유전GT법을 사용하여 생성된 액티브 스케줄의 예를 그림 5에 표시한다. 그림 5에서 GT법은 대부분의 형질이 변경되었으나, 제안방법인 형질유전GT법은 60% 정도의 형질이 보존되고 있음을 입증하고 있다.

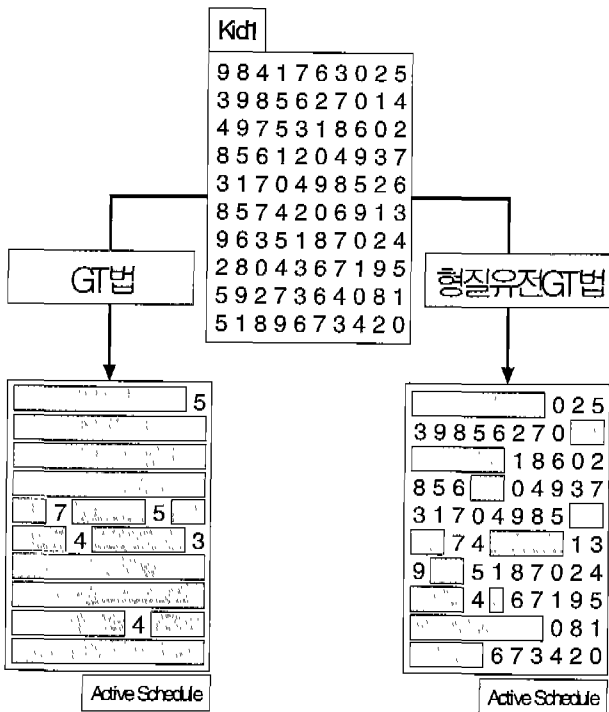
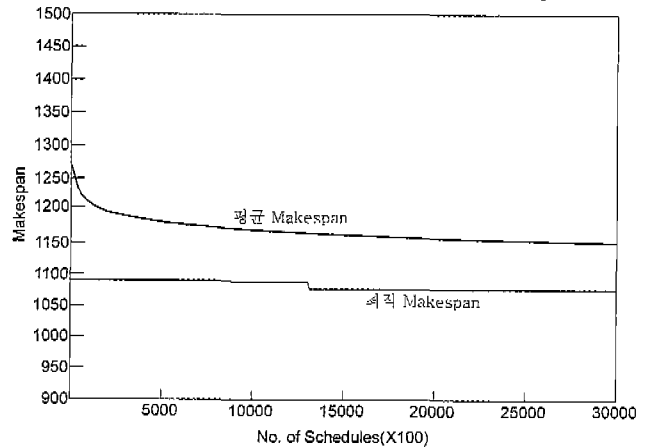


그림 5. GT법과 형질유전GT법에 의해 생성된 액티브 스케줄의 예

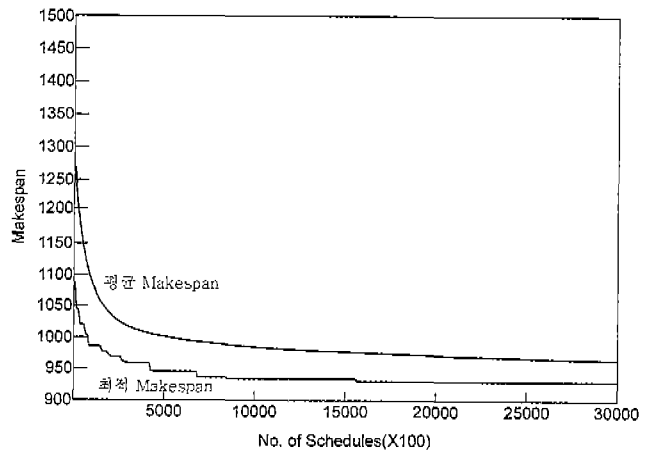
형질유전GT법을 적용하여 강제조작할 때 탐색되는 평균 Makespan과 최적 Makespan의 변화와 최적 Makespan의 빈도분포를 JOX 교배방법에 GT법으로 강제하는 방법과 비교하여 그림 6, 7, 8, 9에 표시한다.

그림 6(a)는 FT10×10문제의 경우, JOX에 의해 생성된 스케줄을 GT법으로 강제조작할 때 탐색된 평균 Makespan과 최적 Makespan의 변화를 보여주며, 그림 6(b)은 JOX에 의해 생성된 스케줄을 형질유전GT법으로 강제조작할 때 탐색된 평균 Makespan과 최적 Makespan의 변화를 보여준다.

그림 7(a)은 FT20×5문제의 경우, JOX에 의해 생성된 스케줄을 GT법으로 강제조작할 때 탐색된 평균 Makespan과 최



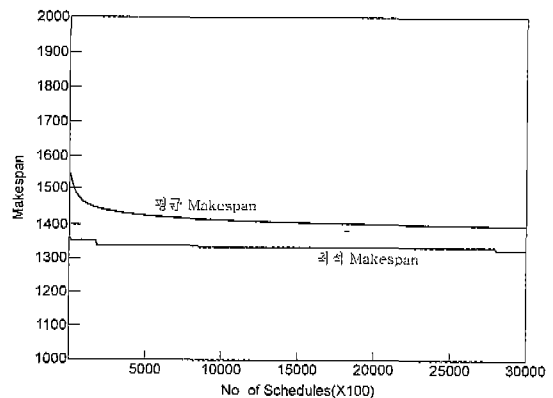
(a). JOX에 GT법 적용



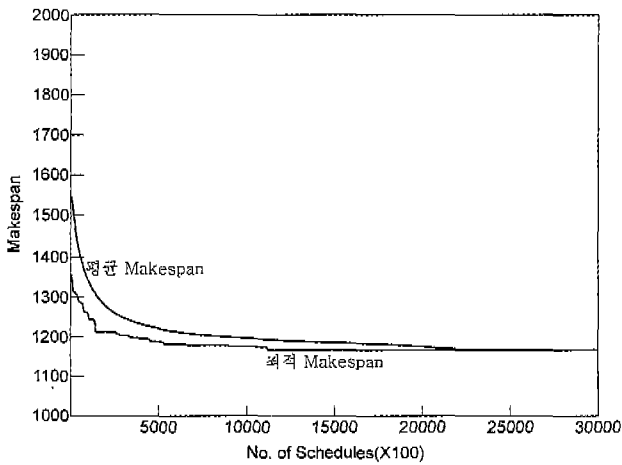
(b). JOX에 형질유전GT법 적용

그림 6. FT10×10의 평균 Makespan과 최적 Makespan의 변화 곡선

적 Makespan의 변화를 보여주며, 그림 7(b)은 JOX에 의해 생성된 스케줄을 형질유전GT법으로 강제조작할 때 탐색된 평균 Makespan과 최적 Makespan의 변화를 보여준다. 그림 8(a)은 FT10×10문제의 경우, JOX에 의해 생성된 스케줄을



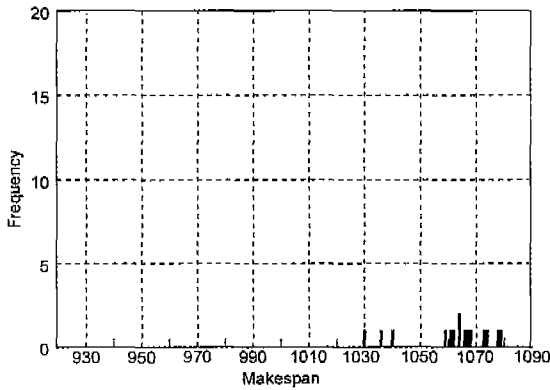
(a). JOX에 GT법 적용



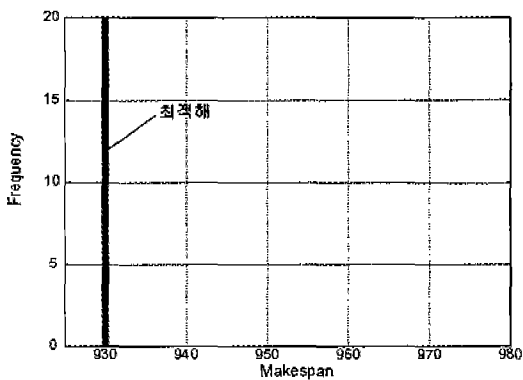
(b). JOX에 형질유전GT법 적용

그림 7. FT20×5의 평균 Makespan과 최적 Makespan의 변화 곡선

GT법으로 강제조작할 때 탐색된 최적 Makespan의 빈도를 보여주며, 그림 8(b)은 JOX에 의해 생성된 스케줄을 형질유전GT법으로 강제조작할 때 탐색된 최적 Makespan의 빈도를 보여준다. 그림 9(a)는 FT20×5문제의 경우, JOX에 의해 생성된 스케줄을 GT법으로 강제조작할 때 탐색된 최적 Makespan의 빈도를 보여주며, 그림 9(b)는 JOX에 의해 생성된 스케줄을 형질유전GT법으로 강제조작할 때 탐색된 최적 Makespan의 빈도를 보여준다.

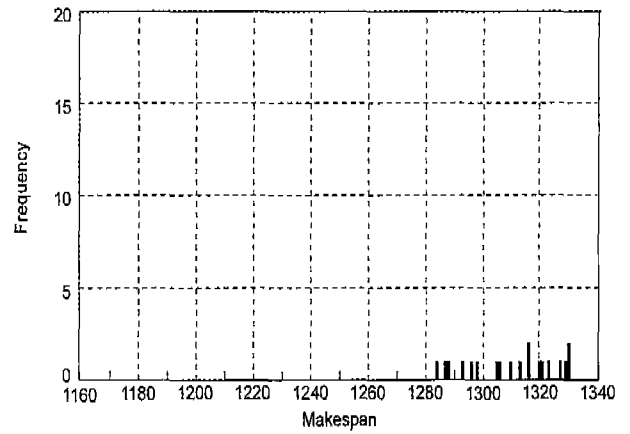


(a). JOX에 GT법 적용

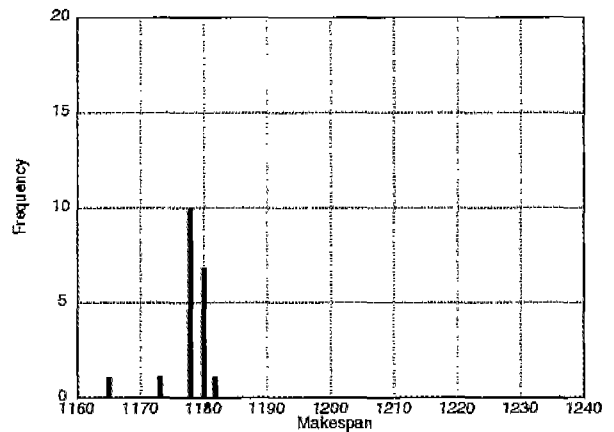


(b). JOX에 형질유전GT법 적용

그림 8. FT10×10의 최적 Makespan 빈도분포



(a). JOX에 GT법 적용



(b). JOX에 형질유전GT법 적용

그림 9. FT20×5의 최적 Makespan 빈도분포

그러나 GT법은 근본적으로 경합집합의 개수가 많으면 탐색의 효율이 저하될 수 있다. 따라서 오른 채움과 왼 채움방식을 통해 경합집합의 개수가 적은 채움 방식을 적용한다. 왼 채움이란 기존의 스케줄링방식과 같이 시작작업을 기

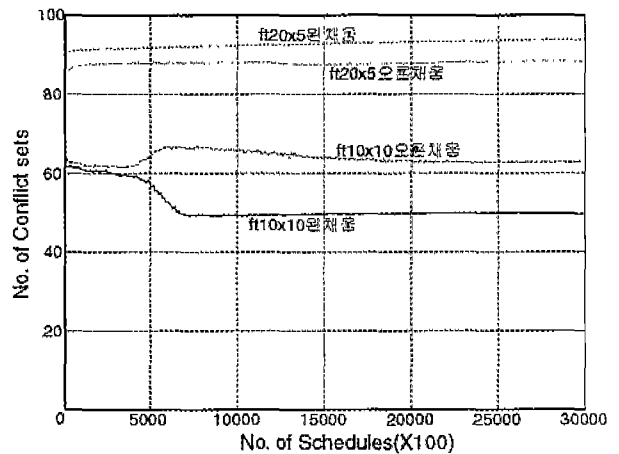


그림 10. 채움 방식에 따른 경합집합의 개수

준으로 스케줄링 하는 방식을 의미하며, 오른 채움은 왼 채움과는 반대로 마지막 작업을 기준으로 하여 스케줄링 하는 방식을 의미한다. 여기서 오른 채움과 왼 채움에 있어서 최적 Makespan은 변하지 않는다. 이때 그림 10은 FT10×10 문제와 FT20×5문제에 대해 왼 채움과 오른 채움방식을 적용할 때에 경합집합 개수의 추이를 나타낸 것이다. x축은 스케줄의 개수를 나타내며, y축은 경합집합의 개수를 나타낸다. 경합집합의 개수가 적은 경우가 교배에 의해 생성된 개체를 액티브 스케줄로 강제함에 있어 유전형질이 더욱 더 잘 보존된다고 할 수 있다.

2.3 집단의 다양성을 고려한 세대교체

세대교체란 모집단에서 교배를 위하여 개체를 선택하고, 교배를 통하여 복수의 새로운 개체를 생성하고, 새로 생성된 개체들 중에서 다음 세대에 생존시키기 위한 개체를 선택하여, 교배를 위해 선택된 개체와 교체하는 것을 의미한다. 본 논문에서 사용하는 세대교체 모델은 MGG(Minimal Generation Gap Model)을 기본으로 사용한다[15]. 세대교체모델 MGG는 모집단에서 2개의 개체(부모)를 무작위로 선택하고, 획득형질의 보존을 위해 복수의 교배를 통하여 복수의 개체(자식)를 생성하고, 모집단에서 복제를 위해 선택한 부모개체와 교배에 의해 생성된 복수의 자식개체 중에서 생존을 위하여 Makespan이 가장 작은 1개와 플릿 월 표본 추출로 1개를 선택하여 부모와 교체하는 부분 세대교체이다.

본 논문에서는 그림 11과 같이 생존선택을 행할 경우, 모집단내에 이미 존재하는 개체가 선택되어 축적될 가능성이 있으므로 모집단의 다양성을 유지하기 위해 모집단의 개체와 동일한 개체를 도태시키고, Makespan이 가장 작은 2개의 개체를 선택하여 부모와 교체한다. 이를 다양성을 고려한 세대교체모델이라 부르기로 한다. 여기서 동일개체의 판단 기준은 “작업순열 동일”, “Critical path 동일”, “작업열 동일”로 구분할 수 있으며, 집단 내의 다양성 유지는 탐색의 효율을 증대시키기 위하여 선별력이 우수한 판단 기준을 적용할 필요가 있다. 각각의 동일개체 판단 기준과 스케줄과의 관계를 설명하면, 작업순열은 스케줄과 1대1 대응관계라고 할 수 없다. 즉, 서로 다른 복수의 작업순열이 동일한 스케줄로 표시될 수 있다. 또한 Critical path는 1개의 스케줄 내에 복수개가 존재할 수 있다. 그러나 작업열은 스케줄과 항상 1대1 대응관계이다.

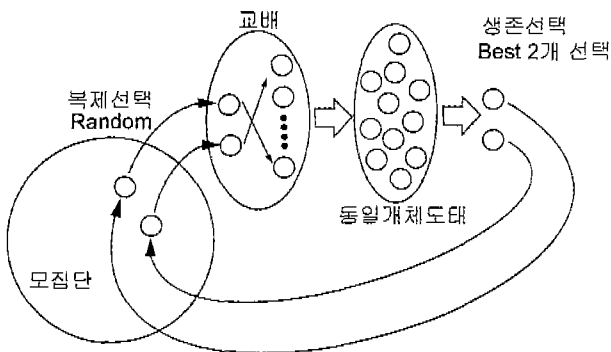


그림 11. 다양성 유지를 고려한 세대교체 모델

본 논문에서는 다음과 같은 다양성유지 기반의 Job-shop Scheduling Problem의 진화적 해법을 제안하며, 이를 도식화하면 그림 12와 같다.

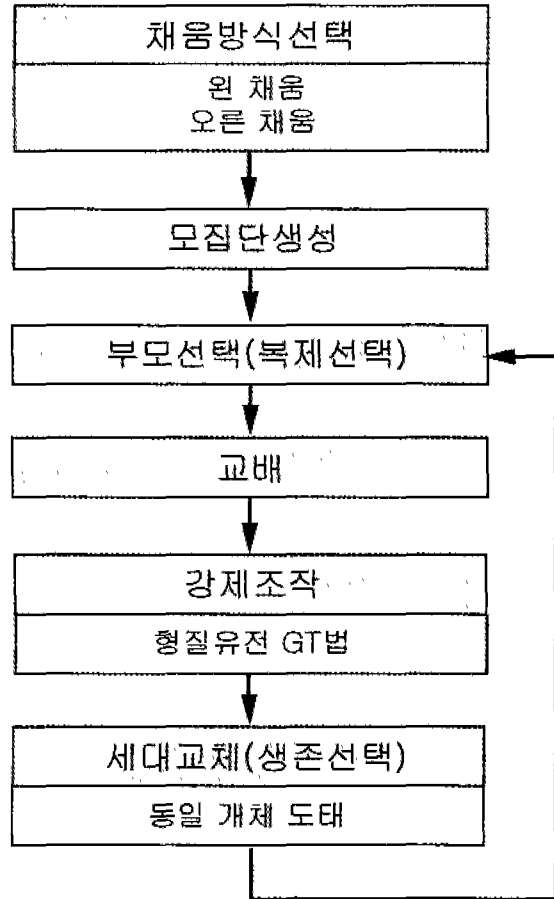


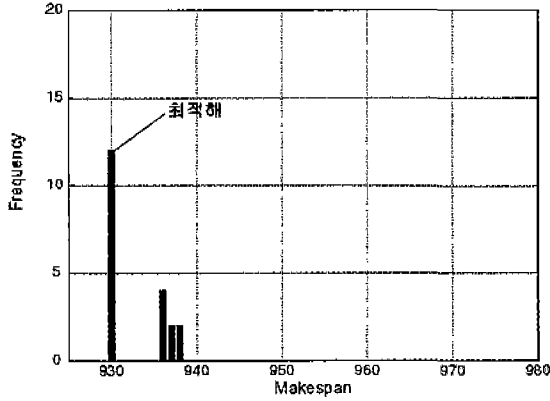
그림 12. 제안 유전자알고리즘의 절차

- step 1. 채움방식선택 : 모집단 후보개체를 형질유전 GT법을 적용하여 생성된 개체들중오른 채움과 왼 채움 방식에 적용하여 경합집합의 개수가 적은 채움 방식을 선택한다.
- step 2. 모집단의 생성 : 액티브 스케줄공간의 개체를 무작위로 생성하여 이를 초기집단으로 한다.
- step 3. 교배를 위한 부모 선택(복제선택) : 교배를 위하여 모집단으로부터 무작위로 2개의 부모개체를 선택한다.
- step 4. 자식 생성(교배) : step 3에서 선택된 부모개체에 대하여 교배 JOX를 n번 적용하여 복수의 자식개체(2n 개)를 생성한다.
- step 5. 강제조작 : step 4에서 생성된 모든 자식에 대하여형질유전GT법을 적용하여 액티브 스케줄로 강제한다. 이때 초기화 단계에서 선택된 채움 방식을 적용하여 스케줄을 생성한다.
- step 6. 세대교체(생존선택) : 2개의 부모와 교배에 의해 생성된 모든 자식 중에서 Makespan이 제일 작은 개체와 두 번째로 작은 개체를 2개의 부모와 교체한다. 단, 모집단내의 개체와 동일한 개체는 도태시킨다.
- step 7. 최소 Makespan과 모집단 평균 Makespan이 같을 때까지 step 2 - step 5를 반복한다.

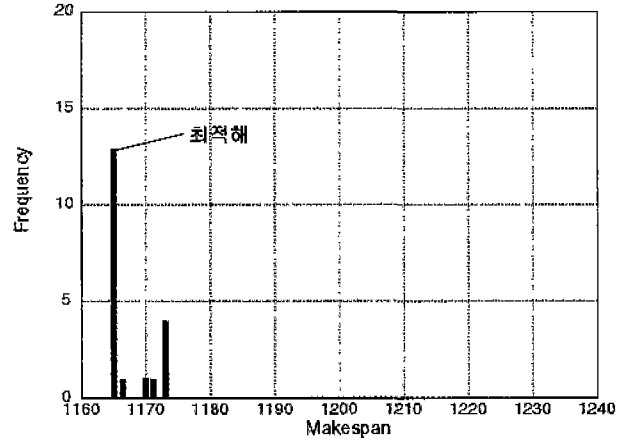
3. 실험 및 고찰

본 실험에서는 기존방법(JOX)과 제안방법인 형질유전GT

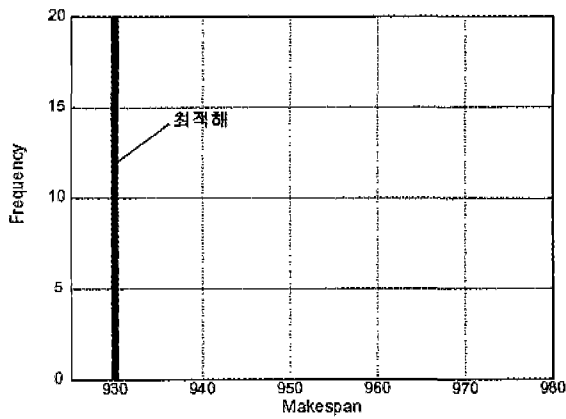
법에 모집단의 다양성을 유지하기 위하여 동일개체 도태의 판단 기준인 “작업순열 동일”, “Critical path 동일”, “작업열 동일”을 각각 Benchmark 문제로 알려져 있는 FT10×10 및 FT20×5 문제에 적용한 결과를 보인다. 실험에서 주어진 각 문제의 최적 Makespan은 930, 1165이고, 모집단의 개체 수는 600개, 교배 횟수는 100번, 평가를 위해 생성하는 스케줄의 개수는 3×10^6 개이다. 제안방법에 대한 실험 결과를 그림 13, 14에 보인다.



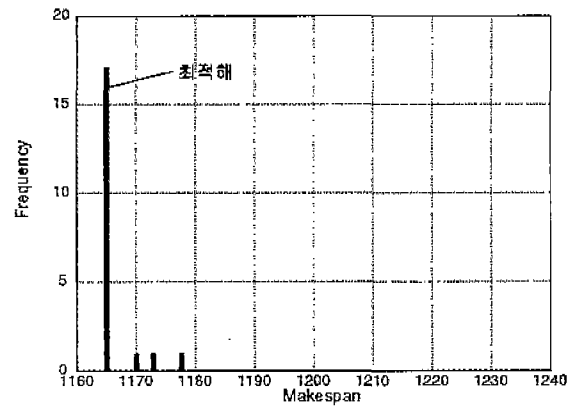
(a). 형질유전GT법에 작업순열 동일개체 도태



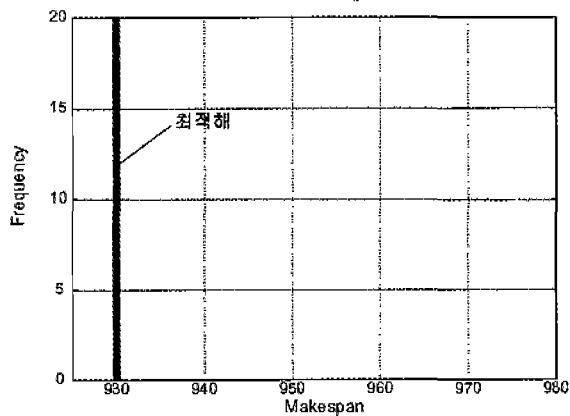
(a). 형질유전GT법에 작업순열 동일개체 도태



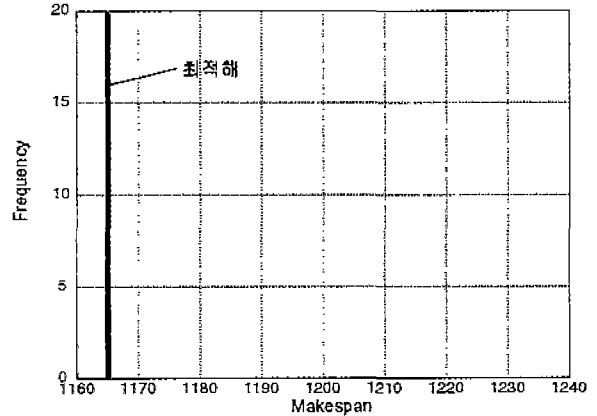
(b). 형질유전GT법에 Critical path 동일개체 도태



(b). 형질유전GT법에 Critical path 동일개체 도태



(c). 형질유전GT법에 작업열 동일개체 도태



(c). 형질유전GT법에 작업열 동일개체 도태

그림 13. FT10×10문제의 최적 Makespan 빈도분포 (다양성유지 세대교체모델을 적용한 경우)

그림 14. FT20×5문제의 최적 Makespan 빈도분포 (다양성유지 세대교체모델을 적용한 경우)

14(a)는 FT20×5문제에 형질유전GT법을 적용하고, 다양성유지를 위하여 작업순열을 동일 개체의 판단기준으로 다양성유지 세대교체모형을 적용한 경우의 최적 Makespan 빈도 분포이며, 그림 14(b)는 동일 개체의 판단기준을 Critical path로 한 경우의 최적 Makespan 빈도분포이고, 그림 14(c)는 작업열을 동일 개체의 판단기준으로 도태시킨 경우의 최적 Makespan 빈도분포이다. 실험에서는 유전자알고리즘의 고 유성능을 비교하기 위하여 돌연변이를 적용하지 않았다. 다양성유지를 위해 작업순열 동일개체 도태를 적용한 경우(그림 13(a), 14(a)), FT10×10문제의 경우, 20회 중 최적 Makespan을 12회 찾았고, FT20×5문제의 경우, 13회의 최적 Makespan을 찾았다. 이는 형질유전을 고려한 교배 및 강제조작법이 탐색능력을 발휘하고 있으나, 다양성유지에 실패한 것으로 보인다. 다양성유지를 위해 Critical path 동일개체 도태를 적용한 경우(그림 13(b), 14(b)), FT10×10문제의 경우, 20회의 시행 중 최적 Makespan을 20회, FT20×5문제의 경우, 17회의 최적 Makespan을 찾았다. 이는 다양성유지의 성능이 향상되었으나, 문제의 유형에 따라 아직 미흡한 것으로 보인다. 다양성유지를 위해 작업열 동일개체 도태를 적용한 경우(그림 13(c), 14(c)), FT10×10문제와 FT20×5문제에 대해 모두 20회 시행 중 최적 Makespan을 모두 찾았다. 이는 모집단의 다양성유지가 문제의 유형에 관계없이 효과적으로 수행되고 있음을 의미한다.

표 1은 小野[15]의 방법과 제안방법의 강제조작 및 다양성유지의 차이점과 최적 Makespan의 탐색비율을 비교한 표이다.

표 2. 기존방법[15]과 제안방법의 성능 비교

구분	강제조작	다양성유지	최적 Makespan(%)	
			FT10×10	FT20×5
기존방법[15]	힌동반	돌연변이	86	100
제안방법1	형질유전GT법	작업순열 동일개체 도태	60	65
제안방법2	형질유전GT법	Critical path 동일개체 도태	100	85
제안방법3	형질유전GT법	작업열 동일개체 도태	100	100

4. 결론

본 논문에서는 각 기계 상에 투입되는 작업의 순서를 유전형질로 판단하고, 효율적인 탐색을 위해 형질유전GT법을 적용하고, 모집단의 다양성유지를 위하여 방법으로 동일개체 배제방법을 이용한 Job-shop Scheduling Problem의 진화적 해법을 제안하였다. 제안방법은 JOX법에 의해 생성되는 새로운 개체가 유전형질을 보존하면서 액티브 스케줄 공간으로 강제되도록 형질유전GT법을 적용하였으며, 효율적인 탐색을 위하여 오른 채움과 왼 채움을 도입하여 교배방법 및 강제조작법을 설계하였다. 그리고 집단의 다양성유지를 위하여 세대교체의 과정 중 생존을 위한 선택방법에서 모집단내에 존재하는 개체와 동일한 개체를 도태시키는 방법으로 작업열 동일개체 배제 전략을 제시하였다.

제안방법을 Benchmark 문제로 알려져 있는 FT10×10 문제와 FT20×5 문제에 적용한 결과, 제안방법인 형질유전GT법에 작업열 동일개체 도태에 의한 다양성유지 세대교체 모델은 FT10×10 문제와 FT20×5 문제에서 100%의 최적

Makespan을 구하였다. 이는 형질유전GT법의 적용으로 효율적인 탐색이 수행되었음을 의미하고, 동일개체 도태의 판단기준으로 “작업열 동일”을 선택하는 것이 모집단의 다양성유지가 효과적으로 수행되고 있음을 입증한다.

참고 문헌

- [1] Davis L., “Job Shop Scheduling with Genetic Algorithms”, *Proceedings of the 1st International Conference on Genetic Algorithms*, pp. 136-140, 1985
- [2] Nakano R. and Yamada T. “Conventional Genetic Algorithms for Job Shop Problems”, *Proceedings of the 4th International Conference on Genetic Algorithms*, pp. 474-479, 1991
- [3] Yamada T. and Nakano R., “A Genetic Algorithm Applicable to Large-Scale Job-Shop Problems”, *Parallel Problem Solving from Nature*, vol. 2, pp. 281-290, 1992
- [4] Fang H., Ross P. and Come D., “A Promising Genetic Algorithm Approach to Job-shop Scheduling, Rescheduling, and Open-Shop Scheduling Problems”, *Proceedings of the 5th International Conference on Genetic Algorithms*, pp. 375-382, 1993
- [5] Mattfeld D.C., Kopfer H. and Bierwirth C. “Control of Parallel Population Dynamics by Social-Like Behavior of GA-Individuals”, *Parallel Problem Solving from Nature*, vol. 3, pp. 16-25, 1994
- [6] Kobayashi S., Ono I. and Yamamura M., “An Efficient Genetic Algorithm for Job-shop Scheduling Problems”, *Proceedings of the 6th International Conference on Genetic Algorithm*, pp. 506-511, 1995
- [7] Shi G., Iima H. and Sannomiya N., “A Method for Constructing Genetic Algorithm in Job Shop Problems”, *第8回 自律分散システム・シンポジウム 資料*, pp. 175-178, 1996
- [8] Ono I., Yamamura M. and Kobayashi S., “A Genetic Algorithm for Job-shop Scheduling Problems Using Job-based Order Crossover”, *Proceedings of 96 IEEE International Conference on Evolutionary Computation*, pp. 547-552, 1996
- [9] Fisher H. and Thompson G.L., “Probabilistic Learning Combinations of Local Job-Shop Scheduling Rules, in *Industrial Scheduling*(eds. by Muth J.F. and Thompson G.L.)”, Prentice-Hall, Englewood Cliffs, N.J., 1963
- [10] Yamada T. and Nakano R., “A Genetic Algorithm with Multi-Step Crossover for Job-Shop Scheduling Problems”, *Proceedings of the 1st IEEE/IEEE International Conference on Genetic Algorithms in Engineering Systems*, *Innovations and Applications*, pp. 146-151, 1995
- [11] 小野, 山村, 小林, “形質遺傳を考慮した順序交叉に基づくジョブショップスケジューリング問題の解法”, 第

23回 知能システム・シンポジウム資料, pp.97-102, 1996

- [12] 鍋島, 스케줄링理論, 森北出版, 1974
- [16] Giffler B. and Thompson G.L. "Probabilistic Learning Combinations of Scheduling Problems", *Operations Research*, vol. 8, pp.487-503, 1969
- [13] 권창근, 오갑석, "유전형질을 고려한 Job shop scheduling의 진화적 해법", *한국 퍼지 및 지능시스템학회 '99 추계학술대회 발표논문집*, vol. 9, no.2, pp. 366-369, 1999
- [14] Satoh H., Yamamura M. and Kobayashi S., "Minimal Generation Gap Model for GAs Considering Both Exploration and Exploitation" *Proceedings of IIZUKA'96*, pp. 494-497, 1996
- [15] 小野, 小林, "形質遺傳に基づくジョブショップ 스케줄링", *生産スケ줄링・シンポジウム '96 講演論文集*, pp. 19-24, 1996

저 자 소 개



권창근(Kwon Chang-keun)

1985년 : 국립경상대학교 경영학과 (학사)
 1998년 : 경공업대학 종합이공학연구과
 지능과학시스템전공 (공학석사)
 2000년 ~ 현재 : 투어인포 대표

관심분야 : 유전자알고리즘, ERP

E-mail : kwon0617@nownuri.net



오갑석(Oh Kab-suk)

1989년 : 부경대학교 전자통신공학과 (학사)
 1993년 : 부경대학교 전자통신공학과 (공학석사)
 1998년 : 동경공업대학 종합이공학연구과 (공학박사)
 1998년 ~ 현재 : 동명대학 정보통신계열
 조교수

관심분야 : 퍼지제어, 지능제어, 프로그래밍 언어, 전자상거래, 멀티미디어

E-mail : oks@yongma.tmc.ac.kr