

다목적성을 갖는 혼합모델 조립라인의 밸런싱과 투입순서를 위한 공생 진화알고리즘*

김여근**† · 이상선**

A Symbiotic Evolutionary Algorithm for Balancing and Sequencing Mixed Model Assembly Lines with Multiple Objectives

Yeo Keun Kim** · Sang Seon Lee**

■ Abstract ■

We consider a multi-objective balancing and sequencing problem in mixed model assembly lines, which is important for an efficient use of the assembly lines. In this paper, we present a neighborhood symbiotic evolutionary algorithm to simultaneously solve the two problems of balancing and model sequencing under multiple objectives. We aim to find a set of well-distributed solutions close to the true Pareto optimal solutions for decision makers. The proposed algorithm has a two-leveled structure. At Level 1, two populations are operated : One consists of individuals each of which represents a partial solution to the balancing problem and the other consists of individuals for the sequencing problem. Level 2, which is an upper level, works one population whose individuals represent the combined entire solutions to the two problems. The process of Level 1 imitates a neighborhood symbiotic evolution and that of Level 2 simulates an endosymbiotic evolution together with an elitist strategy to promote the capability of solution search. The performance of the proposed algorithm is compared with those of the existing algorithms in convergence, diversity and computation time of nondominated solutions. The experimental results show that the proposed algorithm is superior to the compared algorithms in all the three performance measures.

Keyword : Mixed Model Assembly Lines, Balancing and Sequencing Problem, Multiobjective Evolutionary Algorithm, Symbiotic Evolutionary Algorithm, Pareto Optimal Solutions

논문접수일 : 2010년 05월 04일 논문게재확정일 : 2010년 06월 28일

논문수정일(1차 : 2010년 06월 11일)

* 이 논문은 2008년도 전남대학교 연구년교수연구비 지원에 의하여 연구되었음.

** 전남대학교 공과대학 산업공학과

† 교신저자

1. 서 론

혼합모델 조립라인(Mixed Model Assembly Line : MMAL)은 공장부지, 시설 등의 투자를 줄이고 시설의 효율적 이용을 위하여 한 라인에서 유사한 여러 모델의 제품을 생산하는 조립라인이다. MMAL은 모델별로 생산량과 총 작업시간이 다르고 부분적으로 작업내용, 작업시간, 작업방법, 자재 및 이용하는 설비와 치공구가 다를 수 있다. 이러한 라인은 오늘날 자동차, 전자 제품의 생산라인에 보편화되어 있다. MMAL에서 밸런싱 문제와 투입 순서문제는 라인의 효율적 이용을 위하여 중요하게 다루어지고 있다[26]. 밸런싱 문제는 제품 생산에 요구되는 작업들을 작업장에 할당하는 문제이고, 투입순서 문제는 모델들의 생산순서를 결정하는 문제이다. 모델의 투입순서에 대한 최적 판단은 작업할당의 결과에 의존하고, 작업할당의 결과도 투입순서에 의해 영향을 받기 때문에 이 두 문제는 서로 밀접한 관련성을 갖는다. 그러나 대부분의 기존 연구에서는 두 문제를 분리하여 독립적으로 다루었다. 두 문제를 고려한 선구적 연구로 Thomopoulos [26]와 Dar-El and Navidi[8]가 있다. 그러나 이들 연구에서는 밸런싱 문제와 투입순서 문제를 계층적으로 해결하였다. 밸런싱을 먼저 구하고 이를 입력자료로 하여 투입순서를 구하는 계층적 방법은 해공간을 충분히 탐색하는데 한계를 갖는다. 이를 극복하기 위해 Kim et al.[16]은 공진화 알고리즘을 이용하여 두 문제를 동시에 통합적으로 다루었다.

이 연구에서는 다목적성을 갖는 혼합모델 조립라인의 밸런싱 문제와 모델 투입순서 문제를 동시에 통합적으로 다룬다. 조사한 바에 의하면 MMAL에서 밸런싱과 모델 투입순서를 다룬 기존 연구에서는 단일 목적만을 다루고 있다. 그러나 MMAL의 운영과 관련된 여러 목적이 있을 수 있다. 작업 편성과 관련해서는 작업장 수 최소화, 사이클 타임 최소화, 작업장간 작업량 평활화, 작업장간 모델별 작업량 평활화 등의 목적이 있고[2, 18, 27], 모델 투입 순서

와 관련해서는 가외작업(utility work) 최소화[16], 라인 정지위험 최소화[21], 일정한 부품사용률 유지[20], 전체 라인 길이 최소화[5], 전체 준비비용 최소화[6] 등의 목적이 있다. 이들 목적 간에는 양의 상관 관계(예로, 가외작업 최소화와 전체 라인 길이 최소화)도 있으나 상충되는 경우도 발생할 수 있다(예로, 작업장간 작업량의 평활화와 가외작업 최소화). 이 연구에서는 작업장간 작업량 평활화, 가외작업 최소화, 일정한 부품 사용률 유지의 세 목적을 다룬다. 가외작업은 각 모델이 대하여 주어진 작업장 내에서 완료되지 못한 작업의 양이다.

다목적 문제에서 모든 목적에 최적인 해는 일반적으로 존재하지 않는다[1]. 이 문제를 위한 대표적인 해결 방법 중 하나는 의사결정자의 선호에 따라 하나의 해를 제시하는 것이고, 다른 하나는 의사결정자의 선호와는 상관없이 여러 좋은 대안해를 구하여 그 중에서 의사결정자가 선택하도록 하는 것이다. 선호도에 따라 가중치를 부여하여 하나의 목적함수로 변환하거나 효용함수를 이용하는 방법이 전자에 해당한다. 이 연구는 후자에 속하는 것으로 의사결정자에게 제공하기 위한 다양한 파레토(Pareto) 최적해(비지배해)를 찾는 방법을 다룬다. 여기서 파레토 최적해란 모든 목적들에 대해 이 해보다 더 우수한 해가 존재하지 않는 해이다. 앞으로 용어의 혼동을 피하기 위해 파레토 최적해는 문제가 갖는 이론적인 실제 파레토 최적해를, 비지배해는 구해진 여러 해들 중에서 지배되지 않는 해를 나타내기로 한다.

다목적 문제에서 여러 비지배해를 찾는 데 진화 알고리즘이 매우 효과적이라고 알려져 있다[25, 28]. 이는 진화알고리즘이 여러 개체(해)로 구성된 모집단을 운영하면서 병렬적으로 해공간을 탐색하기 때문이다. 다목적 문제의 해결을 위한 진화알고리즘을 다목적 진화알고리즘(Multi-Objective Evolutionary Algorithm : MOEA)이라 부른다. MOEA의 목적은 파레토 최적해 또는 이에 가까운 해이면서도 잘 분포된(다양한) 해의 집합을 찾는 것이다. MOEA는 VEGA(Vector Evaluated Genetic Algorithm)[23]가

처음으로 소개된 이래, 다양한 비지배해를 찾기 위해 여러 연구가 제안되었다. Goldberg[12]는 비지배해의 순위에 의해 적응도를 계산하는 파레토 유전 알고리즘(Pareto Genetic Algorithm : PGA)을 제안하였고, 이 개념에 해의 다양성을 강화하기 위하여 밀집도 또는 적소(niche)의 개념을 결합한 NSGA Nondominated Sorting Genetic Algorithm)[24]와 NPGA(Niched Pareto Genetic Algorithm)[13]의 알고리즘이 개발되었다. 이와 함께 파레토 최적해로의 수렴성을 강화하기 위하여 생산된 비지배해를 보존하는 ‘파레토 보존 전략’을 사용한 여러 알고리즘이 제안되었다. PAES(Pareto-Archived Evolution Strategy)[17], SPEA(Strength Pareto Evolutionary Algorithm)[29], NSGA-II[10], SPEA2[31], MSEA Multiobjective Symbiotic Evolutionary Algorithm) 3]가 이에 속한다.

특히, MSEA[3]는 전체 문제를 부분문제로 분해한 개체들이 다른 개체들과 협조하여 전체의 최적에 기여하도록 설계된 공생 진화알고리즘(symbiotic evolutionary algorithm)을 기반으로 개발되었다. 공생진화 알고리즘은 공생관계를 갖는 서로 다른 종들이 상호작용하고 상호 적응하며 공진화하는 과정을 모방한 확률적 탐색기법으로 복잡도가 높은 최적화문제의 해결에 효율적이라고 알려져 있다[14, 22]. 여기서 복잡하고 동적인 문제를 여러 부분문제로 분해하고, 이들 부분문제를 하나의 종으로 본다.

이 연구에서는 다목적성을 갖는 MMAL의 밸런싱과 모델 투입순서를 위한 공생 진화알고리즘을 제시한다. 이 연구에서 제안한 알고리즘은 2계층 구조를 가지며, 1계층에서는 공생진화과정을, 2계층에서는 Margulis[19]가 제안한 내공생 진화과정(endosymbiosis evolution)을 모방한다. 특히, 1계층에서 다양한 적소의 형성을 위하여 MSEA와는 다르게 이웃에 한정하여 진화하는 지역진화를 사용한다[9, 14]. 이 연구에서 제안한 알고리즘을 다목적 이웃 공생 진화알고리즘(Multi-objective Neighborhood Symbiotic EA : MNSEA)으로 명명한다. MNSEA의 진화 구성 요소로 다목적 문제에 효과적인 적응

도 평가와 선택 방법을 채용하고, MMAL의 밸런싱과 투입순서 문제에 적합한 표현과 유전연산자를 소개한다. 다양한 실험 문제를 가지고 구한 비지배해의 수렴성과 다양성, 그리고 계산소요시간 측면에서 제안하는 MNSEA의 성능을 분석한다.

2. 다목적성을 갖는 MMAL의 밸런싱과 투입순서 결정

2.1 혼합모델 조립라인

이 연구에서 다루는 MMAL은 다음과 같은 상황에서 운영된다고 본다. 첫째, 조립라인과 관련된 가정들이다. 직선형 컨베이어 생산라인으로 일정속도(v_c)로 이동하며, 여러 모델의 제품들이 일정시간간격(γ)으로 투입된다. 라인은 J 개의 폐쇄 작업장(closed station)으로 구성되어 있고, 각 작업장에는 한 작업자가 작업한다. 폐쇄 작업장은 각 작업장의 영역을 벗어나서 작업할 수 없는 작업장으로, 작업장 내에서 완수하지 못한 작업은 가외작업이 된다. 둘째, 조립라인은 혼류로 생산하는 사이클 생산방식을 취한다. 생산계획기간 동안에 M 종류의 모델에 대한 수요는 D_m ($m = 1, 2, \dots, M$)로 두고, 이의 최대 공약수를 h 라 하자. 그리고 벡터 $\mathbf{d} = (d_1, \dots, d_M)$, $d_m = D_m/h$ 로 두자. 즉, 사이클당 각 모델의 수요 (d_1, \dots, d_M) 를 반복 생산하는 사이클 생산을 사용한다. 여기서 각 모델의 수요, \mathbf{d} 를 최소제품 집합(minimum part set : MPS)이라 하며, 이를 반복 생산한다[5]. 즉, MPS를 h 번 반복 생산하여 총 수요를 만족시킨다. 셋째, M 개 모델의 작업 선행관계는 I 개의 작업으로 구성된 결합선행공정도(combined precedence diagram)에 의해 표현된다고 본다. 결합선행공정도는 각 모델에서 수행되는 모든 작업이 중복 표현되지 않도록 하나의 선행공정도로나타낸다. 이 공정도를 이용하여 작업을 편성함으로써 모델에서 공통으로 수행되는 작업들이 서로 다른 작업장에 할당되는 것을 방지할 수 있다. 그리고 작업자의 이동시간은 무시한다.

2.2 다목적

이 연구에서 다루는 다목적은 앞에서 언급하였듯이 작업장간 작업량의 평활화, 가외작업의 최소화, 부품사용의 일정률 유지의 세 목적이다. 각 작업장의 작업량이 평활화되지 않으면 불공평한 작업부하로 작업자의 능률 저하와 불만을 초래할 수 있다. 가외작업의 최소화는 노동비용을 감소시키고, 컨베이어 정지 위험과 라인길이를 줄이는 데 기여한다. 부품사용의 일정률 유지는 JIT(just-in-time)시스템에서 주요 목적이다. JIT(just-in-time)시스템에서 하나의 기본 요구사항은 안정적이고 연속적인 부품 공급이다. 이는 부품 사용률을 일정하게 함으로써 실현할 수 있다. 그러므로 이 목적은 생산을 수요에 맞추어 함으로써 달성할 수 있다. 이 목적을 사용함으로써 수요자의 요구를 만족하면서 완제품과 공정 재고를 줄일 수 있다.

작업장간 작업량의 평활화는 식 (1)과 같이 표현할 수 있다.

$$\text{Min. } f = \frac{1}{J} \sum_{j=1}^J |T_j - \bar{T}| \quad (1)$$

식 (1)에서 T_j 는 MPS 생산 동안 작업장 j 의 작업량으로, $T_j = \sum_{i \in I(j)} \sum_{m=1}^M d_m t_{im}$ 이다. 여기서 $I(j)$ 는 작업장 j 에 할당된 작업의 집합을 나타내고, t_{im} 은 모델 m 에서 작업 i 의 작업시간이다. $\bar{T} (= \frac{1}{J} \sum_{j=1}^J \sum_{m=1}^M d_m t_{im})$ 는 MPS 생산 동안의 작업장의 평균 작업량을 나타낸다.

앞에서 언급하였듯이, 가외작업은 각 제품에 대하여 고정된 길이의 작업장 내에서 완료되지 못한 작업의 양이다. MPS 생산기간 동안 작업할당과 투입 순서가 주어지면 가외작업은 식 (2)에 의해 구할 수 있다[16].

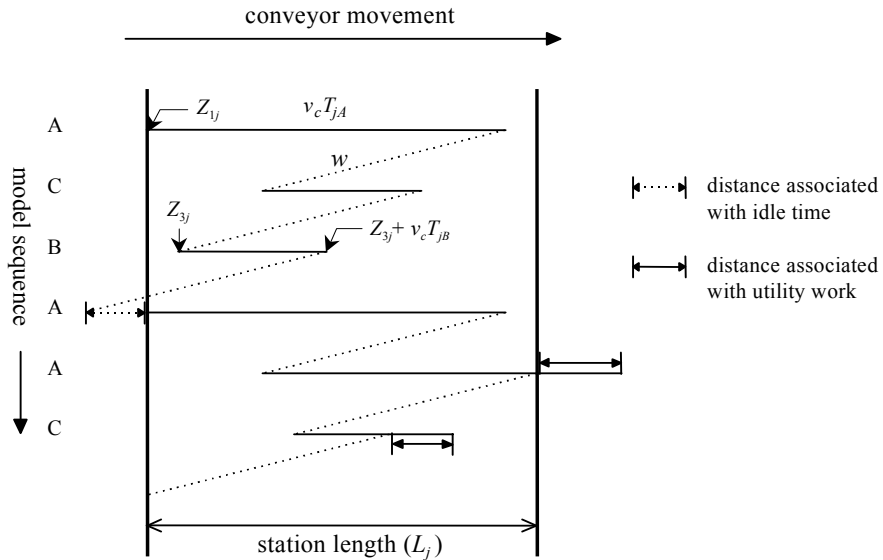
$$\text{Min. } U = \sum_{j=1}^J \left\{ \sum_{s=1}^S \max [0, (Z_{sj} + \nu_c T_{jm_s} - L_j) / \nu_c] + Z_{(s+1)j} / \nu_c \right\} \quad (2)$$

여기서, 작업장의 총수는 J 개이고, $S (= \sum_{m=1}^M d_m)$ 는 MPS 생산 기간의 총 생산량이며, m_s 는 생산 순서가 s 번째인 모델이다. Z_{sj} 는 작업장 j 에서 s 번째 생산 모델의 작업 시작위치이다. T_{jm_s} 는 작업장 j 에서 s 번째 투입 모델 m_s 의 작업량을 나타내고, L_j 는 작업장 j 의 라인길이이다. $\nu_c T_{jm_s}$ 은 모델 m_s 를 작업하는 작업장의 길이를 의미하기 때문에, $Z_{sj} + \nu_c T_{jm_s}$ 는 이 작업이 끝나는 위치를 뜻하고 이 값이 작업장 j 의 길이 L_j 보다 크면, 즉, $Z_{sj} + \nu_c T_{jm_s} - L_j$ 이 0보다 크면 가외작업이 발생한다. 따라서 $\max [0, (Z_{sj} + \nu_c T_{jm_s} - L_j) / \nu_c]$ 은 작업장 j 에서 s 번째 투입모델의 가외 작업시간이고, $\sum_{s=1}^S \max [0, (Z_{sj} + \nu_c T_{jm_s} - L_j) / \nu_c]$ 은 작업장 j 에서 MPS 생산기간 동안에 발생하는 가외작업 시간이다. 모든 작업장에서 생산 사이클(MPS)의 첫 번째 투입모델의 작업 시작위치는 기준점으로 둔다. 즉, $Z_{1j} = 0, \forall j$ 이다. 그러나 일반적으로 이 조건은 만족되지 않는다. 따라서 마지막 모델 다음의 시작점이 기준점에 오지 못하면 그 만큼 다음 MPS 생산에서 가외작업이 발생한다. 두 번째 항은 이러한 가정을 고려하였다.

작업장 j 의 작업 시작 위치는 아래와 같이 계산된다.

$$Z_{(s+1)j} = \max [0, \min (Z_{sj} + \nu_c T_{jm_s} - w, L_j - w)], \quad s = 1, 2, \dots, S, \quad Z_{1j} = 0 \quad (3)$$

여기서, $w (= \nu_c \gamma)$ 는 투입간격 동안의 컨베이어의 이동거리이다. 앞에서 언급했듯이 $Z_{sj} + \nu_c T_{jm_s}$ 는 작업장 j 에서 s 번째 투입모델의 작업이 끝나는 지점이므로 이 값이 작업장 j 의 길이 L_j 보다 크면 가외작업이 이루어 지므로 지점 L_j 에서 끝내게 된다. 따라서 다음 모델의 시작 지점은 $\min (Z_{sj} + \nu_c T_{jm_s} - w, L_j - w)$ 이 된다. 그런데 $Z_{sj} + \nu_c T_{jm_s} - w$ 이 0보다 작으면 앞 작업장의 영역이 되므로 기준점, 즉 지점 0에서 시작하게 된다. 이를 수리적으로 표현하면 식



[그림 1] 폐쇄작업장에서의 가외작업

(3)과 같이 된다.

예로, [그림 1]을 보자. 이 그림은 작업할당에 의해 작업장에서 행하는 각 모델의 작업량이 주어졌고, 모델의 투입순서가(A C B A A C)일 때 가외작업과 유ힴ시간을 도식한 것이다. 점선은 모델의 투입간격 동안 컨베이어의 이동거리($w = v_c \gamma$)이다. 모델의 투입간격은 일정하기 때문에 점선은 평행하다. 다음에 작업해야 할 제품이 작업장내에 있으면 바로 작업을 시작하지만, 네 번째 제품인 A모델 같은 경우에는 제품이 작업장의 기준점에 도달할 때까지 유ힴ시간(idle time)을 가져야 한다. 가외작업은 다섯 번째 투입되는 A모델과 여섯 번째 투입되는 C모델 작업 중에 발생한다. 다섯 번째 투입되는 A모델은 작업이 끝나기 전에 작업장의 길이를 넘어섰기 때문이고, 여섯 번째 투입되는 C모델은 다음 MPS에서 첫 번째 투입되는 모델이 기준점에서 시작해야 한다는 가정에서 비롯된 것이다.

이 연구에서 부품사용의 일정률 유지의 목적함수는 Miltenburg[20]가 제시한 식 (4)을 사용한다.

$$Min. \sum_{s=1}^S \sum_{m=1}^M \left| \sum_{l=1}^s \frac{X_{lm}}{s} - \frac{d_m}{S} \right| \quad (4)$$

여기서, X_{lm} 은 l 번째 생산되는 제품이 모델 m 이면 1, 그렇지 않으면 0이다. 식 (4)에서 첫 항은 s 번째 제품 생산까지 모델 m 의 생산비율이고, 둘째 항은 모델 m 의 수요비율이다.

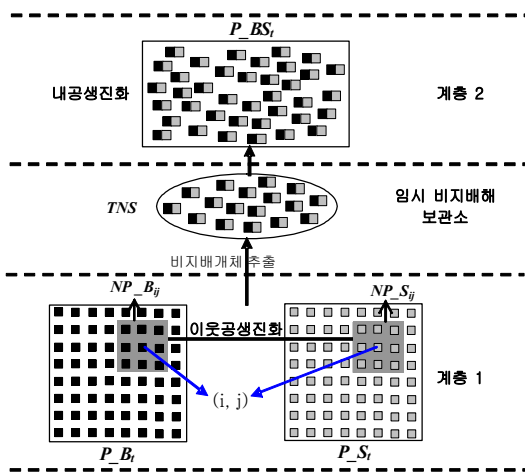
이들 세 목적의 관계에 대해 살펴보자. 먼저 작업할당과 모델 투입순서가 이들 세 목적과 어떻게 관계되는지를 보자. 각 작업장의 작업량은 작업 할당(라인밸런싱)에 의해 결정되고, 가외작업량은 작업 할당과 모델 투입순서에 영향을 받으며, 부품 사용률은 모델 투입순서에 의해서만 결정된다. 작업자들에게 작업 부하를 균등하게 하려는 작업장간 작업량의 평활화는 조립라인 밸런싱에서 하나의 주요 목적이다. 그러나 혼합모델 조립라인에서 이 목적은 작업장에 할당된 모델별 작업량의 편차가 크면 가외작업을 증가시킬 수 있다. 또한 부품사용률 유지를 최대화하는 최적의 투입 순서는 최소 가외작업을 보장하지 않는다. 이는 가외작업을 최소로 하는 투입순서는 각 작업장에 할당된 모델별 작업량에 종속되기 때문이다. 이로부터 세 목적을 동시에 극대화하는 작업할당과 투입 순서가 존재하지 않으므로, 이 연구에서는 다양하고 좋은 여러 비지배해를 찾는 것을 목표로 한다. 이들 비지배해에서 의사 결정자의 선

호에 의해 하나의 해를 선택할 수 있다.

3. 다목적 이웃 공생 진화알고리즘

3.1 개념과 특징

이 연구에서 새로이 제안하는 다목적 이웃 공생 진화알고리즘(MNSEA)은 해의 다양성과 수렴성을 높이기 위해 [그림 2]와 같이 2계층 구조로 설계되었다. MNSEA는 문제의 해를 하나의 개체로 표현하는 기존 MOEA와는 달리 전체문제를 부분문제로 분해한 후, 부분문제들로 이루어진 여러 부분모집단을 운영하여 다양한 비지배해를 효율적으로 탐색할 수 있도록 한다. 계층 1에서는 공생진화를, 계층 2에서는 내공생 진화를 모방하여 알고리즘을 구현하였다.



[그림 2] MNSEA의 구조

이 알고리즘에서 공생진화는 공생관계를 갖는 서로 다른 종들에 대해 각각 모집단을 구성하고, 각 모집단은 다른 모집단과 상호작용하고 상호적응하는 과정을 모방하고 있다. 각 모집단의 개체는 해당 부분문제의 해로 부분 해가 된다. 내공생 진화는 원핵생물들이 내공생을 통해 진핵생물로 진화하는 과정을 모방한 진화전략이다. 내공생은 큰 숙주 원핵생물에 작은 원핵생물이 들어가 결합한 형태로 공생하

는 관계이다. 내공생 진화에서 모집단의 개체는 고전적 진화알고리즘과 같이 부분 해들이 결합된 완전 해를 나타낸다.

제안하는 알고리즘의 개념과 특징에 대해 좀더 설명하면 다음과 같다.

첫째, [그림 2]와 같이 계층 1에서는 밸런싱문제와 투입순서문제를 하나의 종으로 보고, 밸런싱 부분모집단 P_{B_i} 와 순서 부분모집단 P_{S_i} 로 구성된 공생진화 알고리즘을 수행한다. 이들 부분모집단 개체는 다른 부분모집단 개체와 짝을 이루어 적응도를 부여 받고, 이 적응도에 기초하여 진화한다. 여기서 짝을 이루는 상대 개체를 공생자라 부른다. 부분문제 간의 공생진화는 병렬탐색을 강화하여 넓은 해 공간을 효율적으로 탐색하여 해의 다양성을 높이는 데 기여할 것으로 본다.

둘째, 계층 2에서는 완전해를 표현한 개체들로 구성된 하나의 모집단 P_{BS_j} 로 운영되며, 진화는 내공생진화의 과정을 모방하고 있다. 계층 2는 계층 1과 상호 작용한다. 계층 1의 진화 과정에서 얻은 비지배해를 [그림 2]처럼 임시 비지배해 보관소인 TNS에 저장한다. TNS에는 계층 1의 한 세대가 진화하는 동안 발견된 비지배해 개체결합이 저장되어 있다. 이 저장된 비지배 개체결합은 계층 2의 진화시 내공생자(endosymbiont)의 역할을 한다. 즉, 계층 2가 진화 할 때 계층 1에서 적응도가 높은 개체들이 결합된 상태로 계층 2로 이동하여 계층 2의 진화에 이용된다. 이 과정을 통하여 계층 1에서 공생자의 짝은 변동으로 인해 저해 될 수 있는 탐색방향의 유지를 통해 진화 속도의 향상을 유도하여 해의 수렴성을 높이는데 기여할 수 있다.

셋째, 계층 1에서는 공생 진화알고리즘이 가지고 있는 다양성을 좀 더 강화하기 위하여 이웃진화전략을 사용한다. 진화알고리즘에서 모집단의 다양성은 부분 최적해로의 조기 수렴 방지와 함께 해 공간의 효율적 탐색을 위해 중요하게 다루어진다. 이웃진화 전략은 지역적인 상호작용으로 다양한 좋은 개체들이 적소(niche)를 형성하도록 하여 해 공간의 효율적 탐색을 가능하게 한다. MNSEA에서 이웃

공생진화는 [그림 2]와 같이 각 부분모집단의 이웃 모집단을 NP_B_{ij} , NP_S_{ij} 로 각각 정의하고, 두 이웃은 상대 이웃을 공생자로 하여 각기 진화한다.

넷째, 해의 수렴성을 높이기 위하여 계층 2의 통합 탐색과 함께, 계층 1과 계층 2에서 엘리티즘(elitism) 전략을 사용한다. 부모세대의 우수한 개체가 사라지는 것을 방지하기 위해 계층 1의 이웃 모집단과 계층 2의 각 모집단은 $(\mu+\lambda)$ -생존자 선택(survivors selection)을 사용한다. 이 선택 전략은 각 계층의 모집단이 μ 개의 부모와 재생산된 λ 개의 자손의 합인 $(\mu+\lambda)$ 개의 개체에서 적응도가 가장 높은 μ 개를 선택하여 다음 세대 모집단을 구성하는 것이다.

3.2 MNSEA 절차

MNSEA의 절차는 다음과 같다.

단계 1. (초기화)

단계 1.1 : 세대 $t = 0$, $TNS = \phi$ 로 둔다.

단계 1.2 : 각 부분문제를 위한 두 부분모집단 P_B_t , P_S_t 와 상위 모집단 P_BS_t 를 임의로 생성한다.

단계 1.3 : 계층 1의 각 부분 모집단 P_B_t 와 P_S_t 는 상대 모집단에서 임의로 선택한 개체와 짝지어 목적함수 벡터를 계산한다.

단계 1.4 : 구해진 목적함수 벡터를 이용하여 각 부분 모집단에 있는 개체의 적응도를 평가한다.

단계 1.5 : 모집단 P_BS_t 에 있는 개체의 적응도를 평가한다. 그리고 세대 $t = 1$ 로 둔다.

단계 2. (계층 1의 이웃 진화)

단계 2.1 : 임의의 위치(i, j)를 선택하여 각 부분 모집단의 이웃 NP_B_{ij} 와 NP_S_{ij} 를 정의한다. 진화되는 이웃을 $ENP = NP_B_{ij}$ 공생 이웃을 $SNP = NP_S_{ij}$ 로 두고 $F = 0$ 으로 둔다.

단계 2.2 : ENP 에 유전연산을 수행하여 ENP 와

같은 크기의 자손모집단 ENQ 를 만든다.

단계 2.3 : 상대 이웃 SNP 에서 임의로 선택한 개체를 공생자로 하여 ENQ 에 있는 개체의 목적함수 벡터를 계산한다.

단계 2.4 : $TN = ENP \cup ENQ$ 에 있는 개체의 적응도를 평가한다.

단계 2.5 : TN 에서 적응도가 높은 $|ENP|$ 개의 개체를 확정적으로 선택한다. 이때 ENQ 에서 선택된 개체는 ENP 에서 도태되는 개체의 위치에 두어 새로운 ENP 를 구성한다.

단계 2.6 : 진화된 ENP 에서 비지배개체 결합을 TNS 에 저장한다. 서로 다른 비지배개체로 TNS 를 갱신한다.

단계 2.7 : $F = 0$ 이면, $F = 1$, $ENP = NP_S_{ij}$, $SNP = NP_B_{ij}$ 로 두고 단계 2.2로 간다. 그렇지 않으면 단계 3으로 간다.

단계 3. (이웃진화 종료조건)

이웃 진화 종료조건을 만족하면 $P_B_{t+1} \leftarrow P_B_t$, $P_S_{t+1} \leftarrow P_S_t$ 로 둔다. 그렇지 않으면 단계 2로 간다.

단계 4. (계층 2의 진화)

단계 4.1 : P_BS_t 로부터 이진토너먼트 선택과 유전연산을 통해 자손을 생산하여 자손 모집단 Q_BS_t 를 만든다.

단계 4.2 : 합집합 $P_BS_t \cup Q_BS_t \cup TNS$ 의 임시 모집단을 만들고, 임시모집단의 개체 적응도를 계산한다.

단계 4.3 : 임시모집단에서 적응도가 높은 순으로 개체를 선택하여 다음 세대 모집단 P_BS_{t+1} 을 구성한다. 즉, $(\mu+\lambda)$ -서로 다른 생존자 선택에서 사용되는 선택 전략을 사용한다.

단계 5. (종료조건)

종료조건을 만족하면 끝낸다. 이때 P_BS_t 에 있는

비지배해가 구하는 해가 된다. 그렇지 않으면 $t \leftarrow t + 1$, $TNS = \phi$ 로 두고 단계 2로 간다.

[그림 3]은 MNSEA의 절차를 도식한 것이다. 단계 1.4, 단계 2.4, 단계 4.2에서 적응도를 평가하는 방법은 제 3.3절에서 다룬다. 단계 2.2와 단계 4.1에서 이루어지는 유전연산 중 벨런싱 문제를 위한 유전연산은 제 4.2.1절에서, 순서 문제를 위한 유전연산은 제 4.2.2절에서 다룬다. 그리고 파라미터로, 모집단과 이웃의 크기, 교차율과 돌연변이율, 이웃 진화와 알고리즘 종료조건은 실험 결과와 함께 제 5장에서 언급한다. 특히, 단계 3에서 이웃진화의 종료조건은 일종의 파라미터로 이 논문에서는 이웃진화에 의해 생산된 개체 수와 모집단 크기, 그리고 해의 질을 고려하여 결정하였다.

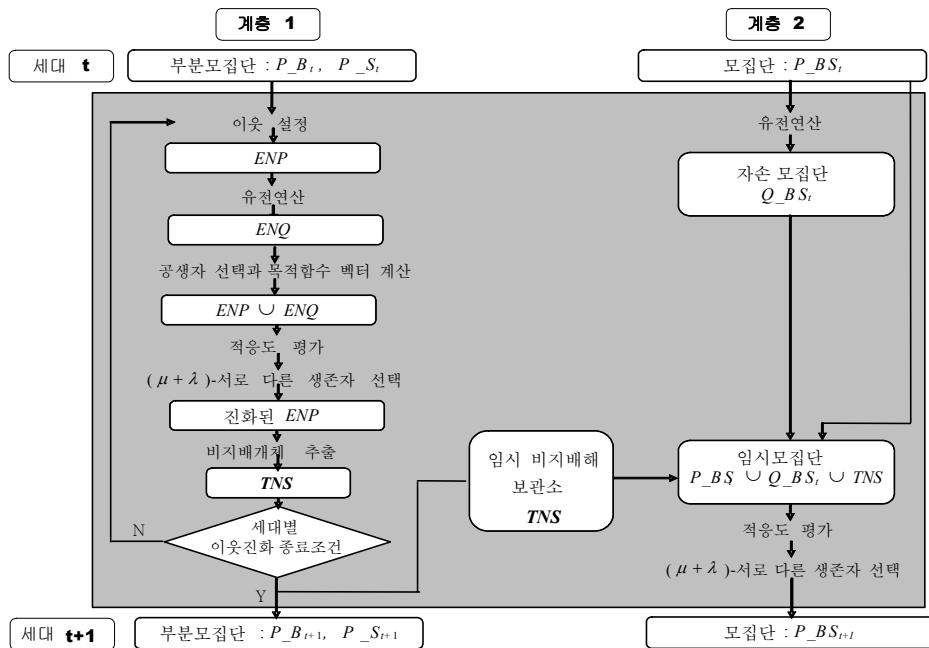
3.3 적응도 평가

계층 1에 있는 개체의 적응도 평가를 위해서는 먼

저 다른 모집단의 이웃에서 임의로 공생자를 선택 (절차에서 단계 2.3 참조)하여 목적함수 벡터를 계산한다. 계층 2는 완전해이므로 공생자가 필요하지 않다. 이렇게 계산된 목적함수 벡터를 기준으로 적응도를 부여한다. 이 연구에서는 NSGA-III[10]에서 사용한 적응도 평가 방법을 채용한다. 개체 i 의 적응도는 다음의 평가함수 $eval_i$ 에 의해 계산되고, 이 값이 낮을수록 높은 적응도를 갖는다.

$$eval_i = r(i) + \frac{1}{1+d(i)} \tag{5}$$

여기서 $r(i)$ 는 개체 i 의 비지배 순위(rank), $d(i)$ 는 개체 i 와 동일한 비지배 순위에 있는 다른 개체들과의 밀집거리를 나타낸다. 비지배 순위, $r(i)$ 는 개체 집합에서 비지배해를 구하여 순위 1로 두고, 구한 비지배해를 이 집합에서 제거한다. 그리고 다시 이 집합에서 비지배해를 구하여 순위 2로 둔다. 이와 같은 방법을 이 집합이 공집합이 될 때까지 반



[그림 3] MNSEA의 흐름도

복한다. 개체 i 의 밀집거리, $d(i)$ 는 다음과 같이 계산한다. 각 목적 별로 목적함수 값의 오름차순으로 정렬한다. 각 목적에 대해 정렬한 리스트에서 개체 i 의 바로 앞과 뒤에 있는 목적 값의 차의 절대값을 계산한다. 모든 목적에 대해 구한 이들 절대 값의 합을 $d(i)$ 로 둔다. 이 때 각 목적에서 가장 앞(최소값)과 가장 뒤(최대값)에 있는 값을 갖는 개체의 밀집거리는 무한대로 둔다. 이 방법은 개체의 비지배 순위를 우선적으로 고려하여 적응도를 부여하고, 같은 비지배 순위에 있는 개체는 밀집거리를 반영하여 밀집거리가 클수록 보다 높은 적응도를 갖도록 한다.

이 연구에서 채용한 Deb et al.[10]의 NSGA-II에서 적응도 평가방법은 일차로 비지배 순위를 고려하고, 비재순위가 같으면 밀집거리에 의해 적응도를 다르게 평가하고 있다. 이렇게 함으로써 비지배 순위에 의해 해의 수렴성을, 밀집거리에 의해 해의 다양성을 고려하고 있다. 또한 비지배 순위를 구하는데 계산의 복잡도를 낮게 하는 방법을 제안하였다. 이는 Deb et al.[10]을 참고할 수 있다. 한편, SPEA2 [31]에서는 좋은 해를 보관하는 보관소를 두어 모집단과 보관소에 있는 개체들을 대상으로 하여 먼저 지배하는 개체 수에 근거하여 적응도를 평가하고, 같은 적응도에 대해서는 이웃하는 개체와의 거리에 의해 이를 차별화하여 평가한다. NSGA-II에서는 우선적으로 비지배 순위에, SPEA2에서는 먼저 지배 해의 수에 근거하여 평가함으로써, NSGA-II는 해의 수렴성을, SPEA2는 해의 다양성에 좀더 유리하게 작용한다고 볼 수 있다.

4. MMAL의 밸런싱과 투입순서 결정을 위한 유전표현과 유전연산

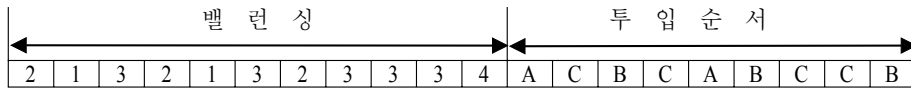
4.1 유전표현과 초기 모집단

진화알고리즘에서 유전표현은 주로 해결하고자 하는 문제의 특성에 따르고 가능한 자연스럽고, 명료하며, 비중복적이어야 한다. 이 연구에서 밸런싱문

제는 그룹번호표현(group-number representation)을 사용한다. 계층 1에서 밸런싱문제의 부분모집단 길이는 I (작업의 수)이며, 각 인자는 1부터 J (작업장 수) 범위에 있는 정수 값을 갖는다. 개체 표현에서 i 번째 인자 값이 j 이면 작업 i 는 작업장 j 에 할당됨을 나타낸다. 예로, (2 1 3 2 1 3 2 3 3 3 4)로 표현된 개체에서 인자 값이 1인 개체가 두 번째와 다섯 번째 인자이므로 작업장 1에 작업 2와 5가 할당된다. 이와 같은 방법으로 작업장 2, 3, 4에는 각각 작업 {1, 4, 7}, {3, 6, 8, 9, 10}, {11}이 할당된다. 이러한 표현은 해석이 용이하고 각 인자가 갖는 정보를 명확히 나타내는 장점이 있다. 초기 모집단을 구성할 때 가능(선행관계를 만족하는 할당) 개체의 생성은 다음과 같이 한다. 선행 작업이 없는 작업을 할당후 모집합으로 구성하고 이들 중 한 작업을 임의로 선택하여 첫 작업장부터 할당한다. 그리고 할당된 작업은 삭제하고 새로이 선행작업이 없는 작업을 후보집합에 추가하고 다시 이들 작업 중에서 임의로 하나를 선택하여 할당한다. 이 과정을 반복하면서 새로운 작업이 할당될 때, 현 작업장의 작업량이 작업장의 평균 작업량을 초과하게 되면 이 작업을 현 작업장 또는 다음 작업장에 0.5 확률로 할당하고 다음 작업장을 연다. 이와 같은 방법으로 ($J-1$)작업장까지 할당하고, 미할당 작업은 모두 마지막 작업장 J 에 할당한다.

계층 1의 투입순서 문제에서 부분모집단 개체들은 간단히 MPS 생산 동안 라인에 투입하는 순서대로 열거한 순서 리스트로 표현한다. 예로, MPS의 $d = (d_A, d_B, d_C) = (2, 3, 4)$ 이고 투입순서가 A, C, B, C, B, A, C, C, B이면 개체는 (A C B C B A C C B)로 표현된다. 이 표현은 투입 순서대로 표현되어 개체해석과 유전인자의 정보이용이 용이하다는 장점이 있다. 초기 모집단은 MPS 제약을 만족하는 개체들을 모집단 크기만큼 임의로 생성하여 구성한다.

계층 2의 모집단 개체는 밸런싱과 투입순서 문제의 염색체를 결합한 형태로 표현한다. 이를 도식하면 [그림 4]와 같다.



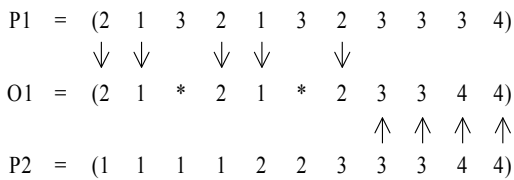
[그림 4] 계층 2의 모집단 유전 표현

4.2 유전 연산자

4.2.1 MMAL의 벨런싱을 위한 유전 연산자

교차연산자는 문제의 특성을 잘 반영하여 부모가 갖고 있는 좋은 정보를 자손에게 전달 할 수 있어야 한다. 또한, 재생산된 자손 개체들은 벨런싱문제의 가능해 유지 조건을 만족해야 한다. 이를 위하여 이 연구에서는 그룹의 정보를 잘 보존하여 자손에 전달 할 수 있는 구조 교차(structural crossover)를 벨런싱문제에 맞게 변형하여 사용한다. 이 방법의 절차는 다음과 같다.

- 단계 1 : 범위 [1, J]에서 임의의 정수 r을 선택한다.
- 단계 2 : 부모 P1에서 1부터 r의 값을 갖는 인자를 자손 O1의 동일한 위치에 상속한다.
- 단계 3 : 자손 O1의 남은 위치에 부모 P2의 같은 위치에서 r+1부터 J의 값을 갖는 인자를 상속한다.
- 단계 4 : 자손 O1의 비어있는 위치는 아래에 설명된 재할당 방법을 이용하여 할당한다.



[그림 5] 구조교차

다른 자손 O2는 P1과 P2의 역할을 바꾸어 생산한다. [그림 5]는 위 절차를 도식한 것이다. 4개의 작업장 중 임의로 선택되는 작업장 r이 작업장 2인 경우, 부모 P1의 인자 중 작업장 1과 작업장 2인 인자들을 자손개체 O1의 같은 위치에 복사하고, 남은 위치에 부모 P2의 같은 위치에 있는 작업장 3과 작

업장 4를 상속한다. [그림 5]에서 작업장이 할당되지 않은 작업 3과 작업 6은 *로 표시하고 재할당 방법을 이용하여 할당한다.

돌연변이는 해공간의 다양성을 유지하고 부분최해에 조기 수렴하는 것을 방지 하는 역할을 한다. 벨런싱문제에서는 개체 돌연변이율에 의해 선택된 개체에 대하여 인자 돌연변이율에 따라 인자를 돌연변이 시킨다. 이들 인자 역시 아래에 설명할 재할당 방법을 이용하여 할당한다.

위에서 설명한 교차와 돌연변이 연산에서 미할당 인자들의 재할당이 요구된다. 이는 아래와 같은 재할당 기법에 의해 이루어진다. 이 기법에서는 선행 제약을 만족하도록 할당한다. 먼저 재할당 기법의 절차에 사용되는 기호를 정의한다.

- R : 할당되지 않는 작업의 집합
- $IP(i)$: 작업 i 의 직선행 작업집합
- $S(i)$: 작업 i 의 후행 작업집합
- $NR(i)$: 집합 R 의 원소 중에서 작업 i 의 직선행 작업집합; $NR(i) = R \cap IP(i)$
- E_i : 개체에서 $IP(i)$ 의 작업들에 해당하는 인자 값 중 가장 작은 값, $IP(i) = \phi$ 이면, $E_i = 1$
- L_i : 개체에서 $S(i)-R$ 의 작업들에 해당하는 인자 값 중 가장 작은 값, $S(i)-R = \phi$ 이면, $L_i = J$
- $AW(i) : \{E_i, E_{i+1}, \dots, L_i\}$ (작업 i 가 할당 가능한 작업장의 집합)
- T'_j : 작업장 j 에 이미 할당된 작업들의 MPS 생산시간의 합

단계 1 : 집합 R 과 $NR(i)$ 를 구하고, $T'_j, j = 1, 2, \dots, J$ 를 계산한다.

단계 2 : 집합 R 의 원소이고, $NR(i) = \phi$ 인 작업

중 MPS 생산 동안의 작업시간이 가장 큰 작업 i^* 를 선택한다.

단계 3: i^* 의 E_i 와 L_i 를 구하고, i^* 의 할당가능 인자 값 집합 $AW(i^*)$ 를 구한다.

단계 4: $T_j' + \sum_{m=1}^M d_m t_{i^*m} \leq \bar{T}$ 이고 $j \in AW(i^*)$ 를 만

족하는 작업장이 존재하면, 이들 중 가장 빠른 작업장에 j^* 를 할당하고, 그렇지 않으면, 가장 작은 T_j' 를 가지는 작업장에 j^* 를 할당한다. 가장 작은 T_j' 가 두 개 이상인 경우에는 이들 중 가장 빠른 작업장에 j^* 를 할당한다.

단계 5: 작업 i^* 를 작업장 j^* 에 할당하고 i^* 를 집합 R 에서 삭제 한다. $R = \phi$ 이면 종료하고, 그렇지 않으면, $NR(i)$, T_j' , $j = 1, 2, \dots, J$ 를 갱신하고 단계 2로 간다.

위 절차에서 단계 1부터 단계 3까지는 선행제약을 만족하는 할당을 위한 과정이다. 특히, 단계 2에서 작업시간이 가장 큰 작업의 선택은 BFD(best-fit-decreasing)기법에 기초한 것이다. 단계 4는 작업장의 작업량 평활화를 고려하면서, 다음 작업의 할당 영역을 높이기 위해 가장 빠른 작업장에 할당하고 있다.

4.2.2 MMAL의 투입순서 결정을 위한 유전연산자

계층 1의 투입순서 부분모집단에서는 수정순서교차(Modified order crossover operator)와 역순(inversion) 연산자를 사용한다. 이들 두 연산자는 MMAL의 투입순서 문제에서 좋은 탐색성능을 보였다[15]. 수정순서교차는 순서교차[1]를 변형한 것으로 부모들의 상대적 순서를 보존하는 성질이 있다. 이 기법의 절차는 다음과 같다.

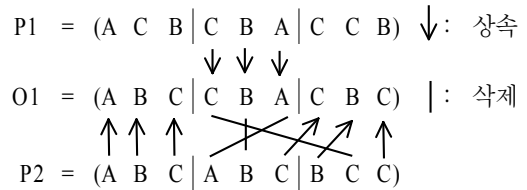
단계 1: 부모개체에 임의의 두 절단점을 지정한다.

단계 2: 부모 P1의 두 절단점 사이에 있는 인자를 자손 O1의 같은 위치에 상속한다

단계 3: 부모 P2에서 O1에 상속된 인자를 임의로 삭제한 후, 부모 P2에 남은 인자들 앞에서부터 자손 O1의 미할당 인자 위치에 차례로 상속시킨다.

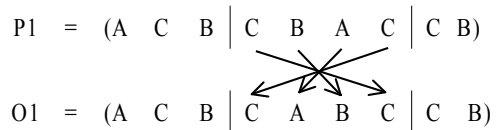
다른 자손 O2는 P1과 P2의 역할을 바꾸어 생산한다.

[그림 6]은 위 절차의 예이다.



[그림 6] 수정순서교차

역순 돌연변이는 부모에서 두 개의 절단점을 임의로 선택하고 절단점 사이의 인자들을 역순으로 하여 자손을 생산하는 방법이다. [그림 7]은 역순연산자를 보여주고 있다. 여기서 두 절단점은 임의로 선택된 것이다.



[그림 7] 역순 연산자.

또한, 계층 2에서 유전연산자는 밸런싱과 투입순서 문제에 해당하는 염색체에 각기 앞에서 설명한 교차와 돌연변이 연산자를 사용한다.

5. 실험과 분석

5.1 실험문제와 비교알고리즘

제안한 MNSEA의 성능은 컴퓨터 모의실험을 통해서 평가한다. 실험문제는 19개의 작업을 갖는 Thomopoulos 문제[27], 61개의 작업을 갖는 Kim et al. 문제[16], 그리고 111개의 작업을 가진 Arcus 문제

[4]를 이용하였다. <표 1>와 같이 실험문제는 작업장 수와 MPS를 다양하게 변화시켜 사용하였다. 실험문제에서 컨베이어 이동속도(v_c)는 계산 편의를 위해 1로 두었으며, 제품 투입시간간격(γ)은 1회의 MPS를 생산하는데 요구되는 총 생산시간($\sum_{i=1}^I \sum_{m=1}^M t_{im} d_m$)

을 제품의 총수($S = \sum_{m=1}^M d_m$)와 작업장 수(J)의 곱으로 나눈 값, 즉 $\gamma = \frac{\sum_{i=1}^I \sum_{m=1}^M t_{im} d_m}{(S \times J)}$ 로 두었다. 각 작업장의 길이는 투입 시간간격의 1.5배로 두었다.

이 연구에서 사용되는 파라미터는 예비실험을 통하여 결정하였다. 첫째, 모집단은 문제의 종류에 따라 다르게 주었는데 이는 Arcus 문제와 같은 대형문제에서는 모집단의 크기가 작은 경우 해 공간을 탐색능력이 떨어져 모집단의 다양성을 유지하기 어렵기 때문이다. 각 문제의 모집단 크기는 Thomopoulos 문제 100(10×10의 격자구조), Kim et al. 문제 196(14×14의 격자구조), Arcus 문제 400(20×20의 격자구조)으로 두었다. 이웃의 크기 각각 (3×3), (4×4), (5×5)의 정방형으로 두었고 단계 3의 이웃진화 종료조건은 각각 한 세대에 15개, 20개, 20개의 다른 이웃을 지정하여 진화하는 것으로 두었다. 둘째 교차율은 벨런싱과 투입순서 문제 모두 1.0으로 하여 이웃 크기와 같은 수의 개체를 재생산하였다. 셋째, 1계층에서 이루어지는 이웃 공생 진화에서 벨런싱 개체의 개체 돌연변이율과 인자 돌연변이율은 모두 0.3으로, 투입순서 개체의 개체 돌연변이율은 0.1로 두었다. 2계층에서는 개체단위의 돌연변이를 사용하였다. 이때 개체 돌연변이율은 0.3으로 하였고, 벨런싱 염색체에서는 돌연변이 개체에 대해 0.3의 인자 돌연변이율을 사용하였다. 마지막으로 Thomopoulos 문제와 Kim et al. 문제는 종료조건을 100세대로 동일하게 적용하였고, Arcus 문제는 150세대를 종료조건으로 적용하였다.

제안하는 MNSEA의 성능 분석을 위해, 대표적 MOEA인 NSGA-II[10]와 제안하는 알고리즘과 유사하나 이웃 진화하지 않고 모집단 단위로 진화하

<표 1> 실험문제

problem	Number of tasks	Number of models	Number of stations	MPS
Thom1	19	3	3	5 3 2
Thom2	19	3	6	2 3 5
Kim1	61	4	6	5 5 6 4
Kim2	61	4	6	8 6 3 2
Kim3	61	4	6	1 2 9 10
Kim4	61	4	9	5 5 6 4
Kim5	61	4	9	8 6 3 2
Kim6	61	4	9	1 2 9 10
Arcus1	111	5	12	5 6 5 6 4
Arcus2	111	5	12	8 7 5 3 2
Arcus3	111	5	12	1 2 3 9 10
Arcus4	111	5	15	5 6 5 6 4
Arcus5	111	5	15	8 7 5 3 2
Arcus6	111	5	15	1 2 3 9 10
Arcus7	111	5	27	5 6 5 6 4
Arcus8	111	5	27	8 7 5 3 2
Arcus9	111	5	27	1 2 3 9 10

는 MSEA[3]를 비교 알고리즘으로 사용한다. NSGA-II는 제안하는 MNSEA와 같은 적응도 평가 방법과 $(\mu+\lambda)$ -생존자 선택 전략을 사용하고 있다. 그러나 MNSEA와는 달리 개체를 완전해로 표현하고 모집단 단위로 진화하는 1계층 구조로 되어있다. MSEA는 MNSEA와 같으나 다른 점은 이웃진화를 사용하지 않고 모집단 단위로 진화한다는 것이다. 비교알고리즘인 NSGA-II와 MSEA에서 모집단 크기, 교차율, 돌연변이율의 파라미터는 MNSEA와 같게 사용한다. MSEA의 종료 세대 수는 MNSEA와 같게 두었으나, NSGA-II에서는 종료 세대 수는 MNSEA의 3배로 두었다. 이는 MNSEA와 MSEA에서는 세 모집단을 운영하는 데 반해 NSGA-II에서는 하나의 모집단만을 운영하기 때문에, 3배의 종료 세대를 길게 하여 재생산된 개체 수를 거의 같게 하기 위함이다. 이는 결국 알고리즘 성능 비교의 형평성을 고려하기 위함이다.

5.2 알고리즘의 성능 척도

앞에서 언급했듯이, 다목적 진화알고리즘의 목표

는 파레토 최적해에 근접한 비지배해 집합을 다양하게 찾는 것이다. 이를 위해서 해의 수렴성과 다양성에 관한 여러 척도가 제시되었다[7, 11]. 이 연구에서는 최적해로의 수렴성 척도는 Zitzler[30]가 제시한 아래 식을 사용한다.

$$M = \frac{1}{|\mathbf{U}|} \sum_{\mathbf{u} \in \mathbf{U}} D(\mathbf{u}),$$

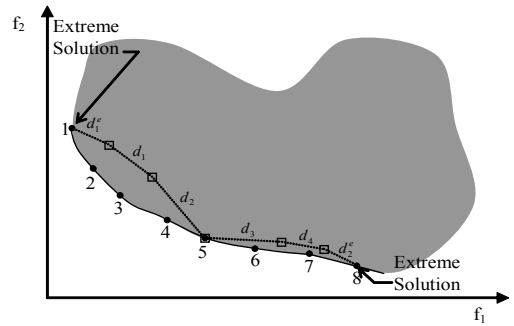
$$D(\mathbf{u}) = \min_{\mathbf{y} \in \mathbf{Y}_p} \sqrt{\sum_{k=1}^K \left(\frac{f_k(\mathbf{u}) - f_k(\mathbf{y})}{f_k^{\max} - f_k^{\min}} \right)^2} \quad (6)$$

여기서, \mathbf{U} , \mathbf{Y}_p 는 각각 구한 비지배해 집합과 파레토 최적해 집합을, K 는 목적의 개수를 나타낸다. f_k^{\max} 과 f_k^{\min} 은 파레토 집합에서 목적 k 의 최대값과 최소값을 각각 나타낸다. 식 (6)으로부터 수렴성의 척도는 구한 각 비지배해에 대해 가장 가까이 있는 파레토 최적해와의 거리(Euclidean 거리)를 구하여 이들의 평균, M 을 사용한다. 이 연구에서는 실제 파레토 최적해를 알지 못하므로, \mathbf{Y}_p 는 각 문제에 대해 충분한 시간 동안 비교알고리즘들을 반복 수행하여 얻은 비지배해를 사용하였다. 또한, 다목적 문제에서 목적간의 스케일(scale)이 현저하게 차이가 나는 경우, 척도 결과가 스케일이 큰 목적에 종속될 수 있어 이를 정규화하였다. M 이 작을수록, 최적해로의 수렴성이 좋다.

해의 다양성 분석을 위하여 Deb[11]가 제시한 식 (7)과 같은 척도를 사용한다.

$$\Delta = \frac{\sum_{k=1}^K d_k^e + \sum_{i=1}^{|\mathbf{u}|-1} |d_i - \bar{d}|}{\sum_{k=1}^K d_k^e + (|\mathbf{u}|-1)\bar{d}} \quad (7)$$

여기서, \mathbf{U} 는 구한 비지배해 집합이고, d_k^e 는 목적 k 에 대한 파레토 최적해의 극점(extreme solution)와 이에 가장 가까운 비지배해와의 거리를 나타낸다. d_i 는 해 공간에서 i 번째와 $(i+1)$ 번째의 이웃하는 비지배해 간의 거리를, \bar{d} 는 이들 이웃해 간의 거리 d_i 의 평균을 나타낸다. 이 다양성 척도 Δ 는 파라미터를 이용하지 않는다는 장점을 가지고 있다.



[그림 8] Deb의 다양성 척도 계산

예로, [그림 8]을 보자. 최소화하는 두 목적이 있고, 해 1부터 8의 파레토 최적해와 사각형으로 표시된 5개의 비지배해가 있다 하자. 해 1과 8은 각 목적에 대해 극점이다. 그리고 여기서 구한 d_1^e 과 d_8^e , 그리고 d_i 는 [그림 8]에서와 같이 두 이웃 해간의 거리이다. $\bar{d} = (d_1 + d_2 + d_3 + d_4)/4$ 가 된다. 구한 비지배해가 극점에 존재하고(즉, 모든 $d_k^e = 0$ 이고), 등간격으로 존재한다면, $\Delta = 0$ 으로 다양성 측면에서 가장 이상적인 해가 된다. 비지배해가 잘 분포될 수록 Δ 값은 작아진다. 이 척도에서 좋지 않은 분포의 경우는 Δ 가 1보다 클 수 있다.

5.3 성능 분석

제안한 MNSEA의 성능 분석을 위해 MSEA와 NSGA-II의 두 진화알고리즘과 비교하였다. 모든 알고리즘은 C++언어로 구현하였으며, 2.4GHz Q6600 CPU를 장착한 IBM-PC에서 수행하였다. 실험은 17개의 실험문제를 20회 반복 수행하였고 그 결과의 평균값을 <표 2>에 나타내었다.

먼저, 최적해로의 수렴성을 보자. M 값으로부터 알 수 있듯이, 모든 실험 문제에서 수렴성에서 MNSEA가 가장 좋고 다음은 MSEA이고, NSGA-II가 가장 좋지 않은 결과를 보였다. 이는 MNSEA가 비교알고리즘에 비해 파레토 최적해에 보다 가까운 해를 찾고 있음을 의미한다. 참고로, NSGA-II는 다목적 진화알고리즘에서 수렴성이 우수한 것으로 알려져 있다[10, 31]. NSGA-II의 구조는 MSEA에서 계층

〈표 2〉 MNSEA와 비교 알고리즘의 결과 비교

problem	M			Δ			Computation time(Sec.)		
	MNSEA	MSEA	NSGA-II	MNSEA	MSEA	NSGA-II	MNSEA	MSEA	NSGA-II
Thom1	0.012	0.016	0.020	0.358	0.364	0.421	5.5	12.4	11.1
Thom2	0.028	0.031	0.040	0.478	0.491	0.572	5.1	11.6	10.9
Kim1	0.011	0.012	0.018	0.474	0.494	0.548	28.0	45.6	42.3
Kim2	0.012	0.013	0.015	0.473	0.493	0.484	28.1	45.3	42.1
Kim3	0.012	0.014	0.018	0.461	0.483	0.498	26.2	44.2	41.5
Kim4	0.017	0.018	0.024	0.471	0.526	0.594	29.2	44.1	41.6
Kim5	0.019	0.020	0.035	0.488	0.488	0.498	27.4	42.9	39.8
Kim6	0.011	0.012	0.022	0.477	0.490	0.501	26.6	43.0	40.4
Arcus1	0.038	0.039	0.046	0.701	0.834	0.850	123.1	334.8	261.1
Arcus2	0.028	0.031	0.031	0.761	0.776	0.806	132.9	299.4	258.6
Arcus3	0.036	0.049	0.057	0.648	0.786	0.804	134.4	330.9	257.7
Arcus4	0.021	0.048	0.053	0.717	0.807	0.844	133.4	287.9	258.3
Arcus5	0.038	0.041	0.058	0.643	0.659	0.889	128.0	281.9	259.4
Arcus6	0.043	0.045	0.052	0.722	0.782	0.850	126.4	303.1	267.1
Arcus7	0.034	0.038	0.043	0.891	0.926	0.905	128.5	322.4	264.4
Arcus8	0.037	0.039	0.041	0.831	0.874	0.865	126.6	320.6	262.5
Arcus9	0.031	0.053	0.054	0.889	0.948	0.920	125.8	341.0	273.9

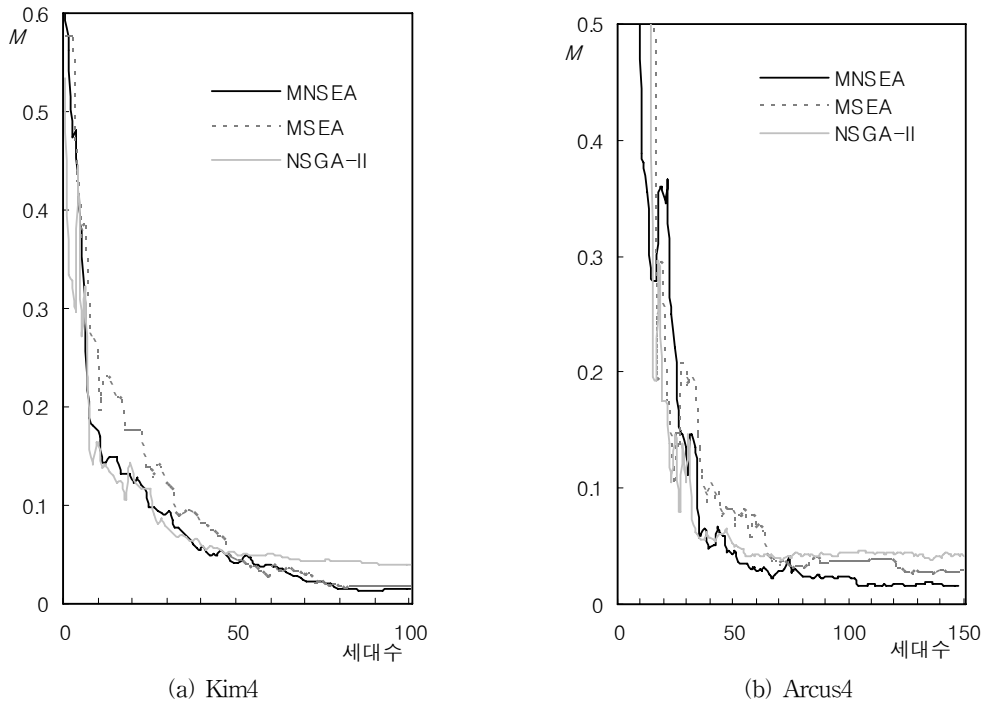
2가 단독으로 진화하는 구조와 같고, MSEA는 계층 1에서 공생진화가 이루어진다. MSEA와 NSGA-II의 수렴성 비교에서, MSEA의 우수성은 공생진화가 이루어지는 계층 1의 비지배 개체들이 계층 2로 전달되는 구조에 기인함을 알 수 있다. 또한 MNSEA가 MSEA보다 수렴성이 좋은 것은 계층 1에서의 이웃진화에서 그 원인을 찾을 수 있다. 이로부터 MNSEA의 이웃진화는 적소를 형성하고 모집단의 다양성을 유지하여[9, 14] 장기적으로 해의 수렴에 기여함을 알 수 있다.

[그림 9]는 실험 문제 Kim4와 Arcus4에 대해 진화에 따른 수렴속도를 나타내고 있다. MNSEA는 초기 단계에는 비교 알고리즘에 비해 수렴속도가 떨어지나, 진화가 진행될수록 수렴성이 좋아진다. 이에 반해, NSGA-II는 초기 단계에는 수렴속도가 빠르나 세대가 진행됨에 따라 수렴성이 떨어진다. 이로부터 MNSEA는 비교알고리즘에 비해 해공간의 지속적 탐색능력이 우수함을 알 수 있다.

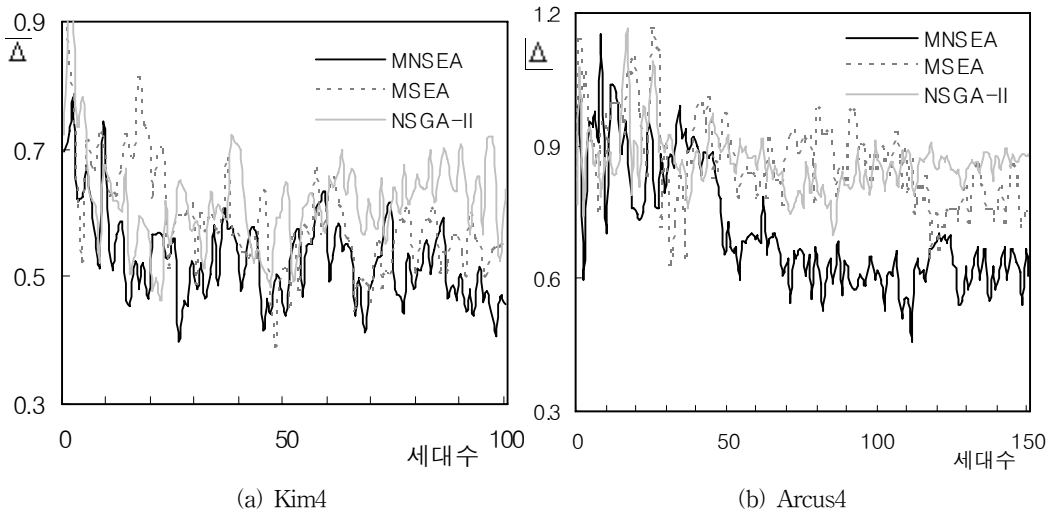
비지배해의 다양성을 보자. <표 2>에서 Δ 값은 모든 실험 문제에서 MNSEA가 가장 작고 다음은 MSEA, NSGA-II순이다. 이는 MNSEA가 비교 알고리즘에 비해 보다 잘 분포된 비지배해를 찾음을 의미한다. MNSEA와 MSEA를 NSGA-II와 비교할 때 가장 큰 차이는 MNSEA와 MSEA가 공생진화를 한다는 것이다. 또한 MNSEA는 MSEA와는 달리 계층 1에서 이웃진화를 한다. 이로부터 공생진화와 이웃진화 전략이 해의 다양성에 기여함을 알 수 있다.

[그림 10]은 Kim4와 Arcus4 문제에서 세대에 따른 비지배해의 다양성을 보여준다. 이 그림에서 MNSEA가 다른 비교알고리즘과 비교하여 세대가 진행됨에 따라 비지배해들이 더 다양하게 분포되는 모습을 확인 할 수 있다. 특히, 문제가 복잡하고 크기가 큰 Arcus4 문제에서 다양성의 우수성을 확연히 보여준다.

다목적 진화알고리즘의 성능 평가에서 최적해로



[그림 9] 문제 Kim4와 Arcus5의 수렴성(M) 비교



[그림 10] 문제 Kim4와 Arcus5의 다양성(Δ) 비교

의 수렴성과 비지배해의 분포와 함께 계산소요시간도 중요하게 다루어진다. <표 2>로부터 계산소요시간은 NSGA-II가 MSEA에 비해 근소하게 빠르

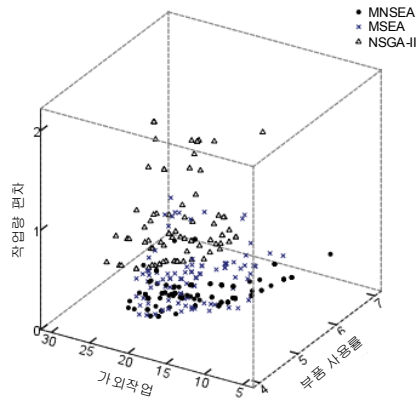
다. 이는 MSEA의 계층 2에서 적응도를 평가할 때 부모모집단과 자손모집단, 그리고 TNS에 있는 개체를 대상으로 비지배해를 분류하는 데 반해, NS

GA-II에서는 부모모집단과 자손모집단을 대상으로 하기 때문이다. 한편, MNSEA의 계산시간은 NSGA-II의 반 정도 소요된다. 이는 MNSEA가 계층 1에서 이웃 진화전략을 사용함으로써 적응도 평가를 위한 비지배 순위의 계산시간이 적게 소요되기 때문이다.

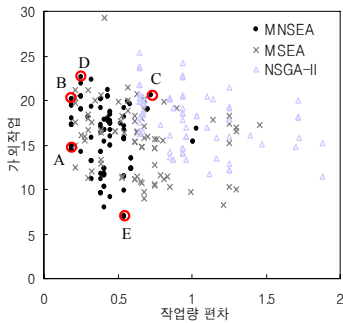
5.4 세 목적간의 결과 분석

이 논문에서 고려하는 세 목적, 즉 작업장간 작업량 평활화, 가외작업 최소화, 부품 사용률 유지의 관계를 구한 비지배해를 통해 분석해 보자. 앞에서 언급했듯이 작업장 평활화는 각 작업의 작업장 할당에, 가외작업은 작업 할당과 투입순서에, 부품 사용률은 모델의 투입순서에 영향을 받는다. 예로, Kim4의 실험문제 결과로 세 목적의 관계를 분석해

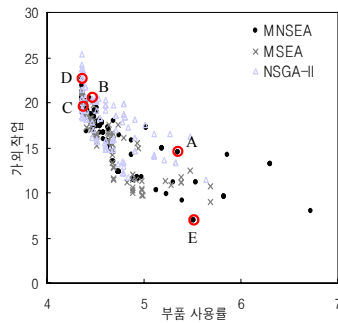
보자. [그림 11]은 이 실험문제에서 비교 알고리즘이 구한 비지배해를 세 목적과 두 목적에 대해 도식한 것이고, <표 3>은 이 그림에서 특정 비지배해의 목적 값을 나타내고 있다. 비지배해들은 목적들이 상충됨을 의미하므로 [그림 11]로부터 세 목적이 여러 형태로 상충됨을 알 수 있다. [그림 11](b)와 (d)로부터 작업량 편차가 일정할 때(하나의 라인 밸런스에서), 모델 투입순서에 의해 가외작업과 부품 사용률이 상충되는 다양한 형태의 비지배해가 존재할 수 있다. 이는 <표 3>의 비지배해 A, B가 한 예이다. 또한, 특정 투입순서에 의해 일정한 부품 사용률을 갖는 경우도, 작업의 작업장 할당에 의해 가외작업과 작업장 편차가 상충되는 여러 비지배해가 존재할 수 있다. 이는 [그림 11](c)와 (d), <표 3>의 비지배해 C, D로부터 확인할 수 있다. [그림 11](b), (c), (d)로부터 가외작업을 최소화하는 작업 할당과



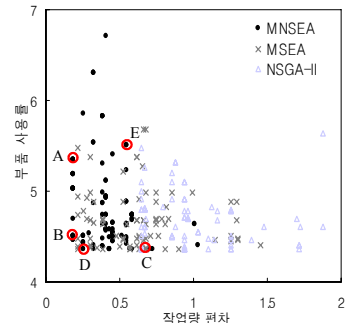
(a) 작업량 편차와 가외작업과 부품 사용률



(b) 작업량 편차와 가외작업



(c) 부품 사용률과 가외작업



(d) 작업량 편차와 부품 사용률

[그림 11] 비지배해의 목적 비교

모델 투입순서는 작업량 편차와 부품 사용률을 각기 최소로 하지 않음을 알 수 있다. 이는 <표 3>의 비지배해 E의 값에서 이를 확인할 수 있다. 여러 실험문제에서도 유사한 결과를 보였다. 일반적으로 이들 세 목적간의 상호 관계는 작업의 선행관계, 각 작업에서 모델별 작업시간의 편차, 생산 모델 수, MPS에서의 모델 비율, 모델별 총 작업량의 편차 등에 의해 영향을 받는다.

<표 3> 비지배해의 예

	작업량 편차	가외작업	부품 사용률
A	0.18	14.5	5.3
B	0.18	20.2	4.4
C	0.72	20.6	4.3
D	0.25	22.6	4.3
E	0.55	7.01	5.5

6. 결 론

이 연구에서는 MMAL의 밸런싱과 투입순서의 다목적 문제에서 잘 분포된 파레토 또는 근사 파레토 최적해를 효과적으로 구할 수 있는 다목적 이웃 공생진화 알고리즘(MNSEA)을 제안하였다. 이 연구에서 제안한 MNSEA는 2계층 구조로 이루어졌으며, 계층 1은 밸런싱 부분 모집단과 투입 순서 부분 모집단이 분리되어 이웃 공생하고, 계층 2는 두 문제가 결합된 통합문제의 모집단이 내공생 진화한다. 이 구조에서 계층 1에서 얻은 비지배해가 계층 2로 전달되어 계층 2에서 진화한다. 진화 알고리즘의 구성 요소로, 다목적 문제에 효율적인 적응도 평가와 선택 방법, 밸런싱과 투입 순서 문제에 적합한 표현, 유전연산자를 제시하였다.

제안한 알고리즘의 성능 평가를 위하여 여러 크기의 실험 문제를 가지고 두 개의 다목적 진화알고리즘과 비교하였다. 실험 결과, 제안한 진화알고리즘(MNSEA)은 비교알고리즘들에 비해 구한 비지배해의 수렴성과 다양성, 그리고 계산소요시간 측면에서 우수함을 보였다. 이로부터 제안한 알고리

즘의 2계층 구조에서 개체가 분리된 형태의 공생진화와 개체가 결합된 형태의 내공생진화의 운영과 이웃 진화가 알고리즘의 성능향상에 효과적임을 알 수 있었다.

제안한 알고리즘의 개념과 구조는 진화알고리즘이 원래 가지고 있는 적용의 유연성에 의해 여러 분야에서 상호 관련된 여러 부분문제로 구성된 복잡도가 높은 다목적 문제를 해결하는데 사용될 수 있다.

참 고 문 헌

- [1] 김여근, 윤복식, 이상복, 『메타휴리스틱』, 영지문화사, 1997.
- [2] 김여근, 이수연, 김용주, “혼합모델조립라인에서 작업부하의 평활화를 위한 유전알고리즘”, 『대한산업공학회지』, 제23권, 제3호(1996), pp.515-532.
- [3] 신경석, 김여근, “다목적 최적화를 위한 공생 진화알고리즘”, 『한국경영과학회지』, 제32권, 제1호(2007), pp.77-91.
- [4] Arcus, A.L., “An analysis of a Computer Method of Sequencing Assembly Line Operations,” Ph.D. dissertation, University of California, 1963.
- [5] Bard, J.F., E.M. Dar-El, and A. Shtub, “An Analytic Framework for Sequencing Mixed Model Assembly Lines,” *International Journal of Production Research*, Vol.30(1992), pp.35-48.
- [6] Burns, L.D. and C.F. Daganzo, “Assembly Lines Job Sequencing Principles,” *International Journal of Production Research*, Vol.25(1987), pp.71-99.
- [7] Coello, C.A.C., G.B. Lamont, and D.A.V. Velthuisen, *Evolutionary Algorithms for Solving Multi-Objective Problems*, Springer, New York, 2007.

- [8] Dar-El, E.M. and A. Navidi, "A Mixed-model Sequencing Application," *International Journal of Production Research*, Vol.19(1981), pp.69-84.
- [9] Davidor, Y., "A Naturally occurring niche and species phenomenon : The Model and First Results," Proceedings of the 4th International Conference on Genetic Algorithms, San Mateo, CA, (1991), pp.257-263.
- [10] Deb, K., S. Agrawal, A. Pratap, and T. Meyarivan, "A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization : NSGA-II," In M.S. et al.(Ed.), *Parallel Problem Solving from Nature-PPSN VI*, Berlin, Springer, (2000), pp.849-858.
- [11] Deb, K., *Multi-Objective Optimization Using Evolutionary Algorithm*, John Wiley and Sons Ltd, Chichester, England, 2001.
- [12] Goldberg, D.E., *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison Wesley, Reading, Massachusetts, 1989.
- [13] Horn, J., N. Nafpliotis, and D.E. Goldberg, "A Niche Pareto Genetic Algorithm for Multiobjective Optimization," *Proceedings of the First IEEE international Conference on Evolutionary Computation*, Vol.1(1994), pp.82-87.
- [14] Kim, J.Y., Y. Kim, and Y.K. Kim, "An Endosymbiotic Evolutionary Algorithm for Optimization," *Applied Intelligence*, Vol.15(2001), pp.117-130.
- [15] Kim, Y.K., C.J. Hyun, and Y. Kim, "Sequencing in Mixed Model Assembly Lines : a Genetic Algorithm Approach," *Computers and Operations Research*, Vol.23(1996), pp.1131-1145.
- [16] Kim, Y.K., J.Y. Kim, and Y. Kim, "A Coevolutionary Algorithm for Balancing and Sequencing in mixed model assembly lines," *Applied Intelligence*, Vol.13(2000), pp.247-258.
- [17] Knowles, J.D. and D.W. Corne, "The Pareto Archived Evolution Strategy : A New Baseline Algorithm for Multi-Objective Optimization," Proceedings of the First IEEE International Conference on Evolutionary Computation, (1999), pp.98-105.
- [18] Macaskill, J.C., "Production-line Balances for Mixed-Model Lines," *Management Science*, Vol.19(1972), pp.423-434.
- [19] Margulis, L., *Symbiosis in Cell Evolution*, W. H. Freeman, San Francisco, 1981.
- [20] Miltenburg, J., "Level Schedules for Mixed-Model Assembly Lines in Just-In-Time Production Systems," *Management Science*, Vol.35(1989), pp.192-207.
- [21] Okamura, K. and H. Yamashina, "A Heuristic Algorithm for the Assembly Line Model-Mix Sequencing Problem to Minimize the Risk of Stopping the Conveyor," *International Journal of production Research*, Vol.17(1979), pp.233-247.
- [22] Potter, M.A., "The Design and Analysis of a Computational Model of Cooperative Coevolution," Ph.D. dissertation, George Mason University, 1997.
- [23] Schaffer, J.D., "Multiple Objective Optimization with Vector Evaluated Genetic Algorithms," In *Genetic Algorithms and their Applications : Proceedings of the First International Conference on Genetic Algorithms*, Lawrence Erlbaum, (1985), pp.93-100.
- [24] Srinivas, N. and K. Deb, "Multiobjective Optimization Using Nondominated Sorting in Genetic Algorithms," *Evolutionary Computation*, Vol.2(1985), pp.221-248.
- [25] Tan, K.C., T.H. LEE, and E.F. Khor, "Evolutionary Algorithms for Multi-Objective Opti-

- mization : Performance Assessments and Comparison,” *Artificial Intelligence Review*, Vol.17(2002), pp.253-290.
- [26] Thomopoulos, N.T., “Line Balancing-Sequencing for Mixed-Model Assembly,” *Management Science*, Vol.14(1967), pp.59-75.
- [27] Thomopoulos, N.T., “Mixed Model Line Balancing with Smoothed Station Assignment,” *Management Science*, Vol.16(1970), pp.593-603.
- [28] Veldhuizen, D.A.V. and G.B., Lamont, “Multiobjective Evolutionary Algorithms : Analyzing the State-of-the-art,” *Evolutionary Computation*, Vol.8, No.2(2000), pp.125-147.
- [29] Zitzler, E. and L. Thiele, “Multiobjective Evolutionary Algorithms : A Comparative Case Study and the Strength Pareto Approach,” *IEEE Transactions on Evolutionary Computation*, Vol.3, No.4(1999), pp.257-271.
- [30] Zitzler, E., “Evolutionary Algorithms for Multiobjective Optimization : Methods and Applications,” Dissertation, Swiss Federal Institute of Technology(ETH) Zurich, 1999.
- [31] Zitzler, E., M. Laumanns, and L. Thiele, “SPEA2 : Improving the Strength Pareto Evolutionary Algorithm,” Technical Report 103, Computer Engineering and Networks Laboratory(TIK), *Swiss Federal Institute of Technology(ETH) Zurich*, Gloriastrasse 35, CH-8092 Zurich, Switzerland, 2001.