Journal of the Korean Institute
of Industrial Engineers
Vol.23, N0.3, September, 1997

혼합모델 조립라인에서 작업부하의 평활화를 위한 유전알고리듬*

A Genetic Algorithm for Improving the Workload Smoothness in Mixed Model Assembly Lines*

김여근** · 이수연** · 김용주**

Yeo Keun Kim** · Soo Yeon Lee** · Yong Ju Kim**

Abstract

When balancing mixed model assembly lines (MMALs), workload smoothness should be considered on the model-by-model basis as well as on the station-by-station basis. This is because although station-by-station assignments may provide the equality of workload to workers, it causes the utilization of assembly lines to be inefficient due to the model sequences. This paper presents a genetic algorithm to improve the workload smoothness on both the station-by-station and the model-by-model basis in balancing MMALs. Proposed is a function by which the two kinds of workloads smoothness can be evaluated according to the various preferences of line managers. To enhance the capability of searching good solutions, our genetic algorithm puts emphasis on the utilization of problem-specific information and heuristics in the design of representation scheme and genetic operators. Experimental results show that our algorithm can provide better solutions than existing heuristics. In particular, our algorithm is outstanding on the problems with a larger number of stations or a larger number of tasks.

* 본연구는 1996년도 한국과학재단 연구비 지원에 의한 결과임 (과제번호: 961-1012-083-1)

** 전남대학교 공과대학 산업공학과

1. 서론

라인밸런싱 문제는 특정 목적에 맞도록 작업장에 작업을 할당하는 문제로, 본 연구에서는 작업부하의 평활화를 목적으로 하는 혼합모델 조립라인 밸런싱(mixed model assembly line balancing: MMALB)문제를 다룬다. 혼합모델 조립라인은 라인의 효율적인 이용을 위하여 한 라인에서 유사한 여러 모델의 제품을 생산하는 조립라인이다. 혼합모델 조립라인은 모델별로 생산량과 총작업시간이 다르고 부분적으로 작업내용, 작업방법 및 작업시간, 자재 및 이용하는 설비 및 치공구가 상이하다. 따라서 MMALB에서는 단순히 사이클 타임이나 작업장수를 최소로하여 균형효율만을 최대로 하는 목적으로 작업편성을 할 경우, 라인의 효율적인 운영이 어려워 진다. MMALB에서는 균형효율을 최대로 하는 목적이외에 단위 사이클 생산기간동안 모델별로 각 작업장에 할당된 작업량이 가능한 균등하게 할당되어야 하고, 또한 단위 사이클 생산기간 동안 작업장에 할당된 총작업량이 가능한 평활하게 할당되어야 한다.

모델별로 작업장에 할당된 작업량의 차가 큰 경우, 작업량이 많은 모델이 라인에 연속적으로 투입되면 작업혼잡 등이 발생되어 컨베이어의 정지위험이 커지고, 조립에 필요한 작업장의 길이와 라인길이, 그리고 발주간격이 길어지게 된다. 한편 단위 사이클 생산기간 동안 각 작업장에 작업량 평활화가 이루어지지 않으면, 작업부하의 불공평성에 의해 작업능률의 저하를 초래할 수 있다. 본 연구에서는 각 모델을 기준으로 하여 작업장에 작업량을 균등하게 할당하는 '모델별 작업량 평활화'와 작업장을 기준으로 하여 각 작업장에 할당된 모든 모델의 작업량의 합을 균등하게 하는 '작업장별 작업량 평활화'를 고려한 혼합모델 조립라인 밸런싱 문제를 다룬다. 본 연구에서 이들 두 평활화에 대한 의사결정권자의 선호를 잘 반영할 수 있는 목적함수를 제시한다.

기존 라인밸런싱에 관한 연구들은 대부분 작업의 선후관계를 고려하여 균형효율만을 높이는 발견적(heuristic) 기법이나 정확한 해를 구하는 모형개발에 중점을 두어왔다. 라인에서 하나의 모델만을 생산하는 단일모델 조립라인(single model assembly line)에 있어서는 Talbot[20]는 기존의 효율적인 여러 발견적 기법을 비교평가하였으며, Baybars[3]는 라인밸런싱의 여러 수리모형을 비교분석하였다. 단일모델에서 작업부하를 균등하게 할 수 있는 규칙과 절차에 관해서는 Rachamadugu와 Talbot[18]에 의해 연구되었다. 혼합모델 조립라인에서의 밸런싱문제는 Arcus[2], Thomopoulos[21]에 의해 제기되었다. 그후 Thomopoulos[22]는 모델별 작업량 평활화 문제를 다루었으나 그 절차를 명확히 제시하지 않았으며, Macaskill[15]은 작업장별 작업평활화에 중점을 둔 라인밸런싱 문제를 다루었다. 그외의 연구로 Roberts와 Villa[19]는 혼합모델 조립라인의 라인밸런싱에서 작업자의 유휴시간을 최소화 하는 정수계획모형을 제시하였고, Chakravarty와 shtub[4]는 재공재고를 최소화 하는 라인밸런싱 기법을 제시하였다. 아직까지 혼합모델 조립라인에서 모델별 작업량과 작업장별 작업량의 균등한 할당을 동시에 고려한 연구는 이루어지지 않고 있다.

본 연구의 목적은 혼합모델 조립라인에서

모델별 작업량의 평활화와 작업장별 작업량의 평활화를 위한 효율적인 유전알고리듬(genetic algorithm: GA)을 제시하는 데 있다. 유전알고리듬은 생물의 진화과정을 모방하여 개발된 일종의 확률적 탐색기법으로, Holland [10]에 의해 소개되었다. GA는 조합최적화 문제에 효과적인 기법으로 알려져 있다[8, 16]. GA를 이용한 라인밸런싱에 관한 연구로 Anderson과 Ferris[1]가 사이클 타임을 최소화하는 문제를 다루었으나, 가능해를 생산하는 유전연산자를 제시하지 않았고 단지 비가능해에 대해 벌금함수(penalty function)를 사용하였다. Leu *et al.*[14]의 연구는 작업장 수를 최소화 하는 문제를 다루고 있으며, 작업장에 작업을 할당하는 순서로 개체를 표현하고, 가능해를 생산하는 유전연산자를 개발하였다. Kim *et al.*[12]은 단일모델 조립라인에서 여러 목적에 대하여 라인밸런싱 할 수 있는 유전알고리듬을 제시하였다. 이들 연구[1, 12, 14]는 모두 단일모델 조립라인을 대상으로 하고 있다. 조립라인 밸런싱에 관한 GA의 연구는 아직 미미한 실정이며, 특히 혼합모델 조립라인 밸런싱에 관한 GA의 연구는 이루어 지지 않고 있다.

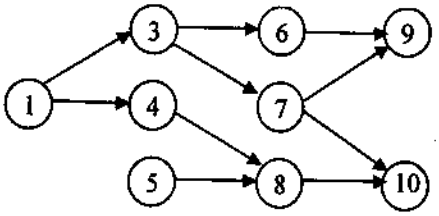
본 연구에서는 혼합모델 조립라인 밸런싱 문제에 GA를 적용하기 위하여 문제에 적합하고 해의 정보이용이 가능한 유전표현에 관하여 연구하고, 개체가 갖는 특정정보를 이용한 발견적 유전연산자를 개발한다. 또한 실험을 통하여 제시한 HGA의 성능을 비교 분석한다. 제시한 유전알고리듬을 발견적 유전알고리듬(heuristic genetic algorithm: HGA)이라 부르기로 한다.

본 연구의 구성은 다음과 같다. 제2장에서는 혼합모델 조립라인의 작업부하 평활화를 위한 목적함수를 제시하며, 제3장에서는 발견적 유전알고리듬을 제안한다. 그리고 제4장에서는 실험을 통하여 발견적 유전알고리듬의 성능을 비교분석한다. 제5장은 결론으로 구성되어 있다.

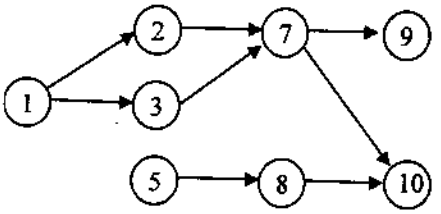
2. 혼합모델 조립라인의 작업부하 평활화

혼합모델 조립라인은 앞에서도 언급하였듯이 각 모델에 따라 총생산시간, 생산량, 각 작업시간과 작업내용, 사용하는 치공구 및 부품이 다를 수 있다. 따라서 혼합모델 조립라인을 각 모델에 대해 독립적으로 라인밸런싱하면, 여러 모델에 동일하거나 유사한 작업들이 다른 작업장(작업자)으로 분산되어 할당될 수 있다. 이 경우 작업능률의 저하와 부품저장공간 및 치공구나 설비에 대한 중복투자가 요구된다. 따라서 이러한 문제를 해결하기 위하여 각 모델의 선행공정도(precedence diagram)를 하나로 통합한 결합선행공정도(combined precedence diagram)를 사용한다. 결합선행공정도는 각 모델에서 수행되는 모든 작업에 대해 동일 작업이 중복되어 나타나지 않도록 하면서 하나의 선행공정도로 나타낸 것이다. 즉, 결합선행공정도상에서 각 노드는 모든 모델 또는 몇개의 모델에서 공통적으로 수행되는 작업을 나타내거나 또는 어느 특정 모델에서만 수행되는 작업을 나타내기도 한다. 모델간에 특정작업이 작업선행 관계가 상충될 수 있다. 예로 작업 A가 어떤 모델에서는 작업 B의 선행작업인데 다른 모델에서는 작업 B의 후행작업일 수 있다. 이 경우에는 각 모델에서 작업 A를 다른

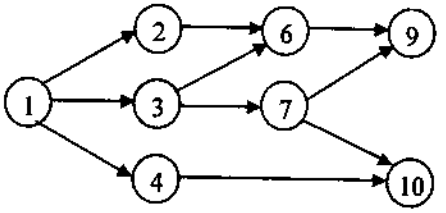
작업으로 취급하여 결합선행 공정도를 작성할 수 있을 것이다. MMALB에서는 이와같은 결합선행공정도를 사용하여 각 모델의 작업 선행 관계를 하나로 나타냄으로써, 모델마다 공통적으로 수행되는 작업들이 서로 다른 작업장에 할당되는 것을 방지할 수 있다. 예로 세 모델에 대한 선행공정도가 그림 1과 같다고 하자. 그림에서 원안에 있는 숫자는 작업번호를 나타낸다. 이를 결합선행공정도로 나타내면 그림 2와 같게 된다.



(a) 모델 1 의 선행공정도



(b) 모델 2 의 선행공정도



(c) 모델 3 의 선행공정도

그림 1. 각 모델의 선행공정도

한편, 혼합모델 조립라인에서는 계획기간동안 생산될 모델의 갯수에 대해 최소부품집합(minimum part set: MPS)을 반복 생산함으로써 계획기간동안의 수요를 만족시키는 사이클 생산방식을 흔히 채택하고 있다. 따라서 결합선행공정도 상에서 각 작업에 대한 작업시간은 모델별 작업시간과 MPS를 고려하여 설정한다.

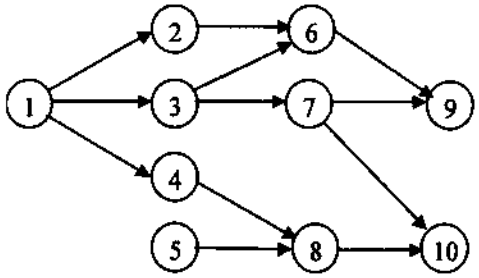


그림 2. 결합선행공정도

작업량의 평활화를 위하여 사용되는 기호는 아래와 같다.

i : 작업 번호, i=1,2,...,N

j : 작업장 번호, j=1,2,...,M

k : 모델 번호, k=1,2,...,L

t_{ik} : 모델 k에서 작업 i의 작업시간

$I(j)$: 작업장 j에 할당된 작업의 집합

d_k : MPS생산 동안 모델 k의 계획 생산량, 즉 MPS = $(d_1, d_2, \cdots, d_L)$

$\hat{t}_{ik} = d_k t_{ik}$ (MPS기간동안 수행되는 모델 k의 작업 i의 총 작업시간)

$\hat{t}_i = \sum_{k=1}^{L} \hat{t}_{ik}$ i=1, 2, ···, N (MPS생산동안 작업 i의 총 작업시간)

$\overline{T} = \frac{1}{M}\sum_{i=1}^{N} \hat{t}_i$ (MPS생산에 대한 작업장의 평균 작업량)

$\overline{T}_k = \frac{1}{M}\sum_{i=1}^{N} \hat{t}_{ik}$(모델 k에 대한 작업장의 평균 작업량)

$T_{jk} = \sum_{i \in I(j)} \hat{t}_{ik}$, j=1, 2, ···, M, k=1, 2, ···, L

(MPS생산동안 작업장 j에 할당된 모델 k의 총작업량)

$T_j = \sum_{i \in I(j)} t_i$, $j=1, 2, \cdots, M$ (MPS생산동안 작업장 j에 할당된 총작업량)

본 연구에서는 작업장별 작업량의 평활화와 모델별 작업량 평활화에 대한 척도는 흔히 사용되는 척도로써[18, 22] 아래와 같이 정의한다.

(1) 작업장별 작업량의 평활화

$$\text{Minimize } f_w = \frac{1}{M}\sum_{j=1}^{M} |T_j - \bar{T}| \quad (1)$$

(2) 모델별 작업량의 평활화

$$\text{Minimize } f_m = \frac{1}{M}\sum_{j=1}^{M}\sum_{k=1}^{L} |T_{jk} - \bar{T}_k| \quad (2)$$

이들 두 목적이 항상 상충관계를 갖지는 않는다. 그러나 하나의 목적에 중점을 두어 작업을 할당하면 다른 목적이 희생되는 경우가 흔히 발생한다. 예를 들어, 어떤 작업장에 할당된 작업이 특정 모델에서는 작업하지 않는 경우(모델의 선택사양 작업 등에서 발생), 작업장별 작업량 평활화를 위해서는 다른 모델의 작업량을 많이 할당해야 한다. 이 경우 모델별 작업량 평활화가 손상을 받게 된다.

본 연구에서는 작업장간 작업량의 편차를 일정범위내에서 허용하면서, 가능한 모델별 작업량을 평활화하고자 한다. 본 연구에서 사용하는 작업량 평활화의 평가함수 $eval$를 최소화 하고자 한다. 이는 아래와 같다.

$$\text{Minimize } eval = \sum_{j=1}^{M} \left\{ \sum_{k=1}^{L} |T_{jk} - \bar{T}_k| + \beta \cdot PT_j^{\gamma} \right\} \quad (3)$$

여기서 $PT_j = \begin{bmatrix} \max(T_j - T_H, 0), \text{ if } T_j > \bar{T} \\ \max(T_L - T_j, 0), \text{ if } T_j < \bar{T} \end{bmatrix}$

$T_L = (1-\alpha)\,\bar{T}$ (작업장에 할당되는 작업시간 허용 하한)

$T_H = (1+\alpha)\,\bar{T}$ (작업장에 할당되는 작업시간 허용 상한)

α, β, γ: 파라미터

여기서 PT_j는 작업장 j에 할당된 총 작업량이 허용 작업량의 상한보다 크거나 하한 보다 작을 때 갖는 일종의 벌금(penalty)으로, 파라미터 α, β, γ에 의해 결정된다. 여기서 α는 작업장에 할당되는 작업시간의 허용편차를 나타내는 파라미터로써 의사결정권자(라인 관리자)에 의해 결정될 수 있다. 그리고 β, γ는 벌금의 부과정도를 나타내는 파라미터이다. 평가함수의 첫째항은 모델별 작업량 편차를, 둘째항은 작업량 허용편차를 초과하는 작업장별 작업량 편차를 나타내고 있다.

평가함수에서 파라미터 α, β, γ에 의해 모델별 또는 작업장별 작업량 평활성에 대한 가중치를 조절할 수 있다. 예로 β, γ가 1보다 큰 경우, α값이 클 수록 모델별 작업량 편차를 강조하는 반면, α가 작을 수록 작업장별 작업량 편차의 최소화를 강조하게 된다. 특히 $\alpha=0$, $\gamma=1$이면 식 (1)과 식 (2)를 두 목적함수로 하고 두 목적간의 가중치는 β로써 결정하는 모형이 된다. 따라서 식 (3)을 목적함수로 갖는 모형은 작업장별 작업량 평활화와 모델별 작업량 평활화에 대한 의사결정자의 선호를 잘 반영할 수 있는 모형으로 볼 수 있다.

3. 발견적 유전알고리듬

유전알고리듬은 여러개의 개체로 이루어진 모집단을 운영하는 일종의 반복적인 탐색기법이다. 유전알고리듬은 먼저 주어진 문제의 여러 잠재해를 임의로 발생하여 초기 모집단을 생성한다. 생성된 모집단에서 평가함수에 의해 각 개체의 적응도를 평가하고, 이에 기초하여 모집단에 생존할 개체들을 선별한다. 선별된 개체로 이루어진 모집단에 교차와 돌연변이의 유전연산에 의해 개체를 재생산하여 다음세대의 모집단을 구성한다. 이 과정을 종료조건이 만족될 때까지 반복한다. 유전알고리듬은 선별과 유전연산에 의해 다양한 해공간의 탐색(exploration)과 좋은 해의 효율적 이용(exploitation)이라는 성질을 갖는다.

전통적인 유전알고리듬은 작업 선후행 제약이 존재하는 라인밸런싱과 같이 강한 제약이 존재하는 문제에 적용할 경우, 상당히 낮은 탐색성능을 보인다[5, 6]. 이것은 전통적인 유전알고리듬이 문제에 적합한 발견적 기법이나 문제가 갖는 특정 정보의 이용보다는 강한 임의성(randomness)을 갖는데 기인한다. 강한 제약을 갖는 여러 조합최적화 문제에 있어서 문제의 특성에 적합한 발견적 기법의 결합은 GA의 탐색능력을 향상시킨다[5].

본 연구에서는 MMALB문제에 GA를 적용하는데 있어서 유전연산자에 문제의 특성에 따른 정보를 이용한 발견적 기법을 결합한 새로운 발견적 유전연산자를 개발한다. 제안된 발견적 유전알고리듬의 기본구조는 다음과 같다. 여기서 $P(t)$는 세대 t에서의 모집단을, N_p 는 모집단 크기를 의미한다.

단계 1. 가능해를 임의로 생성하여 초기모집단 ($P(t=0)$)을 구성하고 평가한다. 그리고 $P(0)$에서 가장 좋은 개체를 보존한다.

단계 2. 모집단 $P(t)$에서 N_p개의 개체를 선별하여, 이를 $P(t+1)$로 둔다.

단계 3. 교차율에 따라 부모개체를 선택하여 발견적 기법과 결합한 교차연산을 행하여 자손을 생산하고, 생산된 자손과 부모개체를 교체한다.

단계 4. 돌연변이율에 따라 개체를 선택하여 발견적 기법과 결합한 돌연변이를 행한다.

단계 5. 모집단 $P(t+1)$을 평가한다.

단계 6. $P(t)$에서 가장 좋은 개체가 $P(t+1)$의 가장 좋은 개체의 적응도보다 높으면, 이를 $P(t+1)$에서 가장 낮은 적응도를 갖는 개체와 교체한다.

단계 7. 종료조건을 만족하면 끝내고, 그렇지 않으면 $t \leftarrow t+1$로 두고 단계 2로 간다.

제안한 유전 알고리듬의 요소들에 관하여 구체적으로 보자.

3.1 표현

GA성능의 향상을 위해서는 문제의 특성에 적합한 개체 표현이 요구된다. GA에서 개체는 자연스럽고 명확하게 잠재해를 나타내야 하고, 가능한 잠재해가 중복 표현되지 않아야 한다. 중복표현될 경우 해공간의 효율적 탐색이 이루어 질 수 없다. 또한 표현방법은 유전연산자와 밀접하게 관련되어 있다. 개체가 갖는 중요한 정보가 유전연산자에 의해

추출되고 자손에게 잘 전파될 수 있도록 개체를 표현하여야 한다.

MMALB문제는 일종의 분할문제이다. 분할문제는 어떤 목적에 의해 객체들을 그룹으로 나누는 문제이다. 분할문제에서는 흔히 객체들이 상호독립적이고 그룹 또한 상호간에 순서 등의 관련성이 없는 경우를 다룬다. 그러나 MMALB문제는 작업(객체)들을 작업장(그룹)에 배분하는데 있어서 작업의 선후행 관계가 존재하고, 이에 따라 작업장의 순서가 있게 된다. 따라서 MMALB문제는 객체를 그룹으로 배분하는데 제약을 갖는 분할문제의 특수한 형태로 볼 수 있다. 이제까지 분할 문제에 GA의 적용을 위해 순열표현, 그룹번호 표현 등 여러 표현방법들이 제시되었다[11].

본 연구에서는 그룹번호 표현을 사용한다. 그룹번호 표현은 인자값이 작업장번호를 나타내며, 인자의 위치는 작업번호를 나타낸다. 즉, 작업 i가 작업장 j에 할당되어 있으면, i번째 인자의 값은 j이다. 예로 그림 2에서 작업장 1, 2, 3에 각각 작업 {1, 2, 3}, {4, 6, 7}, {5, 8, 9, 10}을 할당하였다 하자. 이를 그룹번호로 표현하면 (1, 1, 1, 2, 3, 2, 2, 3, 3, 3)과 같이 된다.

MMALB문제에서 그룹번호 표현은 해석방법이 필요없으며 각 인자가 갖는 정보를 명확히 나타내 주고 있다는 장점이 있다. 이 표현은 가능해를 생산하는 유전연산자의 개발이 어렵다는 단점이 있으나, 이를 극복하는 한 방법으로 각 인자가 갖는 정보를 이용하여 가능해를 유도하는 발견적 기법을 사용할 수 있다.

3.2 평가함수와 선별

평가함수는 개체의 적응도를 평가하는 함수로 흔히 최적화 하려는 문제의 목적함수를 사용한다. 본 연구에서 다루는 MMALB의 목적은 일정한 작업장별 작업량 편차를 허용하면서 모델별 작업량의 편차를 최소화 하려는 것으로 식 (3)을 평가함수로 사용하였다. 식 (3)으로 부터 작업장 j에 대한 평활도의 척도는 다음과 같이 나타낼 수 있다.

$$WD_j = \sum_{k=1}^{L} |T_{jk} - \overline{T_k}| + \beta \cdot PT_j^r, \quad j=1, 2, \cdots, M \qquad (4)$$

선별계획(selection scheme)은 각 개체의 평가함수 값을 기초로 다음 세대의 생존개체를 선택하는 방법이다. 선별방법에는 비례적 재생산(proportionate reproduction), 토너먼트 선별(tournament selection), 순위선별(ranking selection) 등이 있다[8, 16]. 본 연구에서 다루고 있는 MMALB문제는 최소화 문제이다. 본 연구에서는 토너먼트 선별을 사용한다. 토너먼트 선별은 최소화 문제에서도 평가 값의 척도를 재구성할 필요가 없고 토너먼트 크기에 의해 모집단의 선별력을 조정할 수 있다는 장점을 갖는다. 이 절차는 다음과 같다. 먼저, 전체 모집단을 임의의 순열로 나열한다. 이 순열을 l(토너먼트 크기)개의 개체로 구성된 블럭으로 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.

유전알고리듬에서 선별된 후보개체들은 유전연산을 하게 된다. 유전연산을 한 후, 다음 세대를 최종적으로 구성하기 전에 본 연구에서는 우수개체 보존전략(elitism)을 사용하였다. 이 전략은 가장 좋은 개체가 유전 연산을 통해 손실되는 것을 방지하기 위하여 현재까지 생성된 개체중에서 가장 좋은 개체를 모집단에 보존시키는 전략이다.

3.3 발견적 구조 연산자

주어진 문제의 특성을 이용한 발견적 기법을 유전연산자와 결합하면 최적해에 대한 GA의 수렴성을 향상시킬 수 있다. 본 연구에서는 작업장별 정보를 추출하여 평가함수 값이 좋은 작업장을 가능하면 변형시키지 않고 자손에게 유전함으로써 다음 세대에 전파될 수 있도록 하는 발견적 기법을 유전연산자에 결합하였다.

본 연구에서는 von Laszewski[13]가 제시한 유전연산자인 구조교차(structural crossover: SX)와 구조돌연변이(structural mutation: SM)에 기초하여, MMALB문제의 발견적 정보를 효과적으로 이용할 수 있는 발견적 구조 교차(heuristic structural crossover: HSX)와 발견적 구조 돌연변이(heuristic structural mutation: HSM)를 개발하였다. 또한 이들 발견적 유전연산자가 MMALB 문제의 가능해를 생산하면서 좋은 해를 생산하는 재할당 방법을 개발하였다.

SX는 본래 그룹의 크기(포함되는 객체의 수)가 고정된 그래프 분할문제에 적용하기 위해 개발된 교차로서, 그룹의 정보를 잘 보전하여 자손에게 전달하는 특징을 가지고 있다.

본 연구에서 제안한 HSX는 기존 SX와 다음과 같은 차이점을 갖는다. 첫째, SX는 하나의 그룹을 선택하여 자손에 유전하는 반면, HSX에서는 여러개의 그룹(작업장)을 선택하여 자손에 유전한다. 둘째, SX는 임의로 그룹을 선택하는 반면, HSX는 문제의 특성에 따른 정보를 이용하여 그룹을 선택한다. 마지막으로, MMALB에 주어진 제약으로 인하여 SX와 HSX의 재할당 방법이 다르다.

HSX에 대한 설명의 편의를 위해 다음 기호를 정의한다.

$WD_j(p)$: 개체 p에서 작업장 j에 대한 평가값 (i.e., $WD_j = \sum_{k=1}^{L} |T_{jk} - \overline{T_k}| + \beta \cdot PT_j^r$)

$WB_j(p_1,p_2)$: 개체 p_1의 작업장 j에 대한 평가값에서 개체 p_2의 작업장 j에 대한 평가값을 뺀 값, (i.e., $WD_j(p_1) - WD_j(p_2)$, $j \in J$).

3.3.1 발견적 구조 교차

본 연구에서 제시한 발견적 구조 교차는 작업장별 평가함수값이 좋은 작업장의 정보를 자손에게 효율적으로 전달할 수 있도록 한다. 교차율에 의해 모집단에서 부모개체를 선택하고, 이를 임의로 짝지어 두 부모 p_1, p_2의 쌍을 만든다. 각 부모쌍 p_1, p_2의 교차방법은 다음과 같다.

단계 1. 부모 p_1 , p_2 중에서 임의로 한 부모 (p_1)를 선택하고, $WB_j(p_1, p_2)$, j= 1, …, M을 계산한다.

단계 2. $WB_j(p_1, p_2)$가 큰 순서로 Nq ($\langle M$) 개의 작업장을 선택하여 집합 Q를 만든다. 여기에서 Nq는 다른 부모 (p_2)로부터 자손에 유전될 작업장의 수를 나타내는 파라미터이다.

단계 3. Q에 속한 작업장의 번호를 갖는 p_2의 인자들을 모두 자손(o_1)의 같은 위치에 복사한다. 자손 o_1에서 결정되지 않은 인자는 p_1의 같은 위치에 있는 인자를 상속 받는다.

단계 4. p_1로부터 상속 받은 o_1의 인자 중 Q에 속한 작업장의 번호를 갖는 인자와 선후행 관계를 어긴 인자들을 지운다.

단계 5. 단계 4에서 지워진 인자들에 대해서 발견적 기법을 이용하여 작업의 선후행 제약을 만족할 수 있도록 재할당하여 자손 o_1을 생산한다. 여기서 사용된 재할당기법은 3.4절에서 구체적으로 다룬다.

p_1과 p_2의 역할을 바꾸어 자손 o_2를 생산한다.

단계 2는 두 부모로부터 상대적으로 좋은 평가 값을 가진 작업장을 추출하는 과정이다. 여기서 $WB_j(p_1, p_2)$는 작업장 j의 평가함수 값이 부모 p_1이 부모 p_2보다 얼마나 큰가를 나타낸다. 따라서 $WB_j(p_1, p_2)$, j=1, …, M가 큰 작업장을 부모 p_2로부터 선택하여 자손에게 유전하고자 한다. 단계 3은 단계 2에서 추출된 작업장들을 자손에게 상속하는 과정이다. p_2로부터 $WB_j(p_1, p_2)$, j=1, …, M가 큰 N_q개의 작업장을 자손에게 그대로 상속하고, 자손의 나머지 인자들은 p_1로부터 유전한다. 단계 4는 부모 p_2로부터 유전된 인자는 보존하고, 부모 p_1으로부터 유전된 인자들 중 제약조건을 어기는 인자들을 삭제하는 과정이다. 단계 5는 한 부모가 갖는 좋은 인자를 보존하고 다른 부모로 부터 상속된 인자중 제약을 어기는 인자를 보수하기 위하여 발견적 기법을 사용하여 재할당하는 단계이다. 단계 5의 재할당 기법의 절차는 3.4절에서 다룬다.

MMALB 문제에서는 작업장에 할당된 작업에 관한 정보가 중요하게 취급되어야 한다. 그룹번호 표현에서는 작업장을 구성하는 인자들이 개체에 분산되어 위치한다. 개발한 HSX는 분산되어 있는 이러한 인자를 가능한 파괴하지 않고 자손에게 유전할 수 있도록 함으로써 효율적으로 해를 탐색한다. HSX는 가능해를 생산하기 위하여 별도의 재할당 기법을 사용한다.

3.3.2 발견적 구조 돌연변이

GA에서 돌연변이는 해공간의 다양성을 유지하고 부분최적에 조기수렴하는 것을 방지하는 역할을 한다. 따라서 MMALB 문제에서 돌연변이 연산자는 해의 작업장 구조를 직접적이고 다양하게 변화시킬 수 있어야 한다.

제안한 HSM에서 돌연변이율은 개체단위가 아닌 개체의 유전자 단위로 적용된다. 즉, 본 연구에서 돌연변이율은 모집단에 있는 각 인자들이 돌연변이 될 확률을 의미한다. 돌연변이될 평균인자 수는 (모집단의 개체 수) ×(개체의 길이) ×(돌연변이율)이 된다. 본 연구에서 제안하는 HSM은 돌연변이율에 의해 선택된 인자를 지우고, 지워진 인자(작업)를 재할당한다. 이때 재할당 방법은 HSX에서와 같은 방법을 사용한다.

3.4 재할당방법

HSX와 HSM의 연산과정에서 지워진 인자의 재할당을 위한 발견적 기법을 제안한다.

이 기법에서는 할당가능한 작업장의 평활성 척도, 식 (4)를 이용한다.

먼저 사용되는 기호와 용어를 다음과 같이 정의한다.

R : 삭제된 인자에 해당되는 작업 집합.

NR_i : 작업 $i \in R$의 직선행 작업 중 삭제된 수.

$E(i)$: 작업 $i \in R$의 직선행 작업이 할당된 작업장 중 가장 늦은 작업장. 즉, 작업 i가 할당될 수 있는 가장 이른 작업장. 단, 직선행 작업이 없으면 1이다.

$L(i)$: 작업 $i \in R$의 후행 작업 중에서 삭제되지 않은(남아있는) 작업들이 할당된 작업장 중에서 가장 이른 작업장. 즉, 작업 $i \in R$가 할당될 수 있는 가장 늦은 작업장. 단, 후행 작업이 없거나 모두 삭제되었다면 마지막 작업장 번호인 N이다.

WD_j': 식 (4)에서 작업 $i \notin R$로 이루어진(삭제된 작업을 제외한) 작업장 j의 평활성

본 연구에서 개발한 재할당 방법의 절차는 다음과 같다.

단계 1. R과 NR_i, $i \in R$을 구하고, WD_j', j= 1, 2, …, M을 계산한다.

단계 2. NR_i = 0인 작업 중에서 t_i가 가장 큰 작업 i^*를 선택한다.

단계 3. i^*가 할당 가능한 작업장의 범위, 즉 $E(i^*)$와 $L(i^*)$를 구한다.

단계 4. 작업장 j = $E(i^*)$, $E(i^*)+1$, …, $L(i^*)$에 작업 i^*를 할당할 때 WD_j'가 최소가 되는 작업장 j^*를 선택한다.

단계 5. 작업장 j^*에 작업 i^*를 할당하고 i^*를 집합 R에서 삭제한다. $R=\phi$이면 종료하고, 그렇지 않으면 NR_i, $i \in R$와 WD_{j^*}'을 갱신하고 단계 2로 간다.

단, 단계2와 단계 4에서 같은 경우가 발생하면 임의로 선택한다.

이 재할당방법은 일종의 greedy heuristic으로 그룹분할 문제에서 사용된 Best-Fit-Decreasing(BFD) 규칙을 MMALB 문제에 적합하게 수정한 것이다. BFD규칙은 크기가 큰 객체부터 그룹에 할당하되, 개체를 할당할 때 목적에 가장 부합되는 그룹을 찾아 할당하는 방법이다. BFD규칙은 객체의 할당 순서에 제약이 없는 경우에 적용이 가능하다. 본 연구에서 제안한 재할당방법은 BFD규칙을 MMALB 문제에 적합하도록 객체의 할당 순서와 할당 가능한 그룹(작업장)의 범위를 제한한 것이다. 삭제된(재할당해야 할) 작업들 중 직선행 작업이 모두 작업장에 할당된 작업들의 집합 R을 구하고, R에 속한 작업 중 작업시간이 가장 큰 작업 i^*을 작업장에 할당한다. 작업 i^*가 작업의 선후행 관계를 어기지 않고 할당될 수 있는 작업장의 범위는 이미 할당된 작업과의 선후행 관계에 의해 $E(i^*)$부터 $L(i^*)$까지 제한된다. 이 범위의 작업장 중에서 작업 i^*를 할당한 후 작업장별 평가함수값이 가장 좋은 작업장에 할당한다.

4. 실험 및 분석

제안한 HGA의 성능을 분석하기 위하여 먼저, 실험을 통하여 각 실험문제에 적용될 적정한 유전파라미터 값을 구하기 위하여 관련 파라미터를 분석하였다. 그리고 MMALB 문제의 발견적 정보 및 발견적 기법을 이용한 경우와 그렇지 않는 경우의 탐색 성능을 비교하고, 제안한 HGA의 성능을 보이기 위하여 기존의 발견적 기법과 비교하였다. 제안된 HGA는 C++프로그램 언어로 구현되었으며, 66MHz Pentium CPU를 장착한 IBM-PC에서 수행되었다.

실험 결과는 GA을 30회 반복 수행하여 얻은 값을 평균하여 나타내었으며, 초기 모집단을 형성하는 해를 포함하여 재생산된 해가 20,000개가 되면 GA를 종료하였다.

4.1 적용문제

본 연구에서 실험문제는 이미 발표된 19개 작업을 갖는 Thomopoulos 문제[21], 111개 작업을 가진 Arcus 문제[2]를 대상으로 하였다. 이들 두 문제 이외에 기존 연구에서 혼합모델 조립라인의 실험문제를 찾기 어려워 저자가 ABC자동차와의 산학연구에서 얻은 작업이 61개인 ABC문제[23]를 실험문제로 사용하였다. 실험문제에서 Thomopoulos문제와 Arcus 문제는 작업시간에 100을 곱한 값을 사용하였으며, 이때 Arcus 문제는 그의 논문[2] Appendex X-3의 자료를 이용하였다. 또한 작업장 수가 많은 경우를 실험하기 위해서 Arcus문제에서 작업 95의 작업시간을 33491에서 6615로 변경하였다. 이들 세 문제를 모델 수와 작업장 수에 의해 표 1과 같이 문제를 정의하였다.

4.2 유전파라미터 값의 설정

HGA에서는 모집단의 크기 N_p, 토너먼트의 크기 l, 교차율 P_c, 돌연변이율 P_m 그리고 부모로부터 자손에게 유전될 작업장의 수 Nq가 결정되어야 할 파라미터이다. 적절한 유전파라미터의 값은 아직까지 해결하려는 문제의 특성에 따라 실험적으로 결정되어진다. 본 연구에서 토너먼트 크기는 흔히 사용되는

표 1. 실험 문제

문제	작업 수	모델 수	작업장 수	모델별 단위당 총 작업시간					
				모델 1	모델 2	모델 3	모델 4	모델 5	합
Thom 1	19	3	3	510	570	720	-	-	1800
ABC 1	64	4	6	1182	1005	1240	1454	-	4881
ABC 2	64	4	12	1182	1005	1240	1454	-	4881
Arcus 1	111	5	12	149149	150536	146638	146720	146476	739519
Arcus 2	111	5	15	149149	150536	146638	146720	146476	739519
Arcus 3	111	5	27	149149	150536	146638	146720	146476	739519

2로 두었으며, 다른 파라미터는 실험을 통하여 결정하였다. 예비실험에서 MPS는 유전파라미터에 크게 영향을 주지 않았다. 따라서 생산 사이클동안 각 모델을 하나씩 생산하는 것으로 두고 실험하였다.

첫째, 모집단의 적정크기를 결정하기 위해 모집단의 크기를 변화시켜 가면서 실험하였다. 본 연구에서 종료조건은 생산된 개체의 수를 기준으로 하기 때문에 모집단의 크기가 클 수록 적은 세대에 대해 적용한 결과가 된다. 반면에 작은 크기의 모집단의 운영은 탐색이 부분적인 해공간에 국한될 수 있다. 적용문제별로 예비실험의 결과 20에서 40까지의 범위에서 좋은 결과를 얻었다.

수행하였다. 작업 수가 적고 작업장 수가 적은 문제에서는 Nq의 변화가 탐색 성능에 크게 영향을 미치지 않았으나, 작업 수가 많고 작업장 수 또한 많은 경우는 Nq의 변화에 따라 탐색 성능이 민감한 반응을 보였다. 이 경우 Nq는 대략 작업장 수의 1/4부터 1/3까지의 수준에서 좋은 결과를 보였다. Nq가 적정 수준보다 크면 예상했던 바와 같이 해가 조기 수렴하였다.

예비 실험에서 좋은 결과를 얻은 파라미터 값의 범위는 표 2와 같았다. 이후의 비교 실험에서 각 문제에 대한 파라미터의 값은 괄호의 값을 사용하였다.

표 2. 유전파라미터 값

문제	교차율	돌연변이율	교차작업장 수	모집단 크기
Thom	0.6 - 0.9 (0.80)	0.2 - 0.3 (0.25)	1 - 2 (1)	20-30 (30)
ABC 1	0.6 - 0.9 (0.80)	0.1 - 0.2 (0.15)	1 - 3 (2)	30-40 (40)
ABC 2	0.6 - 0.9 (0.80)	0.1 - 0.2 (0.15)	2 - 4 (3)	30-40 (40)
Arcus 1	0.5 - 0.8 (0.60)	0.25 (0.25)	2 - 5 (3)	20-30 (30)
Arcus 2	0.5 - 0.8 (0.70)	0.1 - 0.2 (0.15)	3 - 7 (5)	30-40 (40)
Arcus 3	0.6 - 0.9 (0.70)	0.1 - 0.2 (0.15)	7 - 10 (9)	30-40 (40)

둘째, 교차율과 돌연변이율의 적절한 조합을 위해 0.05, 0.10, 0.15, ... , 0.95로 변화시켜 가면서 실험한 결과, 교차율은 0.50부터 0.80까지의 범위에서, 돌연변이율은 0.10부터 0.30까지의 범위에서 전반적으로 좋은 결과를 얻었다.

셋째, 적정한 교차 작업장 수(Nq)를 구하기 위하여, 실험 문제에 대해 Nq를 1부터 작업장 수의 절반 수준까지 적용하면서 GA를

4.3 발견적 유전연산자 성능분석

본 절에서는 발견적 정보의 이용이 GA의 탐색 성능에 미치는 영향을 분석하고자 한다. 본 연구에서 개발한 HGA는 문제의 특성에 관련된 정보와 발견적 절차를 이용한다. HGA의 효율성을 보이기 위해, HSX에서 부모로부터 자손에게 유전할 작업장(교차 작업장)들을 발견적 방법으로 선택하는 경우(Hx)와 임의로 선택하는 경우(Rx)를 비교하였다. 또한

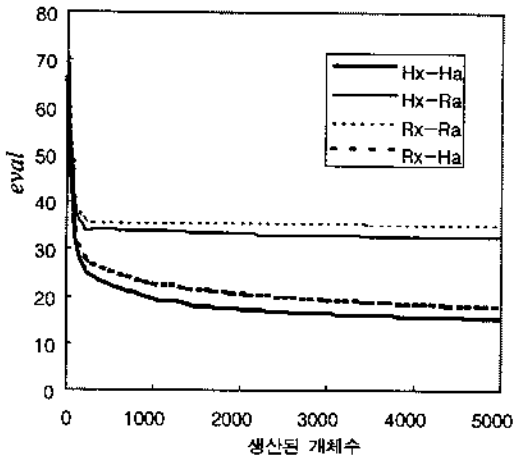
HSX와 HSM에서 삭제된 작업들을 제안한 방법으로 할당하는 경우(Ha)와 임의로 할당하는 경우(Ra)를 비교하였다.

각 실험 문제들에 대해 2가지 교차 작업장 선택 방법과 2가지 재할당 작업장 선택 방법을 조합한 4가지 경우의 GA을 수행하였다. 그림 3은 Arcus 문제와 ABC 문제에서 작업장이 각각 27개, 6개인 경우의 실험결과를 나타낸 것이다. 그림 3에서 X축은 생산된 개체의 수를 나타내고 Y축은 평가함수 값을 나타낸다. 이 결과에서 발견적 기법을 이용한 재할당방법이 HSX에서 교차 작업장을 효율적으로 선택하는 것보다 해의 질에 있어서 크게 영향을 미치고 있음을 볼 수 있다. 또한 예상했던 대로 발견적 정보를 이용한 교차 작업장의 선택이 임의로 선택한 방법보다 해의 탐색에 효과적임을 알 수 있다. MMALB에서와 같이 강한 제약이 존재할 경우, 문제의 특성에 따른 발견적 정보의 이용이 해의 효율적 탐색에 기여함을 알 수 있다.

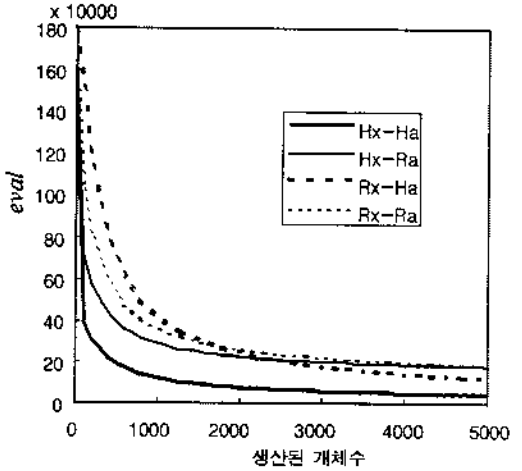
4.4 제안된 알고리듬의 성능 분석

제안된 알고리듬 즉, HGA의 성능분석을 위하여 기존의 발견적 기법과 비교하였다. 비교한 발견적 기법은 할당가능한 작업중에서 작업시간이 가장 큰 것을 할당하는 최대작업시간 규칙(MAXDUR)[17], 자신의 작업과 후행작업 시간의 합이 가장 큰 것을 우선적으로 할당하는 순위위치가중치 규칙(RPW)[15], 그리고 임의로 할당하는 임의 할당방법(RAND)이다. 또한 Hoffmann기법[9]과도 비교하였다. 이 기법은 첫 작업장부터 할당가능한 모든 작업의 조합을 구하여 가장 좋은 작업조합을 할당하고 다른 작업장에도 같은 방법으로 할당한다. 이 과정을 모든 작업이 할당될 때까지 반복하는 기법이다. 각 작업의 시간은 결합선행공정도 상에서 MPS 생산기간 동안의 각 노드의 총 작업시간, 즉 $\hat{t}_i$, i=1,2,…,N을 사용하였다.

Hoffmann기법[9]은 각 작업장에 할당가능한 모든 작업조합을 구하는데 많은 계산시간



a) 작업이 61개인 ABC 문제 (N = 6)



b) 작업이 111개인 Arcus의 문제 (N = 27)

그림 3. 발견적 정보의 이용이 탐색 성능에 미치는 효과

이 소요된다. 따라서 Gehelein과 Patterson[7]는 Hoffmann기법을 수정하여 계산 시간을 줄일 수 있을 뿐만 아니라 작업량의 평활성을 향상시킬 수 있는 작업할당 기법을 제안하였다. 이 기법에서는 한 작업장의 유휴시간이 다음과 같은 범위 내에 있으면 그 작업장을 대상으로 한 작업 할당 절차를 종료하고, 남아있는 작업장에 대해서 이와 같은 절차를 반복한다.

$$0 \le (CT\text{-}T_j) \le \theta \left[CT - \frac{T}{[T/CT]^+} \right], j=1, 2, \cdots, M$$

여기에서, CT는 사이클 타임(발주간격), T_j는 작업장 j에 할당된 작업시간, T는 총작업시간을 의미하며, $[x]^+$는 x보다 크거나 같은 가장 작은 정수이다. θ는 파라미터로서, 최소 작업장 수를 구하기 위해 다양하게 설정할 수 있다. 본 연구에서는 θ = 2.0을 사용하였다.

이들 발견적 기법은 사이클 타임이 주어지고 최소작업장 수를 구하는 Type-I ALB문제를 위하여 개발된 것이다. 본 연구는 작업장 수(N)가 주어진 MMALB문제를 다루고 있다. 따라서 이들 기법과 제안한 기법을 비교하기 위하여, CT를 이론적 사이클타임($[(1/M)\sum_{i=1}^{N} t_i]^+$)부터 평균작업시간의 1%씩 증가시키면서 구해진 작업장 수가 M과 같아질 때까지 반복 수행하여 해를 구하였다. 이들 네가지 발견적 기법들은 작업장에 할당된 총 작업시간에만 관련된다. 따라서 이들 기법에서는 모델별 작업량 평활화는 고려되지 않고 단지 작업장별 작업량의 평활화에 중점을 두고 있다.

표 3은 발견적 기법을 실험문제에 적용한 결과이다. 각 실험문제에 대해 2열과 같이 MPS를 다양하게 변화시켜 가면서 적용한 결과, 작업장별 작업량 평활화(f_w)와 모델별 작업량 평활화(f_m) 모두 HOFF-2.0이 다른 발견적 기법보다 대부분 좋은 해를 구하였다. HOFF-2.0이 다른 비교기법들이 단일경로(single pass) 규칙을 사용하는데 반하여, 각 작업장의 작업할당에 있어서 여러 작업조합을 탐색하고, 작업부하 평활성도 고려하고 있기 때문인 것으로 보인다.

표 4는 HGA에 대해서 α의 변화에 따른 식 (1)의 작업장별 작업량 편차(f_w)와 식 (2)의 모델별 작업량 편차(f_m)를 나타낸 것이다. 여기서 β= 1, γ= 2로 두었다. 표 4의 3열과 4열은 각각 발견적 기법에서 구한 가장 좋은 해의 f_w와 f_m을 나타내고, 4열부터 마지막 열까지는 HGA에서 α값의 변화에 따른 결과를 나타낸 것이다. 이때 HGA의 결과값은 HGA를 30회 수행하여, 각 실험에서 가장 좋은 해의 f_w와 f_m을 평균한 값이다. 그리고 '개선율'은 f_w와 f_m에 대해 각각 개선율 $= \frac{H_{best} - HGA}{H_{best}} \times 100(\%)$로 계산된 것이다. 여기서 H_{best}는 발견적 기법들 중에서 가장 좋은 해이다. 그리고 구한 해중에서 f_w와 f_m에 대해 각각 가장 좋은 해를 굵은 글씨로 표현하였다.

작업장별 작업량 평활화(f_w)는 α값에 따라 차이를 보이지만 전반적으로 제안한 HGA가 비교기법들보다 더 좋은 해를 생산하는 것을 알 수 있다. 특히 α=0일 때 ABC 1문제에서 MPS가 (1, 1, 1, 1)인 경우를 제외한 모든 문제에서 HGA가 발견적 기법보다 더 좋은 해를 생산하였다.

모델별 작업량 평활화(f_m)에 있어서는 단지 Arcus 1문제에서 α=0, MPS= (1, 1, 1, 1, 1), (1, 2, 4, 5, 8)인 경우를 제외하고는 모든 문

표 3. 기존 발견적 기법의 실험 결과

Problem	MPS	RAND		RPW		MAXDUR		HOFF-2.0	
		f_w	f_m	f_w	f_m	f_w	f_m	f_w	f_m
Thom	1 1 1	14.67	196.00	166.67	193.33	40.00	140.00	0.00*	106.67*
	3 2 1	105.00	204.00	166.67	193.33	40.00	140.00	0.00*	106.67*
ABC 1	1 1 1 1	24.77	80.59	32.63	85.88	26.13	72.50	2.38*	65.16*
	1 3 4 5	82.66	250.45	134.38	237.31	83.88	223.97*	7.03*	229.63
	6 4 2 1	75.82	293.91	113.56	289.13	69.00	238.13	16.31*	200.00*
ABC 2	1 1 1 1	29.88	58.85	24.73	60.41	29.03	54.06	5.95*	53.21*
	1 3 4 5	102.25	177.28	81.16	151.97	104.72	163.04	20.81*	139.14*
	6 4 2 1	93.86	180.59	77.57	175.85	60.50	157.59	52.02*	142.55*
Arcus 1	1 1 1 1 1	9781.03	7214.22	9736.22	9759.26	5315.39	5393.87	1762.58*	1247.14*
	5 3 2 1 1	39404.72	38586.43	37381.50	37509.67	20943.67	21014.00	3396.83*	5479.89*
	1 2 4 5 8	48691.43	42876.83	48531.72	48592.21	26065.39	26242.90	6735.93*	7693.93*
	1 4 8 3 1	26724.71	30473.51	33573.00	33652.21	18869.96	18988.29	4237.33*	5160.72*
Arcus 2	1 1 1 1 1	6832.16	6867.06	8317.08	8534.68	6054.61	6259.05	995.68*	1402.85*
	5 3 2 1 1	25188.77	25380.93	32177.73	33501.23	23430.40	24452.58	2843.33*	4775.92*
	1 2 4 5 8	25500.90	25720.85	20241.88	20685.88	29188.31	29207.41	5503.74*	5985.22*
	1 4 8 3 1	17127.75	17347.39	14058.05	14650.05	22084.56	22084.56	3527.65*	4500.28*
Arcus 3	1 1 1 1 1	4621.98	4704.49	4499.25	4503.22	4843.15	4850.85	4146.09*	4162.48*
	5 3 2 1 1	20068.91	20561.59	18479.84	18488.18	17486.64	17503.08	15994.90*	16093.94*
	1 2 4 5 8	23520.43	23674.18	21026.42	21054.54	25507.64	25561.77	18816.63*	18842.63*
	1 4 8 3 1	16831.21	16988.30	15167.78	15181.64	15396.82	15424.32	13559.46*	13596.73*

*: 실험에서 f_w와 f_m이 각각 가장 좋은 해

제에서 월등히 좋았다. α값이 클 수록 작업장에 허용 작업시간이 많아지기 때문에 모델별 작업량 평활화는 좋아지는 경향을 갖는다.

결국 제안한 HGA는 기존의 발견적 기법보다 작업장별과 모델별의 두 평활화를 동시에 개선시킬 수 있음을 알 수 있다. 다만 ABC 1문제에서 MPS= (1, 1, 1, 1)인 경우가 예외인데, 이 경우에도 작업장별 작업량 평활화는 약간 희생하면서(f_w의 값이 2.4에서 2.9로) 모델별 작업량 평활화는 크게 개선(f_m의 값이 65.2에서 43.6으로)시킬 수 있다. HGA에서 α의 변화에 따른 f_w와 f_m값의 비교로부터 의사결정자는 두 목적간의 선호도에 따라 작업할당의 적절한 대안을 선택할 수 있음을 보여주고 있다. 특히 제안한 알고리듬은 작업수와 작업장수가 많은 경우(Arcus 2문제와

표 4. 작업부하 평활화

문제	MPS	발견적 기법의 최선해		제안된 HGA											
				α=0.00			α=0.025			α=0.05			α=0.1		
		f_w	f_m	f_w	f_m	개선율	f_w	f_m	개선율	f_w	f_m	개선율	f_w	f_m	개선율
Thom	1 1 1	0.0*	106.7	0.0*	33.3	(0.0, 68.8)	6.7	20.0	(, 81.3)	13.3	13.3	(, 87.5)	13.3	13.3*	(, 87.5)
	3 2 1	0.0*	106.7	0.0*	90.7	(0.0, 15.0)	15.3	35.7	(, 66.6)	30.7	33.3	(,68.8)	31.7	33.3*	(, 68.8)
ABC 1	1 1 1 1	2.4*	65.2	2.9	43.6	(-20.8, 33.1)	8.7	27.6	(-263.8, 57.6)	11.3	24.3	(-375.1, 62.7)	11.1	20.4*	(-364.4, 68.7)
	1 3 4 5	7.0	224.0	6.5*	159.6	(7.9, 28.7)	25.5	83.2	(-262.2, 62.9)	32.1	74.7	(-357.0, 66.6)	38.8	62.7*	(-451.5, 72.0)
	6 4 2 1	16.3	200.0	8.7*	151.7	(46.8, 24.1)	24.9	87.4	(-52.9, 56.3)	35.5	70.7	(-117.5, 64.7)	30.6	58.6*	(-87.8, 70.7)
ABC 2	1 1 1 1	6.0	53.2	4.2	38.7	(30.2, 27.3)	4.9	33.9	(18.5, 36.2)	7.8	30.6	(-31.1, 42.4)	11.4	26.4*	(-91.3, 50.5)
	1 3 4 5	20.8	139.1	13.4*	136.9	(35.3, 1.6)	18.6	122.3	(10.7, 12.1)	25.1	101.7	(-20.5, 26.9)	39.7	93.1*	(-91.0, 33.1)
	6 4 2 1	52.0	142.6	12.6*	125.1	(75.8, 12.3)	16.2	101.4	(68.8, 28.9)	22.1	87.2	(57.5, 38.8)	29.6	70.5*	(43.1, 50.6)
Arcus 1	1 1 1 1 1	1762.6	1247.1	730.4*	1277.5	(58.6, -2.4)	1084.5	1206.2	(38.5, 3.3)	1131.9	1127.0	(35.8, 9.7)	969.2	1029.7*	(45.0, 17.4)
	5 3 2 1 1	3396.8	5479.9	2831.0*	5390.5	(16.7, 1.6)	4105.7	5297.8	(-20.9, 3.4)	3936.1	5277.0	(-15.9, 3.7)	3903.1	5104.1*	(-14.9, 6.8)
	1 2 4 5 8	6735.9	5963.3	3983.3*	5972.5	(40.9, -0.2)	6106.7	5225.6	(9.3, 12.4)	4785.2	5005.0	(29.0, 16.3)	6359.7	4359.6*	(5.6, 26.9)
	1 4 8 3 1	4237.3	5160.7	3272.2*	5059.6	(22.8, 1.4)	3770.7	4802.0	(11.0, 7.0)	3983.2	4272.3*	(6.0, 17.2)	3470.9	4468.7	(18.1, 13.4)
Arcus 2	1 1 1 1 1	995.7	1402.9	243.2*	888.7	75.6, 36.7)	420.7	732.3	(57.7, 47.8)	440.6	729.0*	(55.8, 48.0)	438.6	741.2	(56.0, 47.2)
	5 3 2 1 1	2843.3	4775.9	1071.7*	4164.3	(62.3, 12.8)	1745.1	3671.2*	(38.6, 23.1)	2528.4	4218.0	(11.1, 11.7)	2158.5	3684.0	(24.1, 22.9)
	1 2 4 5 8	5503.7	5985.2	927.7*	2607.6	(83.1, 56.4)	1756.8	1756.8*	(68.1, 70.7)	2007.1	3100.2	(63.5, 48.2)	2273.7	3266.0	(58.7, 45.4)
	1 4 8 3 1	3527.7	4500.3	768.9*	2707.3	78.2, 39.8)	1298.0	2377.1	(63.2, 47.2)	1419.9	2327.0	(59.8, 48.3)	1461.8	2316.1*	(58.6, 48.5)
Arcus 3	1 1 1 1 1	4146.1	4162.5	858.6*	1140.2	(79.3, 72.6)	992.1	1233.0	(76.8, 70.4)	996.5	1221.5	(76.0, 70.7)	753.9	980.3*	(81.8, 76.5)
	5 3 2 1 1	15994.9	16093.9	2989.6*	4482.9	(81.3, 72.2)	3023.1	4417.4	(81.1, 72.6)	3174.5	4374.6	(80.2, 72.8)	3258.5	4270.8*	(79.6, 73.4)
	1 2 4 5 8	18816.6	18842.6	4858.6	5599.2	(74.2, 70.3)	4412.5*	5172.9	(76.6, 72.6)	4323.9	5002.5*	(77.0, 73.5)	4459.7	5105.1	(76.3, 72.9)
	1 4 8 3 1	13559.5	13596.7	3100.8*	3869.2	(77.1, 71.5)	3133.0	3886.1	(76.9, 71.4)	2924.6	3634.5	(78.4, 73.3)	2743.1	3400.3*	(79.8, 75.0)

Arcus 3문제)에 있어서 기존의 발견적 기법보다 성능이 매우 우수함을 알 수 있다.

5. 결론

본 연구에서는 혼합모델 조립라인에서 작업장별 작업량의 평활화와 모델별로 작업장간 작업량의 평활화를 목적으로하는 라인밸런싱 문제를 해결하기 위하여, 새로운 유전알고리듬을 개발하였다. 이를 위해 먼저 작업장간 총 작업량의 평활성과 작업장간 모델별 작업량의 평활성에 대해 의사결정자의 선호도를 잘 반영할 수 있는 유연한 목적함수를 제안하였다. 그리고 효율적인 유전알고리듬의 개발을 위하여 문제의 해가 가진 정보를 자연스럽게 나타낼 수 있는 표현방법과 해의 탐색능력을 높이기 위하여 발견적 정보를 유전 연산자와 결합시킨 발견적 유전연산자를 개발하였다. 제안한 유전연산자에서 가능해를 만들면서 좋은 해의 생산을 유도하는 재할당방법을 개발하여 사용하였다.

실험 결과, 작업할당에 관한 발견적 정보의 이용은 유전알고리듬의 탐색 성능을 향상시킴을 알 수 있었다. 여러 실험문제를 사용하여 알고리듬의 성능을 비교 분석한 결과, 제안한 알고리듬이 기존의 기법들보다 우수하였다. 특히, 제안한 유전알고리듬은 작업수와 작업장의 수가 많은 문제에 있어서 기존 발견적 기법보다 성능이 월등히 우수함을 보였다.

참고문헌

[1] Anderson, E. J. and Ferris, M. C., "Genetic algorithms for combinatorial optimization: the assembly line balancing problem", *ORSA J. Computing*, vol. 6, pp.161-173, 1994.

[2] Arcus , A. L., "An Analysis of a Computer Method of Sequencing Assembly Line Operations.", *Ph.D. dissertation*, University of California, Berkeley, 1963.

[3] Baybars, I., "A survey of exact algorithms for the simple assembly line balancing problem.", *Management Science*, vol. 32, pp.909-932, 1986.

[4] Chakravarty, A. K. and Shtub, A., "Balancing mixed model lines with in-process inventory", *Management Science*, vol. 31, pp.1161-1174, 1985.

[5] Eiben, A. E., Raué, P. E and Ruttkay, Zs., "How to apply genetic algorithms to constrained problems", *Practical Handbook of Genetic Algorithms* 1: *Applications* (edited by L. Chambers), pp. 307-365. CRC Press, Boca Raton, 1995.

[6] Falkenauer, E. A., "New representation and operators for genetic algorithms applied to grouping problems", *Evolutionary Computation*, vol. 2, pp.123-144, 1994.

[7] Gehrlein , W. V. and Patterson, J. H., "Balancing single model assembly lines: comments on a paper by E. M. Dar-El (Manoor)", *AIIE Trans.*, vol. 10, pp. 109-112, 1978.

[8] Goldberg, D. E., "*Genetic Algorithms in Search, Optimization & Machine Learning*", Addison-Wesley, Reading, 1989.

[9] Hoffmann, T. R., "Assembly line balancing with a precedence matrix", *Management Science*, vol. 9, pp. 551-562, 1963.

[10] Holland, J. H., "*Adaptation in Natural and Artificial Systems*", Univ. of Michigan Press, Ann Arbor ., 1975.

[11] Jones, D. R. and Beltramo, M. A., "Solving partitioning problems with genetic algorithms", *Proc. 4th Int. Conf. Genetic Algorithms* (Edited by R. Belew and L. Booker), pp. 442-449. Morgan Kaufmann, 1991.

[12] Kim, Y. K., Kim, Y. J. and Kim, Y., "Genetic algorithms for assembly line balancing with various objectives", *Computers and Ind. Engng.*, vol. 30, pp. 397-409, 1996.

[13] von Laszewski, G., "Intelligent structural operators for *k*-way group partitioning problem", *Proc. 4th Int. Conf. Genetic Algorithms* (Edited by R. Belew and L. Booker), pp. 45-52. Morgan Kaufmann, 1991.

[14] Leu, Y. Y., Matheson, L. A. and Rees, L. P., "Assembly line balancing using genetic algorithms with heuristic-generated initial populations and multiple evaluation criteria", *Decision Science*, vol. 25, pp. 581-606, 1994.

[15] Macaskill, J.L., "Production line balance for mixed model lines", *Management Science*, vol. 19, pp. 423-434, 1972.

[16] Michalewicz, Z., *Genetic Algorithms + Data Structures = Evolution Programs (2nd edn)*. Springer-Verlag, Berlin, 1994.

[17] Moodie, C. L. and Young, H. H., "A heuristic method of assembly line balancing for assumptions of constant or variable work element times", *J. Industrial Engineering*, vol. 16, pp. 23-29, 1965.

[18] Rachamadugu, R. and Talbot, B., "Improving the quality of workload assignments in assembly lines", *Int. J. Prod. Res.*,vol. 29, pp. 619-633, 1991.

[19] Roberts, S. D. and Villa, C. D., "On a Multiproduct Assembly Line Balancing Problem", *AIIE Transactions*, vol. 2, pp. 361-364, 1970.

[20] Talbot, F. B., Patterson, J. H., and Gehrlein, W. V., "A comparative evaluation of heuristic line balancing techniques" *Management Science*, vol. 32, pp. 430-454, 1986.

[21] Thomopoulos, N. T., "Line balancing-sequencing for mixed-model assembly", *Management Science*, vol. 14, pp. 59-75, 1967.

[22] Thomopoulos, N. T., "Mixed model line balancing with smoothed station assignments", *Management Science*, vol. 16, pp. 593-603, 1970

[23] 김여근 외 5인, 대형트럭라인의 생산성 향상에 관한 연구, 아시아 자동차 공업 (주) 보고서, 1990.

97년 4월 최초 접수, 97년 7월 최종 수정