

공생 진화알고리즘을 이용한 유연조립시스템의 공정계획

김여근^{1*} · 위정미¹ · 신경석¹ · 김용주²

¹전남대학교 산업공학과 / ²도울정보기술

Process Planning in Flexible Assembly Systems Using a Symbiotic Evolutionary Algorithm

Yeo Keun Kim¹ · Jung Mi Euy¹ · Kyoung Seok Shin¹ · Yong Ju Kim²

¹Department of Industrial Engineering, Chonnam National University, Gwangju, 500-757

²Doul INFOTEC, Gwangju, 500-110

This paper deals with a process planning problem in the flexible assembly system (FAS). The problem is to assign assembly tasks to stations with limited working space and to determine assembly routing with the objective of minimizing transfer time of the products among stations, while satisfying precedence relations among the tasks and upper-bound workload constraints for each station.

In the process planning of FAS, the optimality of assembly routing depends on tasks loading. The integration of tasks loading and assembly routing is therefore important for an efficient utilization of FAS. To solve the integrated problem at the same time, in this paper we propose a new method using an artificial intelligent search technique, named 2-leveled symbiotic evolutionary algorithm. Through computational experiments, the performance of the proposed algorithm is compared with those of a traditional evolutionary algorithm and a symbiotic evolutionary algorithm. The experimental results show that the proposed algorithm outperforms the algorithms compared.

Keywords: flexible assembly systems, process planning, symbiotic evolutionary algorithm, loading, routing

1. 서론

유연조립시스템(Flexible Assembly System: FAS)은 조립기능을 수행하는 로봇, 조립품의 운반기능과 저장기능을 수행하는 자동자재 처리설비, 그리고 이들 설비를 제어하는 컨트롤 시스템으로 구성되어 여러 종류의 제품을 자동으로 조립할 수 있는 시스템을 말한다(Lee and Johnson, 1991; Rampersad, 1994). 이러한 시스템은 자동차 조립공정이나 SMT(surface mount technology) 라인, PCB(printed circuit board) 조립 시스템 등 많은 산업분야에서

적용되고 있다. 또한 카세트, 자명종, 볼베어링, 컴퓨터 디스켓 등 고객의 요구가 다양한 제품을 생산하는 시스템에서 폭넓게 사용되고 있다(Groover, 2001).

본 연구에서는 FAS의 공정계획문제를 다룬다. 여기서 공정계획은 제품들을 조립함에 있어 각 제품별로 수행할 각 작업들을 작업장에 할당(loading)하고, 각 제품의 조립경로를 결정(routing)하는 문제로 정의된다. FAS 공정계획은 흔히 많은 제약식과 결정변수를 가지며, 이로 인해 복잡도가 높은 문제이다(Lee and Johnson, 1991; Sawik, 1997). 이러한 이유로, 대부분의 기

이 논문은 2003년도 한국학술진흥재단의 지원에 의하여 연구되었음(KRF-2003-002-D00362)

*연락처 : 김여근 교수, 500-757 광주광역시 북구 용봉동 300번지 전남대학교 산업공학과, Fax : 062-530-1789,

E-mail : kimyk@chonnam.ac.kr

2004년 1월 14일 접수, 1회 수정 후 2004년 5월 3일 게재 확정.

존 연구는 현실적인 여러 상황을 고려하지 않고 문제를 단순화하고 있다. 조립경로가 주어진 조건 아래서 작업할당문제를 다루거나(Agnetis and Nicolo, 1995; Jueung and Ro, 1998), 작업할당과 조립경로를 동시에 통합적으로 다루지 않고 순차적으로 해결하는 방법을 사용하고 있다(Lucertini *et al.*, 1996). 또한 문제의 복잡도를 낮추기 위하여 해공간을 인위적으로 축소하는 방법, 예로 몇 개의 조립경로로 제한하는 방법을 취하고 있다(Sawik, 1997).

본 연구에서는 FAS 공정계획을 함에 있어 작업할당과 조립경로를 동시에 통합적으로 다룬다. 이를 위한 방법론으로는 진화알고리즘(evolutionary algorithm)의 한 형태인 공생(symbiosis) 진화알고리즘을 기반으로 한다. 공생 진화알고리즘은 생물계에서 공생관계를 갖는 서로 다른 종들이 상호작용하고 상호작용하며 진화하는 과정을 모방한 일종의 확률적 탐색기법이다. 공생 진화알고리즘은 복잡도가 높은 큰 규모의 문제를 부분문제로 분할하여 해결하거나, 관련도가 높은 여러 부분문제들로 이루어진 문제를 통합적으로 해결하는 데 효율적인 것으로 인식되고 있다(Potter, 1997; Kim *et al.*, 2003).

본 논문의 구성은 다음과 같다. 본 연구에서 다루는 FAS 공정계획과 그 특징을 2장에서 소개하고, 3장에서는 공생 진화알고리즘의 개념 및 절차를 다룬다. 4장에서는 알고리즘의 적용에 필요한 유전요소들을 제안하고, 5장에서는 제안한 알고리즘의 성능분석을 위한 실험의 결과들을 비교분석한다. 끝으로, 6장에서는 연구성과를 요약하고 결론을 맺는다.

2. 유연조립시스템의 공정계획

2.1 유연조립시스템의 구조와 특징

유연조립시스템은 일반적으로 시스템의 배치, 자체흐름유형, 품종의 다양성과 생산량 등에 따라 여러 형태로 분류할 수 있다(Groover, 2001; Sawik, 1999). 본 연구에서 다루는 유연조립시스템의 구조와 특징은 다음과 같다.

첫째, 작업장 간에 조립작업의 흐름은 직선적인 단방향의 이동뿐 아니라 여러 형태의 작업이동이 가능하다. <그림 1>에서와 같이, 작업장 간 이동에 있어서 전진은 물론이고, 후진(backtracking), 작업장 건너뛰기 등이 가능하다고 본다.

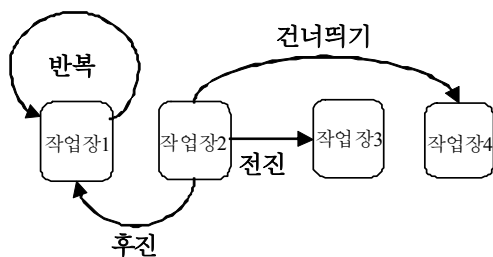


그림 1. 작업들의 작업장 간 이동유형.

둘째, 다기능성 또는 범용기계를 사용함으로써 각 작업장은 높은 유연성을 갖는다. 즉, 한 작업장에서 여러 종류의 작업을 행할 수 있으며, 또한 각 작업은 여러 작업장에서 행하여 질 수 있다. 그러나 각 작업이 모든 작업장에서 작업 가능하지는 않다.

셋째, 유연조립시스템의 각 작업장은 조립기계와 관련 장비로 인해 작업할 수 있는 한정된 공간을 갖는다. <그림 2>에서 보듯이, 조립작업을 위해 필요한 부품공급기(part feeders), 그립퍼(gripper) 등의 시스템 요소들은 로봇이 작업할 수 있는 영역 내에 설치되어야 한다. 따라서 한 작업장에서 행할 수 있는 작업의 수는 제한을 받게 된다. 작업장에서 수행할 수 있는 작업의 수를 유연성 용량(flexibility capacity; Lee and Stecke, 1996) 또는 작업장 공간용량(staging capacity; Park and Kim 1995)이라고 부르고 있다. 본 연구에서는 이를 간략히 공간용량이라 부르기로 한다.

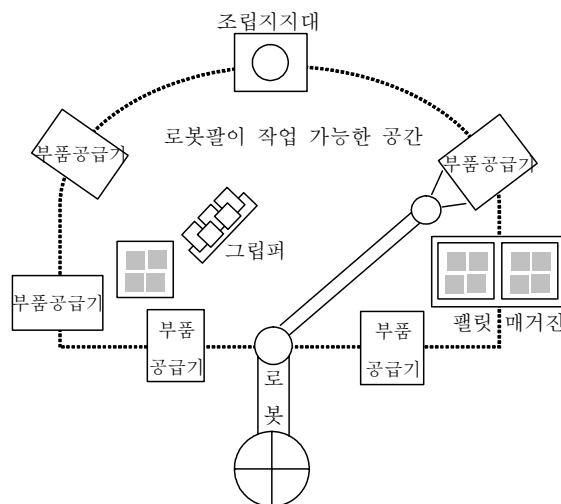


그림 2. 유연조립 작업장 구조.

넷째, 본 연구에서는 공구와 관련된 제약은 무시한다. 유연제조시스템(Flexible Manufacturing System: FMS)에서는 공구매거진의 용량, 가용 공구 수, 그리고 그 수명 등을 중요하게 다루고 있다. 그러나 FAS에서 사용되는 조립용 공구는 FMS의 절삭가공용 공구보다 소모가 적어 공구관리 및 공구교환에 따른 노력과 시간이 상대적으로 적게 요구된다(Lee and Stecke, 1996). 따라서 FAS에서 공구에 대한 제약은 미미하다고 볼 수 있다.

다섯째, 동일한 작업이라 할지라도 작업장에 따라 소요되는 조립시간이 다르다고 본다. 이는 각 작업장에 설치된 조립로봇의 성능과 관련 장비에 따라 다를 수 있기 때문이다.

여섯째, 같은 작업이 제품에 따라 다른 작업장에서 행하여 질 수 있다. 이를 중복할당(duplicate assignment)이라 부른다(Sawik, 1999). 기존의 혼합모델 조립라인에서는 같은 작업은 모델에 상관없이 하나의 작업장에서 행하여 졌다. 중복할당의 허용은 문제의 복잡도를 증가시키지만 작업부하를 분산시킬 수 있어, 각 작업장의 유휴시간과 조립제품의 공정 간 대기시간을 감소(Matta and Hsu, 1995)시킬 수 있다.

2.2 네트워크 표현

본 연구에서는 생산대상 제품들의 조립작업의 선행관계를 전체적으로 표현한 결합선행 네트워크(combined precedence network)를 사용한다. 이 네트워크에서 노드(node)는 작업을, 호(arc)는 선행관계를 나타낸다. 그리고 각 노드에는 해당 작업을 행할 수 있는 작업장, 작업시간, 그리고 이 작업에 요구되는 공간이 표시된다. 이는 <그림 3>과 같다. 그림에서 S는 시작 노드를, F는 끝 노드를 나타내는 가상(dummy)노드로, 한 제품의 조립 시작과 끝을 나타낸다.

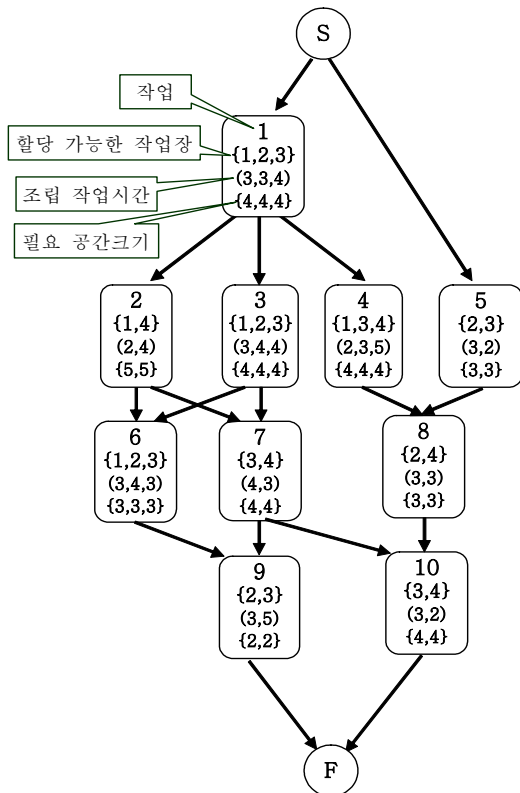


그림 3. 공정계획을 위한 네트워크.

2.3 FAS 공정계획문제

앞에서 언급하였듯이, 본 연구에서 다루는 공정계획은 각 제품의 작업들을 작업장에 할당하고, 각 제품의 조립경로를 결정하는 문제이다. 다루는 문제의 영역과 그 특징은 2.1절과 2.2절에서 이미 언급하였다. 공정계획의 목적은 모든 조립제품의 총 이동시간의 최소화로 둔다. FAS에서는 흔히 작업처리시간이 짧고, 작업장 간 이동의 무방향성으로 인해 조립품의 이동시간이 많이 소요되는 경향이 있다. 따라서 조립품의 이동시간은 시스템의 수행도 평가에 중요한 요소가 될 수 있다. 한편, 이동시간 최소화는 작업장들 사이에 부하의 불균형을 초래할 수 있다. 이는 작업장부하의 상한을 두어 해결하였다.

본 연구에서 다루는 공정계획은 작업의 선행 제약, 작업장의

공간 제약과 작업부하 상한 제약이 주어진 조건 아래서 작업장 간 이동을 최소로 하는 각 제품의 작업할당과 조립경로를 동시에 결정하는 문제이다. 본 연구에서는 다음과 같은 가정 및 제약을 둔다.

1. 각 조립작업은 적어도 하나의 작업장에 할당되어야 한다.
2. 동일 작업이 제품에 따라 다른 작업장에 할당될 수 있다 (중복할당 가능).
3. 작업장에 할당된 작업들의 공간용량의 합은 그 작업장의 공간용량을 넘을 수 없다.
4. 조립품의 작업장 간 이동시간은 주어져 있다.
5. 각 작업의 조립시간에는 기계고장 또는 작업불량으로 인한 평균소요시간, 준비시간, 장착과 탈착시간, 공구교환시간이 포함되어 있다.
6. 조립라인의 조립 기계와 장비, 자동 자재처리설비 등은 이미 결정되어 있다.

본 연구에서 다루는 문제와 유사한 기존 연구를 살펴보자. Ammons *et al.*(1985)은 FAS의 작업부하 평활화와 작업들의 작업장 방문횟수의 두 목적을 일차 결합한 다목적 문제를 다루었다. 그러나 이 연구에서는 작업장의 공간용량을 고려하지 않고 있다. Lucertini *et al.*(1996)은 부품의 작업장 간 이동횟수를 최소화하는 문제에 대해 작업장부하를 고려하지 않은 유형과 작업장부하를 고려한 유형으로 나누어 문제를 다루었다. Sawik (1999, 2000)은 최대 작업장부하량과 최대 이동시간의 최소화를 목적으로 두고, 작업할당과 조립경로를 동시에 결정하는 정수계획모형을 제시하였다. 그러나 규모가 큰 문제에서 정수계획모형은 많은 변수와 제약식으로 인하여 구현하는 데 한계가 있다. 또한 가능 조립경로는 네트워크 상태로 주어진 것이 아니라 한정된 몇 개의 경로만을 대상으로 하고 있다. Ma & Matsui(2002)는 FAS의 작업할당문제에서 각 기계의 조립시간이 확률분포를 따르는 경우를 다루고 있다. 이들 연구(Ammons *et al.*, 1985; Lucertini *et al.*, 1996; Sawik, 1999, 2000; Ma & Matsui, 2002)에서는 각 작업을 행할 수 있는 작업장이 제한되지 않고, 모든 작업장에서 가능한 것으로 가정하고 있다. 한편, Park & Kim(1995)은 FAS에서 작업장의 공간용량을 고려한 부품선택 문제와 작업할당문제를, Khouja *et al.*(2000)은 조립 로봇의 성능을 고려한 작업할당문제를 다루었다. 그러나 이들 연구(Park & Kim, 1995; Khouja *et al.*, 2000)는 조립순서문제를 고려하지 않고 있다.

3. FAS 공정계획을 위한 공생 진화알고리즘

3.1 공생 진화알고리즘의 구조

생물계에서 공생관계에 있는 여러 종(species)들은 상호작용하고 협조하면서 진화한다. 공생 진화알고리즘은 이러한 과정

을 모방한 기법으로, 같은 종류의 개체집단을 모집단으로 나타내고, 종의 수와 같은 수의 모집단을 운영한다. 이 때 각 모집단은 해결하고자 하는 문제를 구성하는 부분문제를 나타낸다.

FAS에서 각 제품의 작업할당과 조립경로는 상호 영향을 미친다. 본 연구에서 다루는 FAS 공정계획은 크게 작업할당과 조립경로 결정의 두 부분문제로 구성된다. 작업할당문제에서 개체표현은 모든 제품의 작업할당을 하나의 개체에 나타내는 형태로서, 이 문제를 위해 하나의 모집단으로 운영한다. 한편, 경로결정문제는 제품별로 모집단을 운영한다. 작업할당문제에서 제품별로 모집단을 운영하면 공간제약을 만족시키기 쉽다. 이로 인해 비효율적인 탐색이 될 수 있다. 그러나 경로문제의 제품별 모집단운영은 제약을 처리하는 데 큰 어려움 없이 넓은 해공간을 다양하게 탐색할 수 있다는 장점을 갖는다.

본 연구에서 제안하는 공생 진화알고리즘은 <그림 4>와 같은 2계층 구조를 갖는다. 계층 1은 제품 1, ..., n의 작업할당을 하나의 모집단, L_Pop 과 n개 제품의 조립경로 모집단, R_Pop_1, \dots, R_Pop_n 로 구성되고, 계층 2는 계층 1에 있는 모든 부분모집단을 결합한 하나의 통합모집단, Int_Pop 으로 구성된다. 제안한 알고리즘을 2계층 공생 진화알고리즘(2-Leveled symbiotic Evolutionary Algorithm: 2-LEA)으로 명명한다.

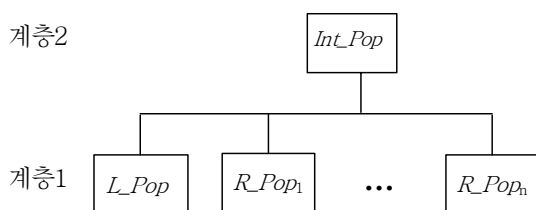


그림 4. FAS 공정계획문제를 위한 2계층 구조.

3.2 제안된 2-LEA의 개념

2-LEA를 설명하는 데 사용되는 기호를 먼저 정의하자. $Pop[l, q]$ 는 계층 $l(=1, 2)$ 의 q 번째 모집단을 나타낸다. 이 때, 각 모집단은 토러스(torus) 형태의 2차원 정방형 격자구조를 갖는다고 본다. $NP_{ij}[l, q]$ 는 모집단 $Pop[l, q]$ 에서 위치 (i, j) 를 중심으로 하는 (3×3) 구조의 이웃을 나타낸다. <그림 5>는 이들 구조와 형태를 보여 주고 있다.

계층 1에서 $Pop[1, 1]$ 은 L_Pop 의 모집단을, $Pop[1, 2], Pop[1, 3], \dots, Pop[1, n+1]$ 은 차례로 제품 1부터 제품 n까지의 조립경로 모집단, $R_Pop_1, R_Pop_2, \dots, R_Pop_n$ 을 각각 나타낸다. $Pop[2, 1]$ 에 있는 개체는 계층 1의 모집단 $Pop[1, 1], Pop[1, 2], \dots, Pop[1, n+1]$ 에 있는 각 개체들의 결합에 의해 생성된다. 2-LEA의 개념과 그 특징은 다음과 같다.

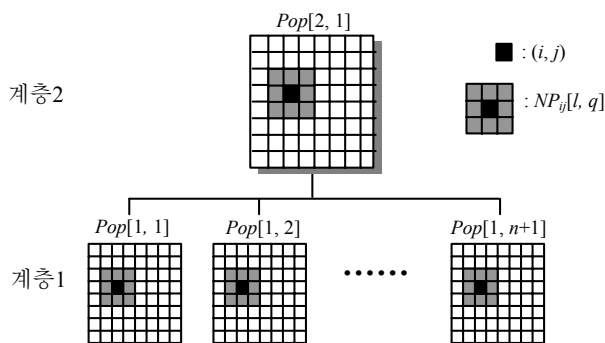


그림 5. 2-LEA의 구조.

첫째, 계층 1에서는 분리된 종들이 공생하는 과정을 모방한 공생 진화알고리즘이 행하여 진다. 각 모집단에 있는 개체의 적응도(fitness)는 다른 모집단들의 개체가 참여(공생)하여 하나의 전체해(complete solution)를 구성하여 평가된다. 이 때, 개체의 적응도에 참여하는 다른 모집단의 개체를 공생 파트너라 부른다(Kim et al., 2003). 공생 파트너 선택전략은 3.3절에서 다룬다. 이렇게 평가된 적응도에 기초하여 각 종(모집단)들은 진화한다.

둘째, 계층 2에 있는 모집단의 진화는 내공생진화(endosymbiotic evolution) 과정(Margulis, 1981)을 모방하고 있다. 계층 1에 있는 개체를 공생자(symbiont)로 보고, 계층 2는 이들이 결합한 상태로 생존하는 내공생자(endosymbiont)로 본다. 계층 1에서의 분리된 공생진화는 병렬탐색을 강화하여 넓은 해공간의 탐색을 강조할 수 있는 반면, 계층 2에서의 결합된 형태의 내공생진화는 유망한 좁은 해공간의 집중탐색에 기여할 수 있다(Kim et al., 2001).

셋째, 계층 1과 계층 2는 상호작용한다. 이 과정은 다음과 같다. 각 모집단, $Pop[1, q], q=1, 2, \dots, n+1$ 의 이웃 $NP_{ij}[1, q]$ 의 적응도 평가과정에서 발견된 가장 높은 적응도를 갖는 개체결합을 잠재내공생자(potential endosymbiont), PE_q 로 두고 그 적응도를 $f(PE_q)$ 로 나타내자. 또한 $NP_{ij}[2, 1]$ 에서 가장 낮은 적응도를 갖는 개체를 E_{old} 로 두자. $f(PE_q) > f(E_{old})$ 이면, E_{old} 를 PE_q 로 대체한다. 그렇지 않으면, 이 과정은 일어나지 않고, 잠재내공생자를 구성하는 개체들은 분리된 상태로 현재의 위치에 생존한다. 이 과정을 통해 적응도가 높은 결합된 개체는 결합된 상태로 자신의 상위 모집단으로 이동하고, 진화가 계속되는 동안 공생 파트너는 변화되지 않는다. 공생 파트너의 잦은 변동은 알고리즘에서 주어진 방향에서의 충분한 탐색을 방해할 수 있다. 이러한 과정은 내공생 진화과정을 모방한 것이다.

넷째, 이웃진화전략을 사용한다. 이웃진화전략은 다양한 종은 개체들의 적소(niche)를 형성하도록 함으로써, 모집단의 다양성 유지를 돕는다(Kim et al., 2000; Kim et al., 2003). 이는 결과적으로 해의 조기 수렴을 줄일 수 있어, 진화알고리즘의 장기적인 성공에 기여한다.

3.3 2-LEA의 절차

2-LEA의 절차는 다음과 같다.

단계 1: 초기화 및 초기 적응도 평가

각 계층에 있는 모든 모집단을 초기화한다. 개체의 초기 적응도는 각 계층별로 모집단의 같은 위치에 있는 개체들을 서로 짝지어 평가하고, 가장 높은 적응도를 f_{best} 로 둔다.

단계 2: 이웃 설정

임의의 위치 (i, j) 를 선택하고, 이웃 $NP_{ij}[1, q], q=1, 2, \dots, n+1$ 와 $NP_{ij}[2, 1]$ 을 설정한다.

단계 3: 적응도 평가 및 계층 간 상호작용

단계 3.1: 계층1의 각 이웃 $NP_{ij}[1, q], q=1, 2, \dots, n+1$ 에 대해,

- (a) 개체들의 적응도를 평가한다.
- (b) 이웃에서 적응도가 가장 높은 잠재내공생자, PE_q 를 구한다. 그리고 $f(PE_q)$ 와 f_{best} 를 비교하여 f_{best} 를 갱신한다.
- (c) $NP_{ij}[2, 1]$ 에서 가장 낮은 적응도를 갖는 개체를 E_{old} 라 하자. $f(PE_q) > f(E_{old})$ 이면, E_{old} 를 PE_q 로 대체한다.

단계 3.2: $NP_{ij}[2, 1]$ 에 있는 개체들의 적응도를 구한다.

단계 4: 이웃의 진화

각 이웃 $NP_{ij}[1, q], q=1, 2, \dots, n+1$ 와 $NP_{ij}[2, 1]$ 에 대해,

- 단계 4.1: 적응도를 기준으로 두 부모 개체를 선택하여 교차한 후 두 자손 개체를 생산한다.
- 단계 4.2: 이웃에서 적응도가 가장 낮은 두 개체를 단계 4.1에서 생산한 자손 개체와 대체한다.
- 단계 4.3: 이웃 개체들에 대하여 돌연변이율에 따라 돌연변이시킨다.

단계 5: 종료조건

종료조건을 만족하면 끝내고, 그렇지 않으면 단계 2로 간다.

단계 3.1에서는 공생 파트너 선택전략이 요구된다. 공생파트너 선택전략은 다양할 수 있으나(Bull and Fogarty, 1995), 이들 전략들은 공생 진화알고리즘 성능에 대한 효과에 유의한 차이를 보이지 않는 것으로 나타났다(Kim et al., 2000). 따라서 본 연구에서 공생 파트너는 알고리즘의 계산시간과 구현의 용이성 관점에서 동일 계층에 있는 다른 모집단의 같은 이웃에서 임의의 한 개체를 선택하는 전략을 사용한다.

단계 4에서 이웃진화는 안정상태(steady-state) 유전알고리즘을 채용하였다. 안정상태 유전알고리즘은 가장 좋은 개체가 보존되도록 한다. 또한 높은 적응도를 갖는 개체는 생산되는 즉시 재생산에 참여하므로, 이 개체의 유전정보를 이용하여 좋은 해의 이용을 높일 수 있다(Kim et al., 2003; Syswerda, 1991).

4. FAS의 공정계획을 위한 유전요소

2-LEA를 구현하기 위해서는 각 모집단의 개체표현, 적응도 평가방법, 그리고 진화를 위한 유전연산방법이 필요하다. 이들 유전요소에 대해서는 다음 장에서 다룬다. 먼저, 사용되는 기호를 정의하자.

i : 조립작업의 첨자, $i=1, 2, \dots, l$.

j : 작업장의 첨자, $j=1, 2, \dots, m$.

k : 제품의 첨자, $k=1, 2, \dots, n$.

$I(j)$: 작업장 j 에 할당된 작업들의 집합.

a_{ij} : 작업장 j 에 작업 i 를 할당할 때 요구되는 공간용량.

b_j : 작업장 j 에 할당된 작업들의 총 소요 공간용량, 즉

$$b_j = \sum_{i \in I(j)} a_{ij}$$

B_j : 작업장 j 의 최대 허용 공간용량.

W_j : 작업장 j 에 할당된 조립작업 부하량(조립작업시간의 합).

t_{ik} : 제품 k 에서 작업 i 의 조립작업시간.

$AS(i_k)$: 제품 k 의 작업 i 가 할당 가능한 작업장 집합. 본 연구에서는 $AS(i_k) = AS(i), \forall k$ 로 둔다.

$FJ(i_k)$: 제품 k 에서 작업 i 의 직선행과 직후행 작업들이 할당된 작업장 집합.

WB : 작업장의 최대 부하량.

4.1 유전표현과 초기 모집단

계층 1에는 작업할당 모집단, L_Pop 과 조립경로 모집단, $R_Pop_k, k=1, 2, \dots, n$ 이 있다. L_Pop 에 있는 개체는 행렬 $(n \times l)$ 형태로 표현된다. 행렬에서 행은 제품을, 열은 작업을 나타낸다. 위치 (k, i) 에 있는 인자값은 제품 k 의 작업 i 가 할당되는 작업장을 나타낸다. 이는 일종의 그룹 번호표현으로, 이 표현은 해석(decoding)이 용이하고 중복표현이 되지 않는다는 장점이 있다. <그림 6>은 한 예를 보여 주고 있다. 여기서 인자값이 0인 요소는 해당 제품의 작업이 행하여지지 않음을 의미한다. 예로, 제품 1에서 작업 2는 행하여지지 않는다.

	i	1	2	3	4	5	6	7	8	9	10
k											
1		1	0	1	3	2	2	3	2	2	4
2		3	1	2	0	2	0	3	2	3	4
3		1	1	1	3	0	2	4	0	3	1

그림 6. 작업할당 모집단을 위한 개체 표현.

L_Pop 의 초기 모집단은 가능한 제약을 만족하는 초기 개체가 생산될 수 있도록 하기 위하여 작업장의 공간용량을 우선적으로 고려한다. 이를 위하여 중복할당을 허용하지 않는다.

즉, 모든 제품에서 동일 작업은 같은 작업장에 할당한다. 그 절차는 아래와 같다.

<작업할당 초기 모집단의 개체 생성절차>

- 단계 1: 작업들의 집합 I 를 생성한다. $b_j=0, \forall j$ 로 둔다.
- 단계 2: 집합 I 에서 임의로 작업 i^* 를 선택한다.
- 단계 3: $j^* = \arg \max_{j \in AS(i^*)} \{B_j - b_j\}$ 인 작업장 j^* 를 모든 제품의 작업 i^* 에 할당한다. j^* 가 여러 개 존재하면 임의로 선택한다.
- 단계 4: $b_j \leftarrow b_j + a_{i^*j^*}$ 로 갱신하고, 집합 I 에서 i^* 를 삭제한다. $I = \emptyset$ 이면 종료하고, 그렇지 않으면 단계2로 간다.

조립경로 모집단의 개체는 순열표현으로 나타낸다. 즉, 제품이 조립되는 작업순서를 차례로 나타낸다. 이 때, 제약은 조립의 선행관계이다. 예로 (1, 2, 4, 5, 3, 6, 8, 7, 9, 10)은 제품이 작업 1 → 2 → 4 → 5 → 3 → 6 → 8 → 7 → 9 → 10의 순서로 조립됨을 의미한다. 본 연구에서는 조립경로를 위해 제품 수, n 개의 모집단을 운용함으로써 모든 제품이 하나의 조립경로를 따르지 않고, 제품마다 다른 조립순서를 가진다. 초기 모집단은 아래와 같은 절차에 의해 생성된다.

<조립경로 초기 모집단의 개체 생성절차>

- 단계 1: 선행 작업이 없는 작업들의 집합 B 를 만들고, 비어 있는 순서열을 만든다.
- 단계 2: 만약 B 가 비어 있으면 종료하고, 그렇지 않으면 단계 3으로 간다.
- 단계 3: 집합 B 에서 임의로 한 작업을 선택하여 순서열의 왼편부터 차례로 삽입한다.
- 단계 4: 선택된 작업을 집합 B 에서 삭제하고, 선행 작업이 모두 할당된 작업으로 집합 B 를 갱신한다. 단계 2로 간다.

각 모집단에 대해, 모집단 크기만큼 위 과정을 반복하여 초기 모집단을 구성한다. 위 절차에서, 모집단의 다양성은 단계 3에서 임의로 작업을 선택함에 의해, 그리고 선행제약은 단계 1과 4에서 고려되고 있다.

4.2 적응도 평가함수

흔히, 다루는 문제의 목적함수 자체가 적응도 평가함수로 사용된다. 그러나 본 연구에서 다루는 문제는 여러 제약들을 포함하고 있다. 제약의 종류가 많은 경우, 모든 종류의 제약을 만족하는 유전표현과 유전연산자를 개발하기가 쉽지 않다. 본 연구에서는 제약을 개체표현에 의해 다룰 수 있는 직접제약과 그렇지 않은 간접제약으로 나누었다. 조립 선행제약과 작업의 할당 가능 작업장제약은 직접제약으로 두고, 이들 제약은 개

체표현과 유전연산에서 항상 만족되도록 하였다. 그리고 공간 제약과 작업장 작업부하 상한제약은 간접제약으로 두고, 이들 제약을 어길 시에는 평가함수에 벌금(penalty)을 부과하는 방법을 사용하였다.

2.3절에서 언급했듯이, 공정계획의 목적은 조립제품의 작업장 간 이동시간의 최소화이다. 개체의 평가결과 나타난 총 이동시간을 Q_{sum} 으로 둔다. 계층 1에 있는 개체의 적응도는 동일 계층의 다른 모집단에서 공생 파트너를 선택하여, 이들을 결합하여 평가된다. 계층 2는 하나의 모집단만 존재하므로, 여기에 있는 개체들의 적응도는 공생 파트너 없이 평가된다. 개체의 적응도 평가함수 $eval$ 는 다음과 같다.

$$\text{Minimize } eval = Q_{sum} + c_1 \sum_j PS_j^\alpha + c_2 \sum_j PW_j^\beta \quad (1)$$

여기서, $PS_j = \max(b_j - B_j, 0)$ 이고, $PW_j = \max(W_j - WB, 0)$ 이며, 상수 c_1, c_2, α, β 는 파라미터이다.

4.3 유전연산자

유전연산자는 흔히 교차(crossover)와 돌연변이(mutation)로 나눌 수 있다. 진화알고리즘에서 교차는 부모가 갖는 정보를 자손에 전달하는 역할을 하고, 돌연변이는 모집단의 다양성을 유지하는 역할을 한다. 특히, 교차연산자를 설계할 때, 적용문제가 가지고 있는 의미 있는 유전정보가 교차에 의해 파괴되지 않도록 주의하여야 한다.

작업할당 모집단(L_Pop)에서 교차는 수정구조교차(Kim *et al.*, 1997)를 사용하며. 그 절차는 다음과 같다.

- 단계 1: 범위 $[1, m]$ 에서 임의의 정수 r 를 선택한다.
- 단계 2: 부모 P1에서 인자값 0부터 r 까지 표현한 인자를 자손 O1의 동일한 위치에 상속한다.
- 단계 3: 자손 O1의 남은 위치에 부모 P2의 동일 위치에 인자값이 $r+1$ 부터 m 까지 표현된 인자를 상속한다.
- 단계 4: 자손 O1에 비어 있는 위치는 재할당방법을 이용한다. 이 방법은 발견적 기법으로 아래에서 설명한다.

P1	1	0	2	3	3	2	4	2	2	3
	2	1	3	0	2	0	3	4	2	4
	1	4	2	3	0	2	4	0	3	4
	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
O1	1	0	2	*	*	2	3	2	2	4
	2	1	4	0	2	0	3	4	3	3
	1	3	2	*	0	2	4	0	3	4
	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑
P2	2	0	2	1	2	2	3	4	3	4
	2	1	4	0	2	0	3	4	3	3
	3	3	4	1	0	2	4	0	3	4

그림 7. 작업할당 모집단을 위한 교차.

자손 O2는 두 부모의 역할을 바꾸어 생산한다. 이 구조교차는 두 부모가 가지고 있는 위치와 인자값 정보를 자손에게 상속한다는 특징을 갖는다. <그림 7>은 이러한 구조교차를 보이고 있다. 여기서 $m=4, r=2$ 로 두었으면, * 표시는 미할당인자를 나타낸다. 이들 인자의 할당을 위한 재할당기법은 4.4절에서 다룬다. O1에서 음영으로 표시된 인자는 P2로부터 상속 받은 것이다.

작업할당 모집단의 돌연변이는 돌연변이율에 따라 각 인자에 해당하는 작업을 행할 수 있는 작업장 중에서 임의로 선택하는 방법으로 수행한다.

다음은, 조립경로 모집단($R_Pop_k, k=1, 2, \dots, n$)의 유전연산에 관하여 다룬다. 이들 모집단에 있는 개체의 표현은 순열표현임을 상기하자. 이들 모집단의 교차는 수정이점교차(Kim et al., 1997)를 사용한다. 그 절차는 다음과 같다. 먼저 임의의 절단점을 정한다. 두 절단점에 의하여 나누어진 앞부분, 중간부분, 뒷부분에 있는 인자 중 앞부분과 뒷부분은 한 부모(P1)로부터 자손(O1)의 같은 위치에 복사한다. 그리고 다른 부모(P2)에서 복사된 인자값을 갖는 인자들을 지운다. P2에 남아 있는 인자들을 개체에서 나타나는 순서대로 O1의 중간 부분에 복사한다. <그림 8>은 이를 잘 보여 주고 있다. 자손 O2는 두 부모의 역할을 바꾸어 생산한다. 이 교차방법은 작업의 선행관계를 만족하고(Kim et al., 1997), 한 부모로부터는 위치와 순서정보를, 다른 부모로부터는 상대적 순서정보를 상속받는다.

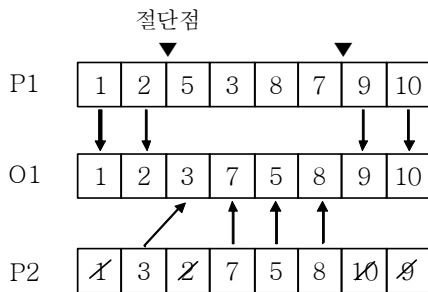


그림 8. 조립경로 모집단을 위한 교차연산.

조립경로 모집단에서 돌연변이는 수정된 삽입방법을 사용한다. 이는 돌연변이율에 의해 선택된 인자(작업)가 이동할 수 있는 범위 내의 임의의 위치에 삽입하는 방법이다. 여기서 이 범위는 작업의 선행관계에 의해 구해진다. 즉, 선행제약을 어기지 않는 임의의 위치에 삽입한다.

4.4 재할당기법

제안된 수정구조교차에서 인자값이 결정되지 않는 인자(미할당작업)들에 그 값을 부여하는 과정이 필요하다. 이를 위하여 본 연구에서 제안하는 발견적 기법은 아래와 같다. 여기서 C는 *로 표시된 미할당된 인자의 집합을 나타낸다. 위치 (k, i)

는 제품 k 의 작업 i 를 나타내고, 기호 i_k 로 표시한다. 그리고

$$\bar{a}_{ij} = \begin{cases} a_{ij}, & \text{if } i \in I(j) \\ 0, & \text{그 밖 에} \end{cases}$$

로 정의한다.

단계 1: $AS(i_k)$ 와 $FJ(i_k), i_k \in C$ 를 구하고, b_j 와 $W_j, \forall j$ 를 계산한다.

단계 2: 집합 C에서 임의로 i_k^* 를 선택한다.

단계 3: $R = \{j | b_j + \bar{a}_{i_k^*j} \leq B_j \text{ and } W_j + t_{i_k^*} \leq WB, j \in AS(i_k^*)\}$ 로 둔다. $R = \emptyset$ 이면, 단계 4로 간다. 그렇지 않으면서, (1) $R \cap FJ(i_k^*) \neq \emptyset$ 이면, $j^* = \arg \min_{j \in (R \cap FJ(i_k^*))} W_j$ 이고, (2) $R \cap FJ(i_k^*) = \emptyset$ 이면, $j^* = \arg \min_{j \in R} W_j$ 이다. 그리고 단계 6으로 간다.

단계 4: $R1 = \{j | b_j + \bar{a}_{i_k^*j} \leq B_j \text{ or } W_j + t_{i_k^*} \leq WB, j \in AS(i_k^*)\}$ 로 둔다. $R1 = \emptyset$ 이면, 단계 5로 간다. 그렇지 않으면서, (1) $R1 \cap FJ(i_k^*) \neq \emptyset$ 이면, $R1 \cap FJ(i_k^*)$ 에서 임의로 j^* 를 선택하고, (2) $R1 \cap FJ(i_k^*) = \emptyset$ 이면, $R1$ 에서 임의로 j^* 를 선택한다. 그리고 단계 6으로 간다.

단계 5: $AS(i_k^*) \cap FJ(i_k^*) \neq \emptyset$ 이면, $AS(i_k^*) \cap FJ(i_k^*)$ 에서, 그렇지 않으면 $AS(i_k^*)$ 에서 임의로 j^* 를 선택한다.

단계 6: 작업 i_k^* (즉 위치 (k, i))에 작업장(인자값) j^* 를 할당한다. 그리고 작업 i_k^* 를 집합 C에서 삭제한다. $C = \emptyset$ 이면 종료하고, 그렇지 않으면 b_{j^*}, W_{j^*} 를 갱신하고 단계 2로 간다.

위 절차에서, 좋은 가능해의 생산을 유도하기 위하여 최적화의 목적과 함께 제약을 고려하고 있다. 작업할당에서 공간용량 제약과 작업부하 제약을 동시에 만족하는 작업장이 존재하면, 이동시간을 줄일 수 있는(직선행 또는 직후행 작업이 할당된) 작업장에 할당(단계 3에서)한다. 이들 두 제약 중에 하나의 제약만을 만족하는 작업장이 존재하는 경우, 이들 작업장 중에서 이동시간을 줄일 수 있는 작업장에 할당한다(단계 4 참조). 이 두 제약을 만족하는 작업장이 존재하지 않는 경우에는 임의로 할당한다(단계 5 참조).

4.5 계층 2의 유전요소

계층 2에 있는 개체는, 3장에서 언급하였듯이, 계층 1에 있는 각 모집단의 개체를 결합한 내공생자이다. 따라서 개체표현은 4.1절에서 다룬 개체표현을 결합하여 나타낸다. 초기 모집단의 개체 또한 4.1절의 작업할당과 조립경로의 초기 모집단의 개체 생성절차에 의해 각각 개체를 구하여 이를 결합함으로써 구성한다. 그리고 유전연산은 계층 1의 각 개체에 해당하는 부분별로 4.3절과 4.4절에서와 같은 방법으로 행한다.

5. 실험설계 및 결과분석

5.1 실험설계

조사한 바에 의하면 본 연구에서 다루고 있는 공정계획문제는 아직까지 연구되지 않고 있으며, 이를 해결하기 위한 기법도 존재하지 않는다. 따라서 비교기법은 진화알고리즘으로 국한하였다. 비교알고리즘은 두 종류로, 그 하나는 전통적 진화알고리즘(Traditional Evolutionary Algorithm; TEA)이고, 다른 하나는 단일 계층으로 구성된 공생 진화알고리즘(Symbiotic Evolutionary Algorithm; SEA)이다. TEA는 하나의 개체에 모든 정보를 표현하여, 이들로 구성된 단일 모집단을 운영하는 알고리즘으로, 2-LEA에서 계층 2만을 운영하는 것과 같다. 두 번째 기법인 SEA는 Kim *et al.*(2002)에 의해 제안된 기법으로, 부분문제마다 모집단을 두어 복수 모집단이 운용되지만, 이들은 단일 계층에서 공생진화한다. 이는 2-LEA에서 계층 1에 있는 종들이, 계층 2와의 상호작용이나 내공생과정 없이, 분리된 형태로 공진화하는 과정과 같다. 이에 대한 자세한 설명은 Kim *et al.*(2002)의 연구를 참조할 수 있다.

실험문제는 19개의 작업을 갖는 Thomopoulos문제(Thomopoulos, 1967), 45개 작업을 가진 Kilbridge와 Wester문제(Kilbridge and Wester, 1964)와 작업이 61개인 ABC문제(Kim and Kwak, 1993)의 결합선행공정도를 이용하였다. 각 작업의 조립시간과 공간용량은 임의로 추출하였으며, 조립시간은 조립을 행하는 작업장에 따라 달리 두었다. PB12~PB14은 PB09~PB11과 같은 상황에서 작업장의 공간용량을 다르게 하였다. 이들 세 문제를 제품 수와 작업장 수에 의해 <표 1>과 같이 구성하였다. 이에 대한 구체적인 것은 Euy(2004)를 참고할 수 있다.

표 1. 실험문제

문 제	작업 수	제품 수	작업장 수	
Thomopoulos	PB01	19	3	5
	PB02	19	3	4
	PB03	19	4	5
	PB04	19	4	4
Kilbridge and Wester	PB05	45	3	4
	PB06	45	4	5
	PB07	45	5	6
	PB08	45	5	6
ABC	PB09	61	4	6
	PB10	61	5	6
	PB11	61	6	6
	PB12	61	4	6
	PB13	61	5	6
	PB14	61	6	6

모든 알고리즘은 JAVA 언어로 구현되었으며, 2.0GHz Pentium CPU를 장착한 IBM-PC에서 수행되었다. 유전 파라미터들은 예비실험을 통하여 결정하였다. 첫째, 모집단의 크기는 100(10×10의 격자구조)으로 하였다. 둘째, 식 (1)에서 사용된 파라미터는 $c1 = c2 = 10$, $\alpha = \beta = 0.5$ 로 두었다. 셋째, 돌연변이율은 모든 모집단에서 개체돌연변이율 = 0.03, 인자돌연변이율 = 0.3을 사용하였다. 끝으로, 종료조건은 개체 평가횟수를 사용하였고, 해공간의 차이를 반영하기 위하여, Thomopoulos문제(PB01~04)는 3.0E+04개, Kilbridge와 Wester문제(PB05~PB08)는 8.0E+04개, ABC문제(PB09~PB14)는 1.0E+05개를 종료조건으

표 2. 알고리즘 성능비교

Problem	TEA			SEA			2-LEA			Improved rate(%)
	Best	Mean	Std	Best	Mean	Std	Best	Mean	Std	
PB01	22.0	26.9	2.8	22.0	24.8	2.7	22.0	24.5	1.9	8.9
PB02	32.0	37.4	3.9	30.0	36.5	10.6	30.0	34.4	3.3	8.0
PB03	38.0	46.1	5.1	36.0	41.1	3.4	32.0	39.8	4.6	13.7
PB04	54.0	64.8	9.0	54.0	64.3	6.7	48.0	61.1	6.6	5.7
PB05	99.0	113.7	8.0	88.0	112.9	16.2	89.0	110.7	13.5	2.6
PB06	220.0	253.1	17.2	223.0	240.4	11.8	212.0	230.8	13.2	8.8
PB07	105.0	434.7	34.8	375.0	399.3	27.5	332.0	343.3	9.9	21.0
PB08	556.0	561.5	7.8	524.0	527.0	4.2	472.0	482.7	9.5	14.0
PB09	468.0	463.4	17.2	420.0	446.0	22.0	392.0	427.0	18.5	7.9
PB10	688.0	694.0	8.5	680.0	685.0	7.1	654.0	663.0	12.7	4.5
PB11	854.0	888.0	37.4	828.0	844.0	22.6	770.0	824.6	21.8	7.1
PB12	406.0	429.7	14.9	396.0	414.0	12.4	380.0	403.2	12.5	6.2
PB13	589.0	608.8	16.3	562.0	598.0	23.9	528.0	561.8	22.3	7.7
PB14	812.0	830.0	25.5	766.0	789.0	20.2	714.0	732.5	15.8	11.7

로 두었다. 그리고 비교 알고리즘에서 사용된 진화전략, 유전 파라미터 및 유전요소들은 2-LEA의 파라미터 및 요소들과 동등한 조건으로 설정하였다.

5.2 실험 결과 및 분석

탐색능력의 평가를 위하여, 2-LEA는 5.1절에서 설명한 두 알고리즘과 함께 비교하였다. 이 실험에서 총 이동시간의 최소화를 목적으로 사용하였으며, 각 실험문제마다 25회 반복실험을 하였다. <표 2>는 실험결과를 보인 것이다. 각 알고리즘에 대해 문제들의 가장 좋은 값(Best)과 평균(Mean), 그리고 표준편차(Std)를 구하였다. 개선율(Improved rate) = $\{(TEA의\ 평균 - 2-LEA의\ 평균) / TEA의\ 평균\} \times 100\%$ 이다

실험결과, 2-LEA가 다른 알고리즘에 비해 모든 문제에서 우수한 성능을 보였다. 그리고 SEA가 그 다음으로 좋은 결과를 보였으며, TEA가 가장 낮은 성능을 보였다. 이 결과로부터, TEA에 사용된 단일 모집단의 운용과 다중 염색체 표현방법은 부분문제가 결합된 통합문제의 해결에 효과적이지 못하다는 것을 알 수 있다. 즉, 2-LEA와 SEA처럼 여러 개의 부분해들의 약한 결합이 부분해들의 강한 결합보다 탐색성능에서 우수함을 알 수 있다. 또한, 2-LEA가 SEA보다 우수한 성능을 보였다. 이는 2-LEA가 내공생 진화과정을 모방하여 분리된 해들이 존재하는 모집단과 결합된 해들이 존재하는 모집단을 동시에 운영하고, 진화가 계속되는 동안 하위계층에서 발견된 좋은 해들이 내공생자로 만들어져 상위계층으로 이동하는 진화전략과 적절한 모집단 계층구조가 모집단의 다양성 및 해의 질 향상에 기여하기 때문으로 본다.

<그림 9>는 실험문제 PB12의 진화과정을 보인 그림이다. 이 그림에서 알 수 있듯이 2-LEA가 다른 비교 알고리즘보다 진화 초기부터 계속하여 우수한 성능을 보인다. 따라서 2-LEA는 적용문제에 따라 요구되는 실시간에 맞추어 종료조건을 유연하게 설정할 수 있다는 장점을 갖는다.

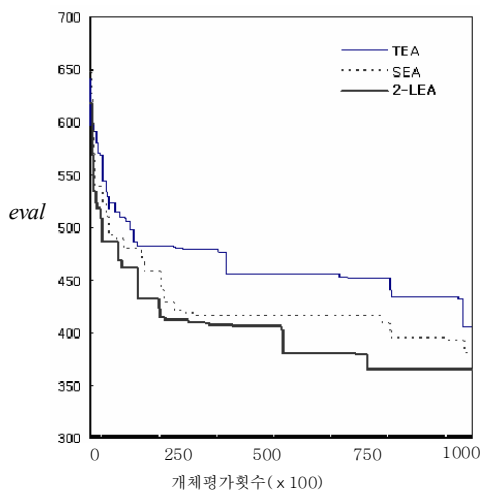


그림 9. 진화알고리즘의 수렴속도 비교(PB12).

한편, 컴퓨터 계산소요시간은 2-LEA에서 평균적으로 PB01~PB04가 3.6초, PB05~PB08이 6.7초, PB09~PB14가 10초 정도 소요되었다. SEA는 2-LEA보다 약 10% 정도 적게 소요되었으며, TEA는 SEA보다 약 8% 정도 적게 소요되는 것으로 나타났다. 이는 SEA의 공생 파트너 선택전략과 2-LEA의 내공생자 생성과 운영에 의한 것으로 보인다.

6. 요약 및 결론

본 연구에서는 유연조립시스템의 공정계획을 다루었다. 이를 위하여, 먼저 여러 형태의 유연조립시스템의 구조와 그 특징을 파악하였다. 그리고 유연조립시스템에서 상호연관성이 높은 작업할당문제와 조립경로문제를 동시에 통합적으로 다루었다. 이들 두 문제의 통합적 접근방법론으로 공생 진화알고리즘이 사용되었다. 그리고 이 문제에 효율적인 2계층 공생 진화알고리즘(2-LEA)을 제안하였다.

2-LEA는 기존의 공생 진화알고리즘에 계층구조와 내공생 진화과정을 포함시킨 알고리즘이다. 내공생진화에서 전체해를 표현한 내공생자는 부분해를 표현한 공생자와 그들의 공생파트너가 결합되어 생성된다. 내공생자의 형성과 진화는 개체의 적응력과 탐색성능 향상에 기여한다. 또한 결합된 내공생자로 구성된 모집단과 분리된 공생자로 구성된 모집단 모두에서 이웃진화전략을 바탕으로 한 안정상태 유전알고리즘을 사용하였다. 이웃진화전략은 모집단의 다양성 유지를 강조하고, 안정상태 유전알고리즘은 좋은 해의 보존과 이용을 돕는다.

제안한 알고리즘의 성능평가를 위하여, 다양한 형태의 실험문제를 생성하여 기존의 진화알고리즘과 성능을 비교분석하였다. 그 결과, 2-LEA는 비교된 알고리즘들보다 우수한 결과를 보였다. 그리고 제안된 알고리즘은 진화알고리즘이 원래 가지고 있는 적용의 유연성에 의해 목적과 제약들이 변화된 다양한 형태의 유연조립시스템의 공정계획문제에 적용될 수 있을 것으로 기대된다.

참고문헌

- Agnētis, A., Nicolo, F., Arbib, C. and Lucertini, M. (1995), Task assignment and subassembly scheduling in flexible assembly lines, *IEEE Transactions on Robotics and Automation*, 11, 1-20.
- Ammons, J.C., Lofgren, C.B. and McGinnis, L.F. (1985), A large scale machine loading problem in flexible assembly, *Annals of Operations Research*, 3, 319-332.
- Bull, L. and Fogarty, T.C. (1995), Artificial symbiogenesis, *Artificial Life*, 2, 269-292.
- Euy, J.M. (2004), Process Planning Problem of the Flexible Assembly System using Symbiotic Evolutionary Algorithm, available at <http://syslab.chonnam.ac.kr/links/FASdata.doc>.
- Groover, M.P. (2001), Automation, production systems and computer-integrated manufacturing, Prentice-Hall.

Jueung, J.J. and Ro, I.K.(1998), A study on even distribution of workloads using simulated annealing method on integrated layout design in cellular flexible assembly systems, *Journal of the Korean Operations Research and Management Science Society*, 23(4), 63-73.

Khouja, M., Booth, D.E., Suh, M. and Mahaney, J.K. (2000), Statistical procedures for task assignment and robot selection in assembly cells, *International Journal of Computer Integrated Manufacturing*, 13, 95-106.

Kilbridge, M.D. and Wester, L. (1964), A heuristic method of assembly line balancing, *Journal of Industrial Engineering*, 12.

Kim, J.Y., Kim, Y. and Kim, Y.K. (2001), An endosymbiotic evolutionary algorithm for optimization, *Applied Intelligence*, 15, 117-130.

Kim, J.Y., Kim, Y.K. and Shin, T.H. (2000), Analysis of partnering strategies in symbiotic evolutionary algorithms, *Journal of the Korean Operations Research and Management Science Society*, 25(4), 67-80.

Kim, Y.K., Kim, J.Y. and Kim, Y. (2000), A coevolutionary algorithm for balancing and sequencing in mixed model assembly lines, *Applied Intelligence*, 13, 247-258.

Kim, Y.K., Kim, J.Y. and Lee, W.K. (2003), A Multi-level Symbiotic Evolutionary Algorithm for FMS Loading Problems with Various Flexibilities, *Journal of the Korean Institute of Industrial Engineers*, 29, 65-77.

Kim, Y.K. and Kwak, J.S. (1993), Mixed model assembly line balancing with the related task consideration, *Journal of the Korean Operations Research and Management Science Society*, 18(2), 1-22.

Kim, Y.K., Park, K. and Ko, J. (2003), A symbiotic evolutionary algorithm for the integration of process planning and job shop scheduling, *Computers & Operations Research*, 30, 1151-1171.

Kim, Y.K., Yun, B.S. and Lee, S.B. (1997), Metaheuristics, Yeongji Moonhwas, Seoul, Korea.

Lee, H.F. and Johnson, R.V. (1991), A Line-Balancing strategy for designing flexible assembly systems, *The International Journal of Flexible Manufacturing Systems*, 3, 91-120.

Lee, H.F. and Steckel, K.E. (1996), An integrated design support method for flexible assembly system, *Journal of Manufacturing Systems*, 15, 13-32.

Lucertini, M., Pacciarelli, D. and Pacifici, A. (1996), Optimal flow management in flexible assembly systems: The minimal part transfer problem, *Systems Science*, 22(2), 69-80.

Ma J. and Matsui M. (2002), Performance evaluation of a flexible machining/assembly system and routing comparisons, *International Journal of Production Research*, 40(7), 1713-1724.

Margulis, L. (1981), Symbiosis in cell evolution, W.H. Freeman, San Francisco.

Matta, R.D. and Hsu, V.N. (1995), An efficient heuristic approach to recognize the infeasibility of loading problem, *The International Journal of Flexible Manufacturing System*, 9, 31-50.

Park, M.W. and Kim, Y.D. (1995), A heuristic for setting up a flexible assembly system, *International Journal of Production Research*, 33, 2295-2310.

Potter, M.A. (1997), The design and analysis of a computational model of cooperative coevolution, Ph.D. dissertation, George Mason University.

Rampersad, H.K. (1994), Integrated and simultaneous design for robotic assembly, John Wiley & Sons, Dutch.

Sawik, T. (1997), An interactive approach to bicriterion loading of a flexible assembly system, *Mathematical and Computer Modeling*, 25, 71-83.

Sawik, T. (1999), Production planning and scheduling in flexible assembly systems, Springer-Verlag, Berlin.

Sawik, T. (2000), An LP-based approach for loading and routing in a flexible assembly line, *International Journal of Production Economics*, 64, 49-58.

Syswerda, G. (1991), A study of reproduction in generational and steady-state genetic algorithm, edited by Gregory J.E, Rawlins, San Mateo, CA, 94-101.

Thomopoulos, N.T. (1967), Line balancing-sequencing for mixed-model assembly, *Management Science*, 14, 59-75.



김 여 근

서울대학교 산업공학과 학사
 서울대학교 산업공학과 석사
 서울대학교 산업공학과 박사
 현재: 전남대학교 산업공학과 교수
 관심분야: 진화알고리즘, 경영과학, 생산라인
 관리, 물류최적화



신 경 석

전남대학교 산업공학과 학사
 전남대학교 산업공학과 석사
 현재: 전남대학교 산업공학과 박사과정
 관심분야: 물류최적화, 진화알고리즘 응용



위 정 미

전남대학교 산업공학과 학사
 전남대학교 산업공학과 석사
 관심분야: SCM, FAS, 조합최적화



김 용 주

전남대학교 산업공학과 학사
 전남대학교 산업공학과 석사
 전남대학교 산업공학과 박사
 현재: (주)도올정보, 동신대학교 겸임교수
 관심분야: Web Application, e-business, 진화알고
 리즘