

진화프로그램에 기반을 둔 혼합모델 조립라인의 투입순서를 위한 대화형 다목적 의사결정 기법*

김여근 · 이수연

전남대학교 산업공학과

An Interactive Multi-objective Decision Making Technique for Sequencing Mixed Model Assembly Lines Based on Evolution Programs

Yeo Keun Kim · Soo Yeon Lee

A mixed model assembly line (MMAL) is a special type of production line where a variety of product models similar in product characteristics are assembled. Determining the model sequence is an important problem for the efficient use of MMALs. This paper considers interactive multiobjective decision making problems for MMAL sequencing. Evolution program is employed as an underlying framework. In this study, a way of approximating the linear utility function is first studied. To improve its search efficiency to the solution space preferred by a decision maker, some modifications of a standard evolution program are made: operating several subpopulations instead of a single population and merging two or more subpopulations to a single subpopulation, and using a Pareto pool. Extensive computational experiments are carried out to verify the performance of the proposed approach. The computational results show that our approach is promising in solution quality.

1. 서론

혼합모델 조립라인은 공장부지, 시설 등의 투자를 줄이고 고객의 다양한 요구를 만족시키기 위하여 하나의 라인에서 유사한 여러 모델의 제품을 생산하는 조립라인이다. 자동차나 가전제품의 조립라인이 그 대표적인 예이다. 이러한 조립라인에서 모델의 투입순서는 라인의 효율적 이용과 관리 그리고 생산성에 크게 영향을 준다(Bard *et al.*, 1992; Okamura and Yamashina, 1979).

혼합모델 조립라인의 투입순서에 관한 연구는 크게 조립라인의 효율적인 이용과 부품의 일정 사용률 유지에 중점을 두어 왔다(Bard *et al.*, 1994). 라인의 효율적인 이용을 위하여 컨베이어 정지위험의 최소화(Okamura and Yamashina, 1979), 라인길이의 최소화(Dar-El, 1978), 총 가외작업(utility work)의 최소화(Yano and Rachamadugu, 1991), 작업준비비용의 최소화(Burns

and Daganzo, 1987) 등을 목적으로 사용하였으며, Just In Time (JIT) 생산시스템에서는 최소의 적정 재고 유지를 위하여 부품 사용의 일정률 유지를 목적으로 사용하였다(Miltenburg, 1989; Monden, 1983). 이와 같은 여러 목적 가운데 총 가외작업의 최소화, 라인길이의 최소화, 컨베이어 정지위험의 최소화는 거의 동등한 것으로 알려져 있다(Tsai, 1995). 그러나 목적들간에 상충이 일어날 수 있다. 예로 가외작업을 최소로 하는 투입순서는 조립라인에서 사용되는 부품의 일정률을 유지하기 어려울 수 있으며, 한편 부품의 일정 사용률 유지의 목적은 더 긴 라인 길이를 요구할 수 있다. 본 연구에서는 다목적으로 총 가외작업의 최소화와 부품사용의 일정률 유지의 두 가지 목적을 다루고자 한다.

본 연구에서는 혼합모델 조립라인의 다목적 투입순서 문제를 다룬다. 다목적 문제를 위한 여러 기법들이 개발되어 왔다. 이러한 기법들의 목표는 여러 대안들 중에서 의사결정자의 선호에 가장 적합한 하나 또는 몇 개의 해를 제시하는 데 있다고

* 본 연구는 1998년도 한국과학재단 핵심전문 연구비(과제번호: 981-1014-078-1) 지원에 의한 결과임

할 수 있다. 기존의 많은 연구들은 대안들이 주어진 경우를 다루고 있다. 그러나 다루는 문제와 같이 해공간에서 대안을 탐색해야 하는 경우, 의사결정자가 선호하는 해공간을 어떻게 효율적으로 탐색할 것인가가 중요한 문제가 된다. 이를 위하여 본 연구에서는 진화프로그램에 기반을 둔 대화형 다목적 의사결정 기법을 개발하여 혼합모델 조립라인의 다목적 투입순서 문제에 적용하고자 한다.

진화프로그램(evolution program)에는 유전알고리즘(genetic algorithm), 진화전략(evolution strategy), 진화프로그래밍(evolutionary programming) 등이 있다. 이들 기법들간에는 해의 표현, 유전연산, 재생산방법 등에서 약간의 차이는 있으나 기본적으로 모두 자연의 진화과정을 모방한 일종의 확률적 탐색기법이다. 진화프로그램은 조합최적화 문제를 푸는 데 아주 효율적임이 입증되었으며, 특히 다목적 문제에 진화프로그램의 적용에 관한 많은 연구가 이루어졌다(Goldberg, 1989; Greenwood et al., 1996; Horn et al., 1994; Hyun et al., 1998; Murata and Ishibushi, 1995; Schaffer, 1985; Srinivas and Deb, 1995). 그러나 기존의 연구는 대부분 의사결정자의 선호(preference)를 반영하지 않고, 다만 다양한 비지배해(nondominated solution)를 찾는 데 중점을 두어왔다. 어떤 해보다 모든 목적에 대해 더 좋은 다른 해가 존재하지 않으면, 이러한 해를 비지배해라 한다. 비지배해가 아닌 해를 지배해라 한다. 의사결정자의 효용함수를 알 수 있다면 쉽게 이를 최대로 하는 대안을 구하면 될 것이다. 그러나 이 효용함수를 정확히 파악하기는 쉽지 않다. 따라서 분석자는 의사결정자로부터 얻은 제한된 정보를 가지고 효과적으로 의사결정자가 원하는 대안을 찾는 방법이 요구된다.

본 연구에서는 의사결정자가 가장 선호하는 투입순서의 결정을 위해 의사결정자의 선호정보를 대화를 통하여 효과적으로 얻는 방법과 이 선호정보를 이용하여 의사결정자의 선호가 반영된 대안들을 효과적으로 탐색할 수 있는 진화프로그램을 개발한다. 이를 위하여 먼저 의사결정자의 선호정보를 반영한 대안의 평가방법을 제시한다. 그리고 복수개의 부분모집단을 운영함으로써 초기에는 다양한 해공간을 탐색하고, 세대가 진행되면서 의사결정자가 선호하는 해 영역을 집중적으로 탐색하기 위한 방법을 개발한다. 또한 실험을 통하여 제안한 기법의 성능을 분석하고자 한다.

2. 혼합모델 조립라인의 투입순서 문제

혼합모델 조립라인은 일정한 속도(v_i)로 이동하는 컨베이어 시스템으로 유사한 여러 모델의 제품들이 일정시간 간격으로 투입되고 있으며, 컨베이어의 이동에 따라 작업자가 이동하면서 작업하는 라인으로 본다. 라인은 J 개의 폐쇄작업장으로 구성되어 있고, 각 작업장은 한 사람의 작업자로 운영된다. 폐쇄작업장은 작업을 위한 설비나 장비의 제약으로 인한 경계가 존재하여 작업 지연으로 후행 작업장에서 작업을 하거나 선행

작업장에서 미리 작업하는 것이 불가능한 작업장이다. 작업자의 이동시간은 무시되며, 각 작업장에는 이미 작업할당이 이루어 졌다고 본다.

혼합모델 조립라인의 설계시 고려되는 요소로는 작업자의 작업계획(operator schedule), 사이클당 생산제품의 수, 제품의 투입시간 간격 등이 있다(Bard et al., 1992). 첫째, 작업자의 작업계획은 빠른 작업계획(early start schedule)을 사용한다(Dar-El, 1978). 이 계획에서는 각 작업장에 기준점(reference point)을 두어 이 위치 앞으로 작업자가 이동하여 작업을 할 수 없도록 계획된다. 이 계획은 작업자의 유휴 시간을 허용한다는 단점이 있으나, 작업혼잡을 피할 수 있고 작업에 소요되는 라인 길이를 줄일 수 있다는 장점이 있다. 둘째, 사이클 생산(cyclic production)을 한다. 혼합모델 조립라인에서 계획기간(planning horizon)동안 M 종류의 모델이 생산된다고 하자. 그리고 각 모델의 생산개수를 각각 D_1, \dots, D_M 이라고 하고, D_1, \dots, D_M 의 최대공약수를 b 라 하자. 각 모델의 수요 $(D_1/b, \dots, D_M/b) = (d_1, \dots, d_M)$ 를 최소 부품집합(minimum part set: MPS)이라고 한다(Bard et al., 1992). 최소 부품집합은 한 생산사이클동안 생산되는 모델 수의 집합을 나타낸다. 한 사이클 동안 생산되는 제품의 총 수, $I = \sum_{m=1}^M d_m$ 이 된다. 셋째, 제품의 투입간격(launch interval) γ 는 1회의 MPS를 생산하는데 요구되는 총 작업시간을 제품의 총 수와 작업장 수의 곱으로 나눈 값으로 둔다.

앞에서 언급했듯이, 본 연구에서 다목적으로는 총 가외작업의 최소화와 부품사용의 일정률 유지를 채용한다. 가외작업은 각 제품에 대하여 고정된 길이의 작업장 내에서 작업이 완료되지 못하여 작업완료 위치가 작업장의 경계를 넘어설 때 발생한다. MPS 생산기간 동안의 가외작업 최소화는 식 (1)과 같이 표현된다.

$$\text{Min } g_1 = \sum_j \left\{ \sum_i \max \{ 0, (Z_{ij} + v_i \cdot T_{jm} - L_j) / v_i \} + Z_{(i+1)j} / v_i \right\} \quad (1)$$

여기서 Z_{ij} 는 작업장 j 에서 투입순서가 i 번째 제품의 작업시작 위치를 나타낸다. T_{jm} 는 작업장 j 에서 투입순서가 i 번째인 제품 m 의 작업시간, L_j 는 작업장 j 의 작업장길이, 그리고 v_i 는 컨베이어 속도를 나타낸 것이며 작업자의 이동시간은 무시한다. 식 (1)의 첫째 항은 작업시작 위치로부터 조립에 필요한 라인길이를 더한 값이 작업장길이를 초과하는 경우만 계산된다. 그리고 이를 컨베이어 속도로 나누어 시간을 척도로 두었다. 식 (1)의 둘째 항은 각 작업장마다 마지막으로 투입되는 제품에 의해 발생하는 가외작업을 나타낸다. 투입순서가 마지막인 제품의 가외작업은 다음 사이클을 기준점에서 시작해야 하기 때문에 필요한 것이다. 그리고 각 작업장에서 첫 번째 투입제품은 항상 기준점에서 시작된다고 본다. 즉, $Z_{1j} = 0, j = 1, 2, \dots, J$ 로 둔다. 빠른 작업계획을 사용하므로 작업장 j 에서 투입순서에 따른 각 제품의 작업시작 위치는 식 (2)와 같이 된다.

여기서 $\eta (= \gamma v_i)$ 는 투입시간간격 동안 컨베이어의 이동거리

이다.

$$Z_{(i+1)j} = \max [0, \min (Z_{ij} + v_c \cdot T_{jmi} - \eta, L_j - \eta)], \quad i=1, 2, \dots, l \quad (2)$$

<그림 1>은 앞에서 설명한 폐쇄작업장에서의 가외작업과 작업자의 유희시간을 보여주고 있다.

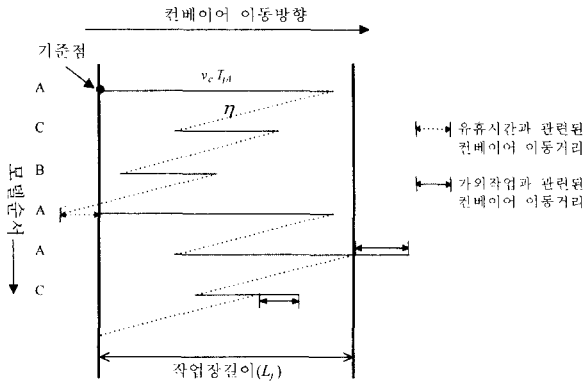


그림 1. 폐쇄작업장에서의 가외작업.

혼합모델 조립라인에서 부품사용의 일정률을 유지하기 위한 목적은 제품생산에 따른 각 모델별 실제 생산비율과 수요 비율의 차를 최소로 함으로써 달성할 수 있다. 부품사용의 일정률 유지를 위하여 Miltenberg(1989)가 제시한 아래 식 (3)을 사용한다.

$$\text{Min } g_2 = \sum_{i=1}^l \sum_{m=1}^M \left| \frac{x_{im}}{i} - \frac{d_m}{I} \right| \quad (3)$$

여기서 x_{im} 은 1번째부터 i 번째까지 투입된 제품 중에서 모델 m 이 투입된 총 수를 나타낸다. 식 (3)에서 절댓값 안의 첫째 항은 i 번째 투입순서까지 생산한 모델의 생산비율을 나타내고, 둘째 항은 수요비율을 나타낸다.

3. 대화형 다목적 의사결정 문제를 위한 진화프로그램

진화프로그램을 이용하여 다목적 문제를 위한 연구는 1984년 Schaffer(1985)에 의해 시작되어, Goldberg(1989), Horn *et al.*(1994), Hyun *et al.*(1998) 등의 여러 연구가 이루어졌다. 이들 연구는 유효경계상(efficient frontier)의 다양한 비지배해를 구하는 데 중점을 두고 있다. 진화알고리즘을 이용하여 의사결정자의 선호가 반영한 연구로는 Greenwood *et al.*(1996)이 있다. 이 연구(Greenwood *et al.*, 1996)에서는 해를 탐색하기 전에 의사결정자의 선호정보를 제공받아, 이를 진화프로그램의 적응도 함수에 반영하여 해공간을 탐색한다. 이와 같이 특정 선호정보에 의한 탐색은

비선형 효용함수인 경우에 해공간의 변화에 따라 의사결정자의 선호를 반영하는 데 한계를 갖는다.

다목적 조합최적화 문제에서 목적의 수와 문제크기가 커지면 비지배해는 아주 많이 존재하게 된다. 다양한 비지배해를 찾는 데는 많은 노력이 요구되고, 이들 많은 비지배해들의 제시의 의사결정자의 선택에 많은 부담을 줄 뿐만 아니라 올바른 의사결정을 방해할 수 있다. 따라서 본 연구에서는 의사결정자가 선호하는 대안들이 존재하는 해공간을 효율적으로 탐색할 수 있는 진화알고리즘을 제시한다. 여기서 비지배해란 생성된 해들에서 지배되지 않는 해를 의미하는 것으로, 실제 비지배해(real nondominated solutions)를 의미하는 것은 아니다. 앞으로 비지배해란 구한 해들 중에서 비지배해를 의미한다.

다목적성을 갖는 혼합모델 조립라인의 투입순서 문제는 조합 최적화 문제로 문제크기가 크거나 목적이 많을 수록 아주 많은 대안(가능해)이 존재하게 된다. 따라서 이들 대안 중에서 의사결정자의 선호를 반영한 최선해를 효과적으로 구하기 위해서는 (1) 의사결정자의 선호정보를 효과적으로 추출해야 하고, 이로부터 (2) 좋은 대안들의 효율적인 탐색을 할 수 있어야 한다. 본 연구에서는 전자를 위하여 대화에 의해 의사결정자의 효용함수를 유도하는 방법을 제안하고, 후자를 위하여 다루는 문제에 적합한 진화프로그램을 개발하고자 한다.

3.1 근사 선형 효용함수

의사결정자의 효용함수를 추론하기 위하여 먼저 대화를 통해 의사결정자의 선호정보를 추출한다. 대화형 의사결정 기법에서 선호정보를 획득하는 방법은 두 대안간의 비교(paired comparison of alternatives)와 주어진 한 점에 대한 상쇄질문(trade-off questions) 방법으로 나눌 수 있다. 두 대안간의 비교는 의사결정자에게 두 대안에 대해 특정대안의 선호 또는 동등 등을 알아보는 것이고, 주어진 한 점에 의한 질문방법은 주어진 점과 다른 한 점을 제시하여 이를 비교하여 상쇄관계의 여부 또는 동등에 관해 알아보는 방법이다. 해 공간이 이산적인 경우에는 전자의 방법을, 그리고 연속적인 경우에는 후자의 방법이 흔히 사용된다. 따라서 본 연구에서는 두 대안간의 비교방법을 사용한다.

한편, 다목적 문제에서 각 목적들은 흔히 서로 다른 척도(scale)를 가지고 있다. 따라서 이들 목적들을 비교 가능한 척도로 만들기 위한 정규화(normalizing)가 필요하다. 본 연구에서는 각 목적에 대해 다음과 같이 정규화 한다. g_k^{\max} 와 g_k^{\min} 를 목적 k 에 대해 각각 가장 큰 값과 가장 작은 값으로 두자. 그리고 $g_k(p)$ 를 대안 p 의 목적 k 의 함수값이라 하자. 목적 k 가 최소화이면 식 (4)와 같이, 최대화이면 식 (5)과 같이 정규화 한다.

$$f_k(p) = \frac{g_k(p) - g_k^{\max}}{g_k^{\min} - g_k^{\max}} \quad (4)$$

$$f_k(p) = \frac{g_k(p) - g_k^{\min}}{g_k^{\max} - g_k^{\min}} \quad (5)$$

대화에 의한 선호정보로부터 의사결정자의 효용함수를 정확히 유도하는 것은 쉽지 않다. 특히, 다루는 문제와 같이 대안들이 한정된 수로 주어지지 않고, 넓은 해공간에서 탐색해야 하는 경우에는 더욱 어렵다. 따라서 본 연구에서는 의사결정자의 효용이 증가하는 방향(direction)으로 대안을 탐색하는 방법을 개발한다.

일반적으로 효용함수는 비선형 함수로 볼 수 있다. 비선형 효용함수는 주어진 어떤 점(대안)에서 가법적 선형 효용함수로 근사시킬 수 있다(Malakooti, 1989). n 개의 목적 (f_1, f_2, \dots, f_n)을 가진 문제에서 주어진 대안 p 에 대해 선형으로 근사한 효용함수는 다음 식(6)과 같이 표현할 수 있다.

$$V(p) = w_1 f_1(p) + w_2 f_2(p) + \dots + w_n f_n(p) \\ \sum_{k=1}^n w_k = 1, \quad (6) \\ w_k \geq 0, \quad k = 1, 2, \dots, n.$$

두 대안비교에 의해 근사 선형 효용함수를 유도할 때 가중치를 결정하는 방법에 관해 여러 연구가 이루어졌다. Bard와 Wambsganss(1991), Malakooti(1989), Srinivasan and Shocker(1991)는 두 대안 쌍들의 비교에서 어느 한 대안의 선호 또는 무차별의 선호 정보를 얻어, 이를 선형 계획법으로 모형화하여 가중치를 구하였다. 이들 연구에서는 의사결정자의 선호정보에 대한 불일치의 정도가 최소가 되도록 하는 가중치를 결정하는 데 목적을 둔 것이다. 선호정보가 일관성이 있는 경우에는 흔히 여러 대안이 존재한다. 이때 이러한 대안집합에서 임의로 하나를 선택하여 가중치를 결정한다.

가중치 벡터는 주어진 점에서의 비선형 효용함수가 증가하는 방향(벡터)으로 해석할 수 있다(Malakooti, 1989). 따라서 의사결정자의 선호조건을 만족하는 가능한 여러 방향(direction) 중에서 임의로 어느 하나를 결정하여 해의 탐색방향으로 선택하는 것은 의사결정자의 선호방향과 서로 다를 수 있다는 위험을 갖는다. 결국 이와 같이 선호제약을 만족하는 가중치가 유일하지 않고 집합으로 존재할 경우 의사결정자의 선호를 보다 정확히 반영할 수 있으면서 적절한 가중치를 결정하기 위한 방법이 필요하다.

이를 위해 본 연구에서는 다음과 같은 방법을 제시한다. 먼저 의사결정자가 제시한 두 대안들간의 비교에 의해 다음과 같은 선호 제약식을 만든다.

$$\sum_{k=1}^n w_k [f_k(p) - f_k(p+1)] \geq \epsilon, \quad (p, p+1) \in \{>\} \quad (7.1)$$

$$\sum_{k=1}^n w_k [f_k(q) - f_k(q+1)] \leq \delta \\ (q, q+1) \in \{\sim\} \quad (7.2)$$

$$\sum_{k=1}^n w_k [f_k(q) - f_k(q+1)] \geq -\delta$$

$$\sum_{k=1}^n w_k = 1 \quad (7.3)$$

$$w_k \geq 0, \quad k = 1, 2, \dots, n \quad (7.4)$$

$$\epsilon \geq 0, \quad \delta \geq 0 \quad (7.5)$$

여기서 δ 는 임의의 작은 양의 실수를 의미한다. 두 대안 (a, b)의 비교에서 $\{>\}$ 는 a 가 b 보다 선호되는, $\{\sim\}$ 는 두 대안이 동등한 비교대안들의 집합을 나타낸다. 제약식 (7.1)은 p 대안이 ($p+1$)대안보다 선호되는 경우의 제약으로, 효용의 차이가 일정한 값 이상을 갖도록 한 것이다. 제약식 (7.2)는 의사결정자가 q 대안과 ($q+1$)대안에 대해 서로 무차별하다고 제시한 경우를 나타낸다. 이때 본 연구에서는 무차별하다고 제시한 두 대안에 대해 의사결정자가 갖는 효용이 정확하게 같기보다는 두 대안이 갖는 효용의 차이가 아주 적은 경우로 판단한다. 따라서 두 대안의 효용 차이의 절대값이 임의의 작은 수(δ)보다 작도록 제약을 수립할 수 있다.

여기서 식 (7.1)~(7.5)를 제약으로 두고 식 (8)과 식 (9)를 각 목적함수로 하는 LP문제의 해를 각각 벡터 W_1^*, W_2^* 라 하자. 그리고 대안 r 과 대안 s 를 비교 대안들 중에서 의사결정자가 가장 선호하는 대안과 가장 선호하지 않은 대안이라 각각 두자.

$$Max \quad Z = \sum_{k=1}^n w_k [f_k(r) - f_k(s)] \quad (8)$$

$$Min \quad Z = \sum_{k=1}^n w_k [f_k(r) - f_k(s)] \quad (9)$$

W_1^*, W_2^* 는 의사결정자가 가장 선호하는 대안과 가장 선호하지 않은 대안의 효용함수값의 차이를 각각 최대, 최소로 하는 가중치 벡터를 의미한다. 본 연구에서는 식 (10)과 같이 두 가중치 W_1^* 와 W_2^* 의 볼록조합(convex combination)으로 가중치 경계상의 W^* 를 구하고, 이를 선형 효용함수의 가중치로 사용한다.

$$W^* = \lambda W_1^* + (1 - \lambda) W_2^*, \quad 0 \leq \lambda \leq 1 \quad (10)$$

대안 r 과 대안 s 에 대해 의사결정자의 선호정도가 큰 차이를 갖는 경우 λ 는 1에, 그렇지 않은 경우에는 0에 가까운 값을 부여한다. λ 는 의사결정자가 직접 부여하거나, 기준을 마련하여 결정할 수 있다. 두 가중치 W_1^* 와 W_2^* 를 이용하여 의사결정자의 선호강도에 보다 근접한 가중치를 결정할 수 있다.

한편, 의사결정자의 선호에 있어서 불일치가 발생하면 제약식 (7.1)~(7.5)를 만족하는 가능해가 존재하지 않게 된다. 이 경우 본 연구에서는 Bard와 Wambsganss(1991)가 제시한 방법을 아래와 같이 수정하여 적용한다.

$$Min \quad Z = \sum_{(p, p+1) \in \{>\}} IP_{p, p+1} + \sum_{(q, q+1) \in \{\sim\}} ID_{q, q+1} \quad (11.1)$$

$$s.t. \quad \sum_{k=1}^n w_k [f_k(p) - f_k(p+1)] + IP_{p, p+1} \geq \epsilon, \quad (p, p+1) \in \{>\} \quad (11.2)$$

$$\sum_{k=1}^n w_k [f_k(q) - f_k(q+1)] + ID_{q,q+1} \leq \delta \quad (q, q+1) \in \{\sim\}$$

$$\sum_{k=1}^n w_k [f_k(q) - f_k(q+1)] + ID_{q,q-1} \geq -\delta$$

$$\sum_{k=1}^n w_k = 1$$

$$w_k \geq 0, \quad k = 1, 2, \dots, n, \quad \epsilon \geq 0, \quad \delta \geq 0$$

$$ID_{q,q-1} \geq 0, \quad (q, q+1) \in \{\sim\}$$

$$IP_{p,p+1} \geq 0, \quad (p, p+1) \in \{>\}$$

이 방법은 불일치 정도를 최소화하는, 즉 의사결정자의 선호조건을 가능한 최대로 만족하는 가중치를 결정하는 것이다.

3.2 대화형 다목적 진화프로그램

효용함수의 형태가 비오목(non-concavity)인 경우, 해공간의 탐색영역에 따라 의사결정자의 선호방향이 다를 수 있으며, 더욱이 최선해는 해공간의 서로 다른 영역에 여러 개가 존재할 수 있다(Greenwood et al., 1996). 따라서 탐색초기에는 가능한 다양한 해 공간을 탐색될 수 있도록 하고, 탐색노력을 줄이기 위하여 탐색이 진행될수록 탐색영역을 좁혀 갈 필요가 있다. 이를 위하여 본 연구에서 진화알고리즘에 기반을 둔 효율적인 탐색기법을 제안한다. 제안한 진화알고리즘에서는 복수 개의 부분모집단을 운영하고, 세대가 진행되면서 유사한 대안을 유도하는 모집단은 병합하는 전략을 사용한다. 그리고 의사결정자가 선호하는 비지배해를 다양하게 구하고 이를 유지하기 위하여 비지배해 풀(pool)을 사용한다. 이에 관해서 구체적으로 다루기로 한다.

3.2.1 복수 부분모집단과 종료조건

다양한 해공간에 대한 의사결정자의 선호정보를 탐색하기 위하여 본 연구에서는 복수 개의 부분모집단을 운영한다. 부분모집단은 적소만들기(niching)의 효과(Goldberg, 1989)를 가져올 수 있다. 반면에 복수 개의 부분모집단을 운영함으로써 의사결정자와의 대화수가 많아지며, 탐색이 이루어질수록 같은 해공간을 서로 다른 부분모집단이 동시에 탐색할 수 있다는 단점을 갖는다. 이러한 문제를 해결하기 위해 본 연구에서는 일정조건을 만족하면 부분모집단들을 하나로 병합시키는 전략을 사용한다.

둘 이상의 부분모집단에서 탐색결과가 유사하여 더 이상 분리하여 탐색할 필요가 없을 때 부분모집단간 병합이 이루어진다. 두 부분모집단간의 여러 비교대안들에 대해 유사성을 고려할 수 있으나, 본 연구에서는 각 부분모집단에서 의사결정자가 가장 선호하는 각 대안에 대해서만 유사성을 고려한다. 유사성(similarity)의 척도로 L_τ 함수(Graves et al., 1992)를 사용한다. 이 방법에서 대안 p 와 q 에 대한 유사성(L_τ)은 다음과 같이

측정한다.

$$L_\tau = \left\{ \sum_{k=1}^n (|f_k(p) - f_k(q)|) \right\}^{\frac{1}{\tau}} < \sigma \quad (12)$$

여기서 τ 는 흔히 사용되는 2로 두었다. 이는 두 대안에 대한 목적함수값들의 직선거리를 의미한다. 그리고 허용거리, σ 는 예비실험에 의해 적절히 주어진 값으로 1.0으로 두었다. 둘 이상의 부분모집단이 병합되는 경우, 병합된 부분모집단들에서 부분모집단의 크기만큼 개체를 선별하였다. 이때 선별은 병합되는 부분모집단이 갖는 가중치들의 평균을 가중치로 사용한 근사 선형 효용함수에 의해 좋은 개체들을 선택하였다.

제안한 기법에서 종료조건으로는 복수 개의 부분모집단을 운영하고, 의사결정자와의 대화과정을 포함하기 때문에 (1) 모든 부분모집단에서 진화된 세대수, (2) 모든 부분모집단에서 생산된 개체 수 등과 같이 흔히 사용되는 종료조건 이외에, (3) 총 대화의 회수 등이 사용될 수 있다. 그리고 실제 운용에 있어서는 (4) 의사결정자가 만족하는 최선해를 찾으면 당연히 종료될 것이다.

3.2.2 비지배해 풀

각 부분모집단에 존재하는 비지배해는 다른 부분모집단에 존재하는 어떤 비지배해에 의해 지배될 수 있다. 또한 각 부분모집단 내에서만 진화가 이루어지면 부분모집단은 부분최적에 조기 수렴할 가능성이 높아진다. 이를 극복하기 위하여 본 연구에서는 비지배해 풀(pool)을 운영한다. 이 운영방법은 다음과 같다. 먼저 각 부분모집단의 매 세대마다 자신의 부분모집단에서 비지배해를 구하여 이들을 비지배해 풀에 넣는다. 다음은 이 풀에 있는 어떤 해에 지배당하는 해를 제거하여, 이 풀이 비지배해만으로 구성되도록 한다. 이렇게 풀을 갱신하여 가면서 다음 세대의 각 부분모집단을 만들 때, 이 풀에서 가장 좋은 일부 개체를 선별하여 부분모집단의 일부 개체들이 되도록 한다. 그리고 나머지 개체는 자신의 부분모집단에서 적응도에 따라 선별한다. 다음 <그림 2>는 비지배해 풀의 운영과정을 보여주고 있다.

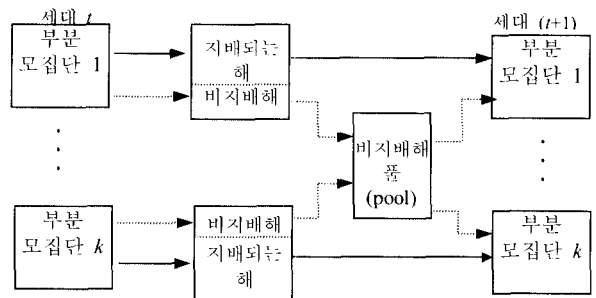


그림 2. 파레토 풀(pool)의 운영절차.

3.2.3 진화프로그램의 전체 절차

본 연구에서 제시한 기법은 개체(대안)를 생산하는 진화프로그램 국면과 의사 결정자로부터 선호정보를 추출하여 효용 함수를 구하는 국면으로 구성되어 있다. 이들 두 국면이 서로 번갈아가며 정보를 주고 받으면서 최선해를 탐색해 간다. 제안한 기법의 전체 절차는 다음과 같다. 아래 절차에서 l 은 부분모집단의 수이고, $P_i(t), i=1, 2, \dots, l$ 은 세대 t 에서의 부분모집단 i 를 나타내며, $|P_i(t)| = N_p, i=1, 2, \dots, l$ 로 둔다.

- 단계 1: (초기해 생성) $N_p \times l$ 개의 개체와, 가중치 벡터 W_1, W_2, \dots, W_l 을 임의로 생성한다.
- 단계 2: (평가) 각 $i=1, 2, \dots, l$ 에 대해, 생성된 가중치 벡터, W_i 를 식 (6)에 대입하여 모든 개체의 적응도를 구한다. 이때 각 개체는 l 개의 적응도를 갖는다.
- 단계 3: (초기 부분모집단 구성) 각 $i=1, 2, \dots, l$ 에 대해, W_i 에 의해 구해진 적응도에 의해 초기모집단에서 개체를 추출하여 부분모집단 $P_i(t)$ 를 구성한다.
- 단계 4: (부분모집단별 해의 탐색) 각 $i=1, 2, \dots, l$ 에 대해, 부분모집단 $P_i(t)$ 를 일정 횟수(세대) 동안 다음을 반복한다.
 - 1) 비지배해를 구하여 비지배해 풀을 갱신한다.
 - 2) 비지배해 풀에서 가장 좋은 $N_{parent} (< N_p)$ 개의 개체를 선별하고, $P_i(t)$ 에서 적응도를 기준으로 $N_p - N_{parent}$ 개의 개체를 선별하여 $P_i(t+1)$ 로 둔다.
 - 3) 교차율과 돌연변이율에 따라 유전연산시킨다.
 - 4) 부분모집단 $P_i(t+1)$ 을 평가한다.
- 단계 5: (종료조건) 종료조건을 만족하면 알고리즘을 종료한다.
- 단계 6: (부분모집단의 병합) 병합조건을 만족한 부분모집단은 하나의 부분모집단으로 병합한다.
- 단계 7: (의사결정자와의 대화) 각 부분모집단별로 적응도가 가장 좋은 대안과 이에 인접한 적은 수의 비지배해를 의사결정자에게 제시하여 이들 대안의 선호순서 정보를 받는다.
- 단계 8: (가중치 벡터 결정) 의사결정자의 선호정보로부터 식 (10) 또는 식 (11)에 의해 새로운 가중치 벡터를 결정한다. 그리고 단계 4로 간다.

단계 1에서 가중치 벡터를 임의로 생성하는 것은 탐색초기에 의사결정자의 선호와 무관하게 해공간을 넓게 탐색하기 위함이다. 단계 2에서는 각 개체에 l 개의 가중치 벡터를 이용하여 각 가중치에 따른 적응도를 구한다. 단계 3에서 각 부분모집단을 구성할 때, 초기모집단에서 적응도에 따라 부분모집단의 크기만큼 추출한다. 이때 초기모집단에 있는 개체는 다른 모집단에 중복하여 선택될 수도 있고 어떤 부분모집단에도 선택되지 않을 수 있다. 단계 4는 각 부분모집단을 진화시키는 과정이다. 이때 비지배해 풀의 갱신에 대해서는 3.2.2절에서 언급하였다. 단계 5에서 종료조건은 3.2.1절에서 언급한 여러 기준이 있으나, 본 논문에서는 의사결정자로부터 얻은 정보량에

기초하기 위하여 의사결정자와의 총 대화수로 하였다. 단계 6의 병합조건은 3.3.2절에서 다루었다. 단계 7에서 의사결정자와의 대화가 이루어지며, 의사결정자가 가장 선호하는 해와 가장 선호하지 않는 해에 대한 선호정도에 관한 정보를 받을 수 있다. 그리고 본 논문에서 비교대안의 수는 5개로 두었다. 단계 8에서 선호순서를 만족하는 가중치 집합이 존재할 경우 식 (10)에 의해, 존재하지 않을 경우에는 식 (11)에 의해 새로운 가중치 벡터를 결정한다.

4. 다목적 혼합모델 조립라인의 투입순서에 진화 프로그램의 적용

4.1 표현방법

혼합모델 조립라인에서 사이클생산에 따른 투입순서를 결정하는 문제는 일종의 순서문제이지만 한 사이클 내에서 같은 모델의 제품이 생산될 수 있다는 점에서 외관원 문제와 같은 전형적인 순서문제와 차이가 있다. 예를 들면 한 사이클 동안의 생산제품이 모델 A, B, C가 각각 2, 3, 4개일 때 투입순서가 A, C, B, C, B, A, C, C, B인 경우 한 사이클 내에서 같은 모델이 다시 나타나므로, 외관원 문제와 특성이 다른 문제이다. 본 연구에서는 가장 자연스런 표현방법으로 사이클생산 내에서 생산되어야 하는 모델들의 목록을 투입순서대로 나열하는 방법을 사용한다. 예로 사이클 생산동안 모델 A, B, C가 각각 2, 3, 4개 생산되어야 한다고 하자. 그리고 하나의 가능해로 투입순서가 A, C, B, C, B, A, C, C, B라면 개체의 표현방법은 모델의 투입순서대로 (A C B C B A C C B)와 같이 표현한다. 초기모집단은 사이클생산 동안 각 모델의 수요량을 만족하는 임의의 해를 사용한다.

4.2 적응도 평가와 선별

본 연구에서 다루는 목적은 가외작업의 최소화와 부품사용율 유지의 두 가지 목적에 대해 의사결정자의 효용을 최대로 하는 대안을 찾는 데 있다. 각 개체에 대한 두 목적은 각각 식 (1)과 식 (3)에 의해 평가된다. 이를 식 (4)에 의해 정규화한 후 식 (10) 또는 식 (11)에 의해 구한 가중치 벡터를 식 (6)에 대입함으로써 의사결정자의 근사효용을 구한다. 이렇게 구한 효용을 각 개체의 적응도로 사용한다.

선별은 각 개체의 적응도에 기초하여 다음 세대의 생존개체를 선택하는 것이다. 선별을 위한 여러 방법이 제안되었으나, 본 연구에서는 토너먼트 선별을 사용한다. 이 선별방법은 개체간의 적응도 차이와 무관하게 사용할 수 있으며, 아주 좋은 개체의 중복선택으로 모집단이 조기 수렴되는 것을 막을 수 있어 모집단의 다양성을 유지할 수 있다는 장점을 갖는다. 이 절차는 다음과 같다. 먼저, 전체 모집단을 임의의 순열로 나열한다. 이 순열을 일정 개수(토너먼트 크기)의 개체로 구성된 블

력으로 나누고, 각 블럭에서 가장 좋은 적응도를 갖는 개체를 선별하여 다음 세대를 구성할 후보개체로 둔다. 모든 블럭에 대해 선별이 이루어지면, 같은 방법으로 모집단의 개체를 다시 임의의 순열로 만들어 블럭으로 나누어 후보개체를 선별한다. 이러한 과정을 모집단의 크기만큼 선별될 때까지 반복한다. 본 연구에서 토너먼트 크기는 흔히 사용되는 2로 두었다.

4.3 유전연산자

유전연산은 생물학의 유전법칙을 모방하여 자손을 생산하는 과정으로, 교차와 돌연변이가 있다. 본 연구에서는 Davis (1985)가 제시한 순서교차(order crossover)를 다루는 문제에 적합하게 수정하였으며, 돌연변이로는 역순(inversion) 연산자를 사용한다. 이들 연산자는 혼합모델의 투입 순서에서 좋은 성능을 나타내었다(Kim et al., 1996). 수정된 순서교차는 다음과 같다.

부모개체 P1과 P2에서 2개의 교차점을 임의의 선택하여 P1의 교차점 사이에 있는 인자들을 그대로 O1에 유전시킨 후에, 유전된 인자들을 P2에서 임의로 지우고 남은 인자들을 순서대로 O1의 아직 나타나지 않은 인자에 유전한다. 예로 <그림 3>과 같이 교차점이 임의로 선택되었다고 하자(<그림 3>에서 ▼는 교차점을 의미한다). P1의 교차점 사이에 있는 인자들을 O1에 그대로 유전하면 O1 = (xxxBBBCxxx)가 된다. 여기서 기호 x는 현재상태에서 모르는 유전인자를 의미한다. O1에 유전된 인자를 P2에서 임의로 제거(P2[2]=B, P2[5]=B, P2[3]=C)하면 P2 = (AxxAxCBCC)가 된다. 이를 순서대로 O1의 아직 나타나지 않은 인자에 순서대로 삽입하면 O1 = (AACBBCCBCC)가 된다. 이 교차에서는 부모 개체가 갖는 인자들의 상대적 순서가 자손에 유지된다.

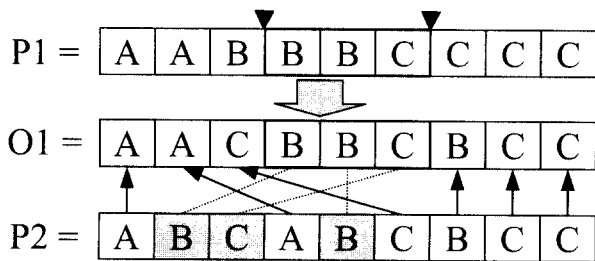


그림 3. 수정 순서교차.

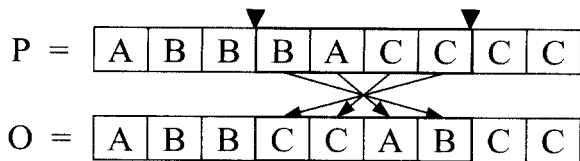


그림 4. 역순연산자.

역순연산자는 한 부모에서 두 개의 절단점을 임의로 선택하고 이 절단점 사이의 인자들을 역순으로 하여 자손을 생산한다. <그림 4>는 역순의 예를 보여주고 있다.

5. 실험 및 분석

제안한 알고리즘의 성능분석을 위하여 먼저, 예비실험을 통하여 파라미터를 결정하였다. 그리고 제안된 알고리즘의 탐색 성능을 입증하기 위하여 기존 알고리즘들과 성능을 비교하였다. 제안한 알고리즘은 C++프로그램 언어로 구현되었으며, 166MHz Pentium CPU를 장착한 IBM-PC에서 수행되었다. 비교 분석의 신뢰성을 높이기 위해 각 문제마다 10회를 반복 실험하여, 매회 가장 좋은 해의 평균치를 사용하였다.

5.1 실험문제와 파라미터 설정

실험을 위하여 혼합모델 조립라인은 다음과 같이 이루어졌다고 본다. 총 12개의 폐쇄작업장으로 구성되어 있고, T_m 은 $20 \leq T_m \leq 40$, (모든 j, m 에 대하여) 범위를 갖는 일양분포(uniform distribution)로부터 추출하였다. 컨베이어의 속도(v_i)는 계산상의 편의를 위해 1로 두었다. 실험문제는 MPS 생산사이클 동안 생산되는 제품 수, I , 모델의 수, M 그리고, MPS에 의해 분류하였다. 이는 <표 1>과 같다.

표 1. 성능평가를 위한 문제 설계

Problems	I	M	MPS
S-1	20	10	(1,1,3,1,3,4,2,1,3,1)
S-2	20	10	(2,1,3,1,4,3,1,3,1,1)
M-1	30	15	(1,1,2,1,2,2,1,4,1,3,5,1,3,1,2)
M-2	30	15	(2,1,3,1,5,3,1,4,1,2,2,1,2,1,1)
L-1	40	20	(3,2,2,1,5,2,1,3,1,2,1,3,2,2,1,1,2,1,3,2)
L-2	40	20	(3,1,4,3,5,2,1,2,1,2,1,2,1,2,2,1,1,3,2,1)

제안한 기법에서 파라미터로는 부분모집단의 수와 크기, 교차율, 돌연변이율, 의사결정자와의 대화주기 그리고 종료조건이 있다. 예비실험에 의해 얻은 파라미터의 값은 다음과 같다. 부분모집단의 수와 그 크기는 5.2절에서 구체적으로 다룬다. 선별은 비지배해 풀과 자신의 부분모집단에서 각각 10%와 90%를 추출하였다. 모든 실험에서 교차율과 돌연변이율은 각각 0.7과 0.3으로 두었다. 의사결정자와의 대화는 각 부분모집단을 40세대 진화시킨 후 행하였다. 그리고 종료조건은 모든 부분모집단에서 행하여진 대화의 수를 사용하였다. 대화의 총 수는 12회로 두었다.

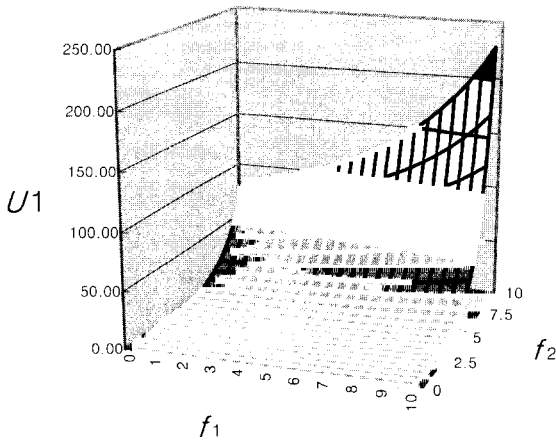
의사결정자의 효용함수를 정확히 안다면 진화프로그램에서 이를 평가함수로 사용하여 최선해를 탐색할 수 있다. 본 연

구에서는 효용함수를 알지 못하는 상황에서 의사결정자로부터 얻은 부분선호정보로부터 근사적인 효용함수를 추론한다. 이 근사 효용함수를 평가함수로 사용하고 있다. 실험을 위하여 아래와 같은 두 개의 효용함수를 사용하였다. 물론, 분석자는 이 효용함수를 알지 못하고, 의사결정자는 제시된 대안들을 비교할 때 묵시적으로 이 효용함수에 의해 선호를 판단하여, 대안의 선호순서만을 분석자에게 제공한다고 본다.

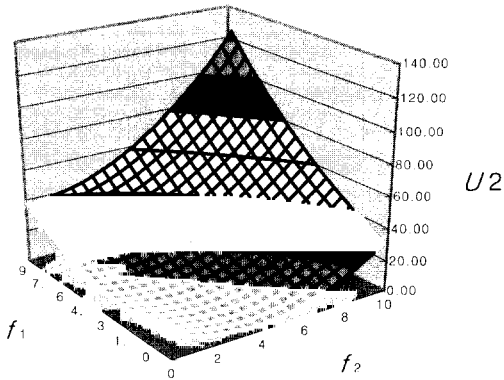
$$\text{효용함수 1: } U1(f_1, f_2) = \frac{(f_1 - 1)^4 + 3(f_2 - 2)^2 + f_1 \cdot f_2}{80 - 0.5f_1 \cdot f_2}$$

$$\text{효용함수 2: } U2(f_1, f_2) = 0.7(f_1 - 4)^2 + 0.4(f_2 - 2)^2 + 0.7f_1 \cdot f_2$$

여기서 f_1 과 f_2 는 각각 총 가외작업, g_1 과 부품사용률, g_2 를 식 (4)에 의해 정규화한 후 10을 곱한 값이다. <그림 5>는 이들 효용함수의 형태를 보여준다.



(a) 효용함수 1의 형태



(b) 효용함수 2의 형태
그림 5. 효용함수의 형태.

5.2 복수 부분모집단의 운영효과 분석

먼저 복수 부분모집단의 운영효과를 분석하기 위하여, 부분모집단의 수가 1개, 2개, 3개, 그리고 5개인 경우를 실험하였다. 모든 문제에서 전체 모집단의 크기는 일정하게 300으로 두고 각 부분모집단의 크기는 같게 두었다. 예로 5개의 부분모집단인 경우 그 크기는 60으로 두었다. 의사결정자와 대화는 40세대마다 행하였으며 총 대화횟수가 12회가 되면 종료하였다. 해의 벤치마크를 위하여, 의사결정자의 효용함수를 적용도함수로 사용하여 충분한 탐색을 수행(2,000세대 동안 진화)하였으며, 이를 10회 반복하여 가장 좋은 해를 구하였다. 이때 300개의 단일 모집단을 갖는 표준 진화알고리즘을 사용하였다. <표 2> (a)는 이를 보여준다.

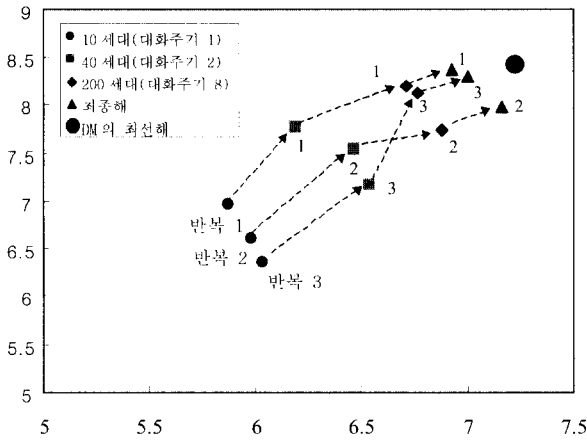
<표 2> 에서 (b)~(e)는 비지배해 풀을 사용하고 부분모집단의 수가 1, 2, 3, 5개 인 경우, 각 실험문제에서의 효용을 나타낸 것이다. 실험결과, 모든 문제에서 단일 모집단보다 복수 개의 부분모집단을 운영할 때 더 좋은 결과를 보였다. 부분모집단의 수가 3개일 때, S-1과 L-2를 제외한 모든 문제에서 가장 좋은 해를 탐색하였다. 여기서 부분모집단의 수가 많고 충분한 대화와 탐색이 이루어지면 다양한 해 공간에 대한 탐색과 함께 (근사)전체 최선해에 대한 효율적인 탐색이 이루어질 수 있다. 그러나 부분모집단의 수가 많을수록 의사결정자와의 대화수가 많아지고 많은 계산시간이 요구된다. 따라서 부분모집단의 수는 의사결정 상황에 따라 적절히 결정되어야 한다. 실험에서 대화수를 12회로 한정하였기 때문에, 5개의 부분모집단의 경우에는 각 부분모집단이 의사결정자의 선호를 반영한 충분한 탐색을 하지 못한 것으로 보인다. 5.3절의 성능실험에서 부분모집단의 수와 크기는 각각 3과 100으로 둔다.

앞의 실험에서는 모두 비지배해 풀을 사용하였다. <표 2> (f)는 부분모집단의 수가 3개일 때 비지배해 풀을 사용하지 않은 경우의 실험결과를 나타내고 있다. 이로부터 비지배해 풀의 사용은 이를 사용하지 않는 경우보다 좀더 좋은 해를 유도할 수 있음을 알 수 있다.

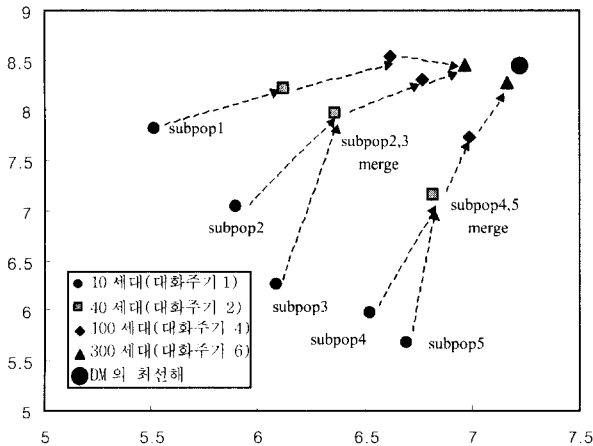
제한한 기법에서 세대수의 증가에 따른 각 부분모집단의 탐색과정을 분석하고자 한다. <그림 6> (a)는 단일모집단을 3회 반복 실험한 결과이고, <그림 6> (b)는 5개의 부분모집단을 운영한 경우 각 부분모집단의 탐색과정을 나타낸 것이다. 단일모집단의 경우 해 공간의 일부 영역만을 탐색하는 데 반해, 복수 개의 부분모집단의 경우 탐색초기에는 다양한 해 공간을 탐색하고, 대화가 반복될수록 최선해가 존재하는 영역을 집중적으로 탐색한다는 것을 알 수 있다. 결국 복수 개의 부분모집단을 운영함으로써 탐색과정에 의사결정자에게 다양한 해 공간에 관한 선호정보를 이용하여 보다 효율적인 탐색이 가능하다는 것을 알 수 있다.

표 2. 복수 부분모집단과 비지배해 풀의 운영에 따른 탐색성능

Utility function	Problems	Using UF (a)	1 Pop		2 Subpops		3 Subpops		5 Subpops		3 Subpops (No Pareto pool)	
			Utility (b)	Eff. (b/a)	Utility (c)	Eff. (c/a)	Utility (d)	Eff. (d/a)	Utility (e)	Eff. (e/a)	Utility (f)	Eff. (f/a)
U1	S-1	126.6	96.2	0.76	119.0	0.94*	117.7	0.93	111.4	0.88	113.9	0.90
	S-2	138.2	120.2	0.87	128.5	0.93	131.3	0.95*	123.0	0.89	123.0	0.89
	M-1	121.9	101.2	0.83	110.9	0.91	113.4	0.93*	106.1	0.87	107.3	0.88
	M-2	138.7	109.5	0.79	123.4	0.89	126.2	0.91*	119.2	0.86	108.1	0.78
	L-1	134.6	113.1	0.84	127.9	0.95	129.2	0.96*	113.1	0.84	109.0	0.81
	L-2	129.2	112.4	0.87	119.5	0.93*	118.8	0.92	111.1	0.86	104.7	0.81
U2	S-1	84.4	65.8	0.78	79.9	0.95*	79.4	0.94	77.6	0.92	76.4	0.91
	S-2	85.0	69.7	0.82	74.8	0.88	78.7	0.93*	75.7	0.89	74.0	0.87
	M-1	94.3	70.7	0.75	82.0	0.87	84.9	0.90*	81.1	0.86	81.1	0.86
	M-2	97.3	81.7	0.84	89.5	0.92	93.4	0.96*	78.8	0.81	91.4	0.94
	L-1	89.3	66.1	0.74	81.7	0.92	84.9	0.95*	78.6	0.88	81.3	0.91
	L-2	87.8	69.4	0.79	82.7	0.94*	82.6	0.94	75.5	0.86	81.7	0.93



(a) 단일 모집단의 운영(3회 반복)



(b) 복수모집단의 운영

그림 6. 세대수 증가에 따른 비지배해 탐색과정.

5.3 알고리즘의 성능비교

본 연구에서 제안된 알고리즘인 대화형 진화프로그램(Interactive Evolutionary Program: IEP)의 성능을 분석하기 위하여, 다양한 비지배해를 구하는 것을 목적으로 하는 Vector Evaluated Genetic Algorithm(VEGA)(Schaffer, 1985), Pareto Genetic Algorithm(PGA)(Goldberg, 1989), Niche Pareto Genetic Algorithm(NPGA)(Horn *et al.*, 1994)의 기법들과 의사결정자의 선호가 반영된 Greenwood, Hu와 D'Ambrosio(GHD)(1996)에 의해 제안된 다목적 진화알고리즘과 비교한다.

이들 다목적 문제를 위한 유전알고리즘은 개체의 선별방법이 표준 유전알고리즘과 크게 다르다. VEGA는 목적1의 값이 가장 좋은 개체부터 차례로 나열하여 순위를 주고, 이 순위에 의해 하나의 부분모집단을 만들고, 같은 방법으로 목적 2에 의해 다른 부분모집단을 선별하여 이들을 유전 연산한다. PGA 또한 비지배해 순위에 따라 각 개체의 선별 확률을 부여한다. NPGA에서는 비교집합의 수는 모집단 크기의 10%로 두었다. 그리고 GHD는 초기 모집단에서 3개의 비지배해를 임의로 선택하여 선호정보를 받고, 선호순위에 따라 확률적으로 선별한다. 이들 모두에서 사용된 선별확률은 기하분포로부터 얻었다. 각 기법에서 전체 모집단의 크기는 300으로 두었으며, 관련 파라미터는 제안한 기법에서와 같은 값들을 사용하였다.

<표 3>은 기존 기법과 제안한 기법(IEP)을 실험문제에 적용한 결과이다. <표 3>(a)는 <표 2>의 (a)와 같다. 비교기법의 최선해는 구한 비지배해 중에서 효용함수값이 가장 좋은 대안을 나타낸 것이다.

실험결과, 모든 실험문제에서 IEP가 비교기법들 보다 더 좋은 성능을 보였다. 이는 앞에서 언급하였듯이 기존 기법들은 다양한 비지배해를 생성하는 데 목적을 두고 있기 때문에 의사결정자가 선호하지 않는 비지배해를 생성하는 데 많은 노력

표 3. 제안한 기법의 최종해에 대한 탐색성능 비교

Utility Function	Problem	Using UF (a)	VEGA		ParetoGA		NPGA		GHD		IEP	
			Utility (b)	Eff. (b/a)	Utility (c)	Eff. (c/a)	Utility (d)	Eff. (d/a)	Utility (e)	Eff. (e/a)	Utility (f)	Eff. (f/a)
U1	S-1	126.6	100.0	0.79	108.8	0.86	112.6	0.89	115.2	0.91	117.7	0.93*
	S-2	138.2	114.7	0.83	116.3	0.84	115.8	0.84	96.0	0.70	131.3	0.95*
	M-1	121.9	90.2	0.74	93.9	0.77	92.6	0.76	85.1	0.70	113.4	0.93*
	M-2	138.7	117.8	0.85	116.5	0.84	115.1	0.83	109.5	0.79	126.2	0.91*
	L-1	134.6	117.1	0.87	117.8	0.88	119.0	0.88	110.4	0.82	129.2	0.96*
	L-2	129.2	104.6	0.81	107.2	0.83	105.9	0.82	98.2	0.76	118.8	0.92*
U2	S-1	84.4	75.1	0.89	70.0	0.83	74.2	0.88	77.6	0.92	79.4	0.94*
	S-2	85.0	74.0	0.87	66.7	0.79	67.3	0.79	78.2	0.92	78.7	0.93*
	M-1	94.3	67.9	0.72	69.8	0.74	73.5	0.78	68.8	0.73	84.9	0.90*
	M-2	97.3	83.6	0.86	81.7	0.84	83.0	0.85	75.8	0.78	93.4	0.96*
	L-1	89.3	78.6	0.88	76.9	0.86	75.6	0.85	81.3	0.91	84.9	0.95*
	L-2	87.8	71.1	0.81	67.6	0.77	66.7	0.76	75.5	0.86	82.6	0.94*

이 소요된다. 반면에 제안한 기법은 의사결정자가 선호하는 비지배해가 존재하는 해공간의 탐색에 노력을 기울임으로써 보다 좋은 해를 생산한다는 것을 보여준다.

한편 비교 기법들 가운데 GHD방법이 많은 문제에서 좋은 탐색효율을 가져왔다. 이는 다른 비교기법들에 비해 GHD방법은 탐색하기 전에 의사결정자로부터 제공받은 선호정보를 이용하여 의사결정자의 최선해가 존재할 수 있는 해 영역을 탐색하기 때문인 것으로 보인다. 그러나 U1의 L-1, L-2, 그리고 U2의 M-2, L-2문제에서 가장 낮은 결과를 가져왔다. 이는 하나의 특정 선호정보에 의한 탐색은 해공간의 변화에 따라 의사결정자의 최선해와 거리가 있는 영역만을 탐색할 위험이 있음을 의미한다. 결국 IEP는 해의 탐색과정에 의사결정자의 선호정보를 지속적으로 반영함으로써 의사결정자가 만족하는 최선해를 효율적으로 탐색할 수 있다는 것을 알 수 있다.

제안된 IEP에서는 근사 선형 효용함수의 가중치를 구하는데, 선형계획문제를 풀어야 한다. 사용되는 선형 계획모형에서 제약식의 수는 변수의 비음제약을 제외할 때, (비교대안의 개수 + 동등한 대안 쌍의 수)개이고, 변수의 수는 많아야(제약식의 수 + 목적의 수 - 1)개이다. 의사결정자의 혼란을 피하기 위하여 비교대안의 수는 많지 않아야 한다. 따라서 현실 문제에 적용하는 데 사용될 수 있는 목적의 개수와 비교대안의 개수를 수용하는 데 많은 계산시간이 소요되지 않을 것으로 본다. 본 연구에서 다른 실험문제의 경우, 선형계획문제를 풀어 가중치를 구하는 데 아주 작은 시간(0.1초 이내)이 소요되었다. 실험에서 선형 계획문제를 푸는 시간을 포함하여 컴퓨터 계산 시간은 21~25초 정도, 그리고 비교기법은 23~26초의 계산시간이 소요되었다. IEP에서 의사결정자와의 대화시간은 제외된 것이다. VEGA, PGA, NPGA는 비지배해를 구하는 데 대안들간의 비교가 이루어져야 하고, GHD와 IEP는 선형 계획문제를 풀어야 한다. 컴퓨터 계산시간은 비교기법들 간에 거의 차이가 없었다.

6. 결론

본 연구에서는 부품사용의 일정률 유지와 총 가외작업의 최소화 두 가지 목적을 갖는 혼합모델 조립라인의 투입순서 문제에서 의사결정자의 최선해를 탐색하기 위한 기법을 개발하였다. 이를 위해 먼저 의사결정자의 선호정보를 이용하여 효용함수를 결정하는 방법을 제시하였다. 그리고 의사결정자의 선호가 반영된 대안을 효과적으로 탐색할 수 있는 진화프로그램의 개발을 위하여 복수개의 부분모집단을 운영하고, 대화가 반복되면서 유사한 대안을 유도하는 모집단을 병합하는 전략을 사용하였다. 또한 부분모집단간 정보교환과 의사결정자가 선호하는 다양한 비지배해를 구하고 이를 유지하기 위한 비지배해 풀을 운영하였다.

실험결과, 복수 개의 부분모집단의 운영과 비지배해 풀의 운영은 의사결정자가 선호하는 대안을 효과적으로 탐색한다는 것을 알 수 있었다. 여러 실험문제를 사용하여 비교한 결과, 모든 문제에서 제안한 기법이 기존의 기법들보다 탐색효율이 우수한 결과를 가져왔다.

제안한 기법의 기본 절차는 혼합모델 조립라인의 다목적 투입순서 문제뿐만 아니라 다양한 형태의 다목적 조합최적화 문제의 대화형 의사결정에 적용될 수 있다.

참고문헌

Bard, J. F., Dar-El, E. M. and Shtub, A. (1992), An analytic framework for sequencing mixed model assembly lines, *Int. J. Prod. Res.*, 30, 35-48.
 Bard, J. F., Shtub, A. and Joshi, S. B. (1994), Sequencing mixed-model assembly lines to level parts usage and minimize line length, *Int. J. Prod. Res.*, 32, 2431-2454.

- Burns, L. D. and Daganzo, C. F. (1987), Assembly line job sequencing principles, *Int. J. Prod. Res.*, 25, 71-99.
- Dar-El, E. M. (1978), Mixed-model assembly line sequencing problems, *Omega*, 6, 313-323.
- Goldberg, D. E. (1989), *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, New York.
- Graves, S.B., Ringuest, J. L. and Bard J. F. (1992), Recent developments in screening methods for nondominated solutions in multiobjective optimization, *Computer & O.R.*, 7, 683-694.
- Greenwood, G. W., Hu, X. and D'Ambrosio, J. G. (1996), Fitness functions for multiple objective optimization problems: Combining preference with pareto rankings, *Foundations of Genetic Algorithms 4.*, 455.
- Grefenstette, J., Gopal, R., Rosmaita, B. and Gucht, D. V. (1985), Genetic algorithms for the traveling salesman problem. *Proc. 1st Int. Conf. Genetic Algorithms and Their Applications*, 160-168.
- Horn, J., Nafpliolas, N. and Goldberg, D. E. (1994), A niched pareto genetic algorithm for multiobjective optimization. *Proc. 1st IEEE Conference on Evolutionary Computation*, 82-87.
- Hyun, C. J., Kim, Y. H. and Kim, Y. K. (1998), A Genetic Algorithm for Multiple Objective Sequencing Problems in Mixed Model Assembly Lines, *Computer & O.R.*, Vol. 25, pp. 675-690, 1998.
- Kim Y.K., Hyun C.J., and Kim Y.H., Sequencing in mixed model assembly lines: A Genetic Algorithm Approach, *Computer & O.R.*, 23, 1131-1145.
- Malakooti, B. (1989), Identifying nondominated alternatives with partial information for multiple-objective discrete and linear programming problems, *IEEE Trans. Sys. Man and Cyber.*, 19, 95-107.
- Miltenburg, J. (1989), Level schedules for mixed-model assembly lines in just-in-time production systems, *Mgmt. Sci.*, 35, 192-207.
- Monden, Y. (1983), *Toyota Production System*, Institute of Industrial Engineers Press, Norcross, GA.
- Murata, T. and Ishibushi, H. (1995), MOGA: Multi-Objective Genetic Algorithms, *IEEE*, 289-294.
- Okamura, K. and Yamashina, H. (1979), A heuristic algorithm for the assembly line model-mix sequencing problem to minimize the risk of stopping the conveyor, *Int. J. Prod. Res.*, 17, 233-247.
- Schaffer, J. D. (1985), Multiple objective optimization with vector evaluated genetic algorithms, *Proc. 1st Int. Conf. On Genetic Algorithms and Their Applications*, 93-100.
- Shin, W. S. and Ravindran, A. (1991), Interactive multiple objective optimization: survey I -continuous case, *Computers: Ops Res*, 18, 97-114.
- Srinivas, N. and Deb, K. (1995), Multi objective Optimization Using Nondominated Sorting in Genetic Algorithms, *Evolutionary Computation*, 2(3), 221-248.
- Tsai, L. (1995), Mixed model sequencing to minimize utility work and the risk of conveyor stoppage, *Mgmt. Sci.*, 41, 485-495.
- Yano, C. A. and Rachamadugu, R. (1991), Sequencing to minimize work overload in assembly lines with product options, *Mgmt. Sci.*, 37, 572-586.