

대칭형 다계층 공진화 알고리즘을 이용한 FMS 공정계획 FMS process planning using a symmetric multi-layered coevolutionary algorithm

김재윤¹, 김여근²

¹ 전남대학교 고품질전기전자부품 및 시스템연구센터

² 전남대학교 산업공학과

Abstract

This paper proposes a new method of solving integrated problems that consist of several sub-problems in a symmetric multi-layered structure, and also demonstrate the applicability of the method. The proposed method is named *Symmetric Multi-layered Coevolutionary Algorithm* (SMCA). The SMCA imitates partly the natural process of endosymbiotic evolution, which is a special type of coevolution. The SMCA is applied to the process planning problem in flexible manufacturing system (FMS), taking account of the flexibility of machine, tool, process, and sequence. To do this, SMCA's components are studied and its strategies are developed to improve the performance. The proposed algorithm is compared with the existing ones in terms of solution quality. The experimental results confirm the effectiveness of our approach.

1. 서론

최근 여러 학문분야에서는 단위문제들을 큰 규모의 문제에 대한 부분문제로 보고, 이들 부분문제가 결합되어 통합된 하나의 큰 문제를 해결하기 위한 해법 개발에 많은 관심을 갖고 있다. 통합문제를 해결하는 기존의 방법으로는 다양한 문제상황을 제약으로 간주하여 문제를 단순화시키는 방법, 부분문제들의 관련성을 무시하고 부분문제들을 각각 독립적으로 해결한 후 이들 해를 결합하여 적용하는 방법, 그리고 한 문제를 먼저 해결하고 이 결과를 제약으로 다른 문제를 순차적으로 해결하는 방법 등이 있다. 그러나 이러한 방법들로부터 구한 해들은 실제로 적용 불가능한 해일 가능성이 높고, 비효율적이거나 해공간을 효과적으로 탐색하는데 한계를 갖는다. 따라서, 통합문제의 효과적인 해결을 위한 기법의 개발이 요구되고 있다.

통합문제는 구조적인 측면에서 살펴볼 때, 다계층 구조(multi-layered structure)를 갖는 경우가 많다. 즉, 통합문제는 부분문제가 복수개 결합되어 있고, 상위계층의 문제들은 하위계층의 여러 부분문제들이 서로 결합되어 구성된다. 또한, 다계층 구조의 문제는 부분문제들의 깊이가 동일한 문제와 서로 다른 문제로 구분할 수 있다. 전자를 대칭형 다계층 통합문제(symmetric multi-layered integrated problem), 후자를 비대칭형 다계층 통합문제(asymmetric multi-layered integrated problem)로 정의하기로 한다. 다계층 통합문제의 문제구조는 이를 위한 해결기법의 개발 과정에서 서로 다른 요소들을 고려하게 할 것이다. 본 연구에서는 공진화 알고리즘(coevolutionary

algorithm)을 기반으로 대칭형 다계층 통합문제를 효과적으로 해결할 수 있는 알고리즘을 제안하고, 그 적용성을 보이고자 한다. 제안한 알고리즘을 대칭형 다계층 공진화 알고리즘(SMCA: Symmetric Multi-layered Coevolutionary Algorithm)이라고 부른다.

공진화 알고리즘은 진화알고리즘의 특별한 형태로, 자연계에서 물 또는 그 이상의 종(species)들이 상호작용하고 상호적응해 나간다는 공진화 현상을 모방한 확률적 탐색기법이다(Koza, 1992). 공진화 알고리즘은 통합문제에서 적용의 우수성을 갖는 것으로 알려져 있다(Potter, 1997). 그러나 통합문제에 적용되었던 기존 알고리즘들(Maher and Poon, 1996; Potter, 1997; Kim *et al.*, 2000b; Kim *et al.*, 2003b)은 동일한 하나의 계층에 있는 부분문제들의 통합에만 적용될 수 있었다. 다계층 문제를 단일계층 문제로 변환하여 기존의 알고리즘을 해결의 방법론으로 채택할 수도 있으나, 이러한 경우에 부분문제의 수가 많아지면, 한 계층에 모든 모집단을 두고 동시에 운영해야 하므로 좋은 해에 관한 정보를 추출하고, 이를 유지 또는 진화시키는데 비효율적일 수 있다. 이와 관련하여, 본 연구의 저자들은 FMS (flexible manufacturing system)에서 부하할당(load) 문제를 위한 다계층 알고리즘을 개발하였다(Kim *et al.*, 2003a). 본 연구는 이전 연구의 확장으로 문제 구조가 대칭형인 다계층 통합문제를 위한 일반화된 알고리즘을 제안하고, 제안한 알고리즘의 적용성을 보인다. 적용성 검증을 위한 문제는 FMS 공정계획(process planning)문제이다. 이를 위하여, 다루는 문제를 효과적으로 해결할 수 있는 여러 유전 요소들을 개발한다. SMCA의 성능은 컴퓨터 실험을 통하여 기존 진화알고리즘들에 의해 구한 해들과 해의 질 측면에서 비교하여 분석한다.

2. 대칭형 다계층 공진화 알고리즘

2.1 알고리즘의 개념

SMCA의 개념적 설명에 앞서, 기호들을 정의하자. SMCA는 <그림 1>에서 보는 바와 같이 대칭형 다계층 구조의 형태를 보이는 문제에서 운영되고, 각 계층에 존재하는 모집단들은 트러스 형태의 2차원 정방형 격자구조를 갖는다고 가정한다. $Pop[i, g]$, $i = 1, 2, \dots, L$, $g = 1, 2, \dots, Q_i$ 은 계층 i 의 g 번째 모집단, $NP_{ij}[i, g]$ 는 $Pop[i, g]$ 에서 위치 (i, j) 와 그 주위 8개의 개체로 구성된 이웃으로 정의한다.

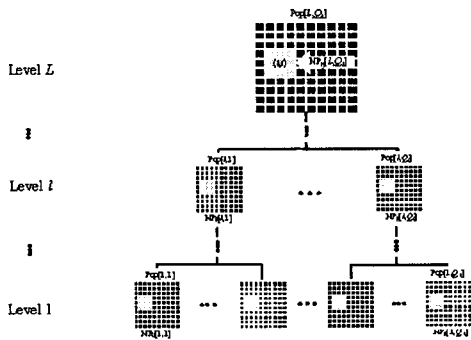


그림 1. 대칭형 다계층 구조의 예

SMCA의 기본 개념은 다음과 같다. 최하위 계층은 통합문제를 최소단위로 나눈 부분문제들로 구성된다. 계층의 수준이 올라갈수록 하위계층의 문제들이 결합되어 좀더 복잡한 구조를 갖는 문제들이 상위계층에 존재하고, 최상위 계층에는 완전한 통합문제가 위치한다. 각 문제들은 자신이 속해있는 계층의 다른 문제들과 상호작용하고, 계층간에는 하위계층의 정보가 상위계층으로 전달되는 상향식 정보교환을 통해 해를 탐색해 나간다. 계층내 상호작용은 동등한 진화단계의 생물들이 분리된 형태로 공생(symbiosis)해 나가는 과정을 모방하였다. 계층간 정보교환은 진화수준이 하위인 생물들이 하나의 생물처럼 서로 결합하고 이들이 상위수준의 생물로 진화해 나가는 내공생(endosymbiosis) 과정을 모방한 것이다. 이는 진화연산에서 공생진화가 해의 다양성을 강화시키고, 내공생진화는 탐색의 집중력을 높여줄 수 있다는 이들 두 효과(Kim et al., 2000b; Kim et al., 2001)를 적절히 결합하고자 함이다.

2.2 알고리즘의 절차

SMCA의 절차는 <그림 2>에서 보인 바와 같다. 각 단계에는 SMCA의 특징적인 전략들이 포함되어 있다. 단계별 의미와 실제 적용방법들은 <그림 2>의 아래에 구체적으로 설명되어 있다. 아래 절차에서 $f(\alpha)$ 는 개체 α 의 적응도를 의미하고, $\text{ImH}(NP_{ij}[l, q])$ 는 $NP_{ij}[l, q]$ 의 직상위 모집단에 있는 동일한 위치 (i, j) 의 이웃으로 정의한다.

단계1: 초기화 및 초기 적응도 평가

각 계층의 모든 모집단에 개체들을 임의로 생성하고, 초기 적응도를 평가한다. 가장 높은 적응도를 f_{best} 로 둔다.

단계2: 이웃 설정

임의의 위치 (i, j) 를 선택하고, $NP_{ij}[l, q]$, $l = 1, 2, \dots, L$, $q = 1, 2, \dots, Q_l$ 을 설정한다.

단계3: 적응도평가 및 계층간 상호작용

단계3.1: $l=1, q=1$ 로 설정한다.

단계3.2: $l < L$ 이면, $NP_{ij}[l, q]$ 개체들의 적응도를 평가한다. 공생파트너는 모든 $q' (\neq q)$ 인 $NP_{ij}[l, q']$ 에서 선택한다. 여기서, 가장 높은 적응도를 갖는 개체결합을 PE_{lq} 로 둔다.

$f(PE_{lq})$ 가 f_{best} 보다 높으면 f_{best} 를 갱신하고, 단계 3.3으로 간다. 그렇지 않으면 ($l = L$), 공생파트너의 선택없이 $NP_{ij}[l, q]$ 개체들의 적응도를 평가하고, 단계 4로 간다.

단계3.3: $\text{ImH}(NP_{ij}[l, q])$ 의 가장 낮은 적응도를 갖는 개체를 E_{old} 라 하자. 만약, $f(PE_{lq}) > f(E_{old})$ 이면, PE_{lq} 의 염색체에서 $\text{ImH}(NP_{ij}[l, q])$ 에 대응하는 부분을 복사하여 E_{old} 와 대체한다.

단계3.4: $q \leftarrow q+1$ 로 둔다. 만약, $q \leq Q_l$ 이면 단계 3.2로 간다.

단계3.5: $l \leftarrow l+1$ 로 둔다. 만약, $l \leq L$ 이면 $q=1$ 로 하고, 단계 3.2로 간다.

단계4: 이웃 진화

단계4.1: $l=1, q=1$ 로 설정한다.

단계4.2: $NP_{ij}[l, q]$ 에서 적응도를 기준으로 두 부모개체를 선택하여 교차한 후, 두 자손 개체를 생산한다.

단계4.3: $NP_{ij}[l, q]$ 에서 낮은 적응도를 갖는 두 개체를 선택하여 자손개체와 대체한다.

단계4.4: 돌연변이율에 따라 $NP_{ij}[l, q]$ 개체들을 돌연변이시킨다.

단계4.5: $q \leftarrow q+1$ 로 둔다. 만약, $q \leq Q_l$ 이면 단계 4.2로 간다.

단계4.6: $l \leftarrow l+1$ 로 둔다. 만약, $l \leq L$ 이면 $q=1$ 로 하고, 단계 4.2로 간다.

단계5: 종료조건검사

종료조건을 만족하면 알고리즘을 종료하고, 그렇지 않으면 단계 2로 간다.

그림 2. SMCA의 단계별 절차

단계 1에서, 계층 1은 각 모집단이 나타내려는 부분해의 표현방법에 따라 임의로 개체를 생성한다. 계층 2부터 계층 L의 개체들은 직하위 모집단의 표현을 결합한 형태로 임의 생성한다. 여기서, 직하위 모집단이란 계층도에서 가지가 직접 연결된 하위계층의 모집단을 말한다. 따라서, SMCA를 실제 문제에 적용할 때에는 계층 1의 각 모집단에 대한 표현방법만 결정하면 된다. 이러한 표현방법은 자연계에서 분리되어 생존하는 공생자(symbiont)들이 좀더 진화해 나가면서 서로 결합한 상태로 생존하는 내공생자(endosymbiont)의 관계를 갖는 것과 같다. 이는 내공생 진화알고리즘(endosymbiotic evolutionary algorithm)에서 사용한 표현방법을 채용한 것이다. 이 알고리즘은 이러한 개체표현을 사용함으로써, 여러 공진화 알고리즘들 중에서 탐색성능이 우수하다고 알려져 있다(Kim et al., 2001). 초기 적응도 평가와 관련된 내용은 단계 3에서 언급한다.

단계 2는 진화의 대상이 되는 이웃을 설정하는 단계이다. 진화범위가 이웃으로 한정됨으로써, 다양하고 좋은 개체들이 적소(niche)를 형성하도록 하고, 해의 조기 수렴을 방지할 뿐만 아니라 해 공간의 효율적 탐색을 가능하게 한다(Kim et al., 2000b).

단계 3은 개체의 적응도를 평가하고, 계층간 상호작용을 하는 단계이다. 계층 1부터 계층 L-1을

구성하는 개체들은 통합문제에 대한 부분해들이므로, 적응도는 동일한 계층의 다른 모집단내 개체들과 결합되어야만 평가할 수 있다. 적응도 평가를 위한 다른 모집단의 개체를 공생파트너(symbiotic partner)라고 부른다. 공생파트너의 선택전략은 다양할 수 있으나, 이들은 알고리즘의 탐색성능에 통계적으로 유의한 차이를 보이지 않은 것으로 나타났다(Kim *et al.*, 2000a). 따라서 본 연구에서는 구현의 용이성 및 계산의 편의성을 고려하여, 단계 1에서는 동일한 계층의 같은 위치에 있는 다른 모집단의 개체들을 공생파트너로 선택하였고, 단계 3에서는 다른 모집단의 이웃에서 임의로 한 개체를 선택하는 전략을 사용하였다. 단, 최상위 계층인 계층 L 은 하나의 모집단만 존재하고, 이 모집단의 개체들은 모든 하위 모집단의 개체들이 결합되어 완전한 해를 표현하므로 공생파트너없이 독립적으로 적응도를 평가할 수 있다. 한편, 단계 3.3에서 이루어지는 계층간 상호작용은 적응도가 높은 개체 결합이 발견되면, 적응도를 기준으로 하여 상위 계층으로 전달하는 과정을 통해 이루어진다.

단계 4에서는 모든 계층에 존재하는 모집단의 이웃들이 진화한다. 이때 진화는 성능향상을 위하여 안정상태 유전알고리즘(steady-state genetic algorithm) 형태를 따른다. 단계 4.3에서 대체될 개체들은 낮은 적응도를 갖는 개체의 선택확률을 높게 하는 확률바퀴 선택전략에 의해 결정된다. 한편, 실제로 유전연산을 위해서는 개체표현과 유사하게 계층 1에 있는 모집단에 적합한 연산자들만 개발하면 된다. 계층 2부터 계층 L 에 있는 개체들은 자신의 직하위 모집단들에 적용되었던 유전연산자들을 그대로 사용하면 된다.

3. FMS 공정계획문제

3.1 개요

FMS 공정계획문제는 부품의 형상, 원자재, 공차 등의 부품관련 정보와 가공 기술적 제약을 포함한 기계관련 정보를 이용하여 각 부품마다 공정경로를 결정하고, 공정에 기계와 공구를 할당하는 문제이다. FMS에서 각 부품의 가공정보는 다른 부품의 생산에 영향을 미친다. 즉, 부품별 공정계획문제는 통합문제의 부분문제가 되고, 모든 부분문제가 동시에 해결되어야만 FMS의 완전한 공정계획이 수립되는 상황이 된다. 또한, 각 부품별 문제를 작은 단위의 완전한 문제로 보면, 각 공정에 기계와 공구를 할당하고, 공정경로를 결정하는 문제는 각 부품별 문제의 부분문제가 된다. 따라서, FMS 공정계획문제는 동일한 계층에 있는 문제들은 관련성을 가지고 상호작용하며, 계층간에는 하위 계층의 단위들이 결합되어 그 상위계층의 단위를 구성한다. 또한, FMS 공정계획문제는 결정변수들이 계층도상에서 서로 동일한 수준에 위치하므로, 각 부분문제의 계층깊이가 동일한 다계층 구조를 형성한다고 볼 수 있다.

FMS 공정계획문제는 문제에서 고려하는 유연성(flexibility)의 종류 및 범위에 따라서 결정변수의

종류 및 수, 그리고 문제의 복잡도와 구조가 결정된다(Guerrero *et al.*, 1999). 본 연구는 SMCA의 적용성을 보이기 위하여 FMS 공정계획문제를 다룬 것이다. 따라서, 여기에서는 문제를 정의하는데 요구되는 유연성의 측면에서만 기존 연구들을 살펴본다. FMS의 유연성들에 대한 분류와 정의는 Stecke and Raman (1995)의 연구를 참조할 수 있고, FMS의 여러 생산계획문제의 해결기법들에 대한 연구현황은 Guerrero *et al.*(1999)나 Vidyarthi and Tiwari(2001) 등을 참조할 수 있다.

기존 연구들은 고려할 수 있는 유연성의 종류를 축소하거나(Kuhn, 1995; Nayak and Acharya, 1998), 다양한 유연성을 고려하였더라도 유연성으로 인하여 발생하는 가능한 경우의 수를 제한시키는 형태(Liang and Dutta, 1993; Guerrero *et al.*, 1999; Tiwari and Vidyarthi, 2000; Kumar and Shanker, 2000)로 문제를 해결하였다. 즉, 각 공정들은 여러 대안기계에 의해 작업 가능하지만 사용되는 공구는 고정된 한 두개로 가정하거나, 반대 경우로 단순화하였다. 또한, 대안 공정경로는 거의 고려되지 않았고, 고려한 연구들에서도 이들에 대한 경우의 수를 제한하였다. 문제의 단순화는 FMS에서 유연성이 갖는 강점을 약화시킨다.

3.2 문제정의

본 연구에서 다루는 FMS 공정계획문제는 여러 개의 부품들을 자원 제약 하에서 생산함에 있어, 각 공정에 기계와 공구를 할당하고, 공정경로를 결정하는 문제로 정의한다. 이때, 유연성으로는 기계(machine), 공구(tool), 공정(process), 순서(sequence)의 4가지 유연성을 고려한다. 기계유연성은 대안기계에 의한 작업 실행의 가능성으로, 공구유연성은 대안 공구에 의한 작업 실행의 가능성으로 정의한다. 공정유연성은 같은 형상을 가공하는데 있어 여러 대안공정들이 있음을 말하며, 순서유연성은 제조 공정간의 순서가 바뀔 수 있는 가능성을 말한다.

FMS 공정계획은 다양한 목적들에 의해 결정될 수 있다(Guerrero *et al.*, 1999; Vidyarthi and Tiwari, 2001). 본 연구에서는 위의 4가지 유연성이 모두 영향을 미치는 목적함수로 셋업비용(setup cost) 최소화를 사용한다. 이때, 셋업은 연속되는 두 공정들간의 기계와 공구의 변경만을 고려하고, 공정 변경에 따른 셋업은 고려하지 않았다. 목적함수를 수식으로 표현하면 식 (1)과 같다.

$$SC = MC \times \sum_i \sum_j \Phi(M_{x(j)}, M_y) + TC \times \sum_i \sum_j \{ [1 - \Phi(M_{x(j)}, M_y)] \times \Phi(T_{x(j)}, T_y) \} \quad (1)$$

$$\Phi(x, y) = \begin{cases} 1, & \text{if } x \neq y, \\ 0, & \text{otherwise.} \end{cases} \quad (2)$$

여기서, MC 는 기계변경에 따른 단위당 셋업비용, M_y 는 부품 i 의 j 번째 공정을 수행하는 기계번호, TC 는 공구변경에 따른 단위당 셋업비용, T_y 는 부품 i 의 j 번째 공정을 수행하는 공구번호를 나타낸다. 그리고 식 (1)내의 함수는 식 (2)와 같다. 기계내

공구간 셋업 및 기계간 셋업은 생산시간의 소비를 포함할 수 있으므로, 산출물 완성시간의 최소화 관점에서 중요한 목적이 된다(Stecke, 1983; Modi and Shanker, 1994).

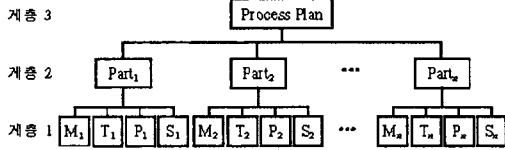
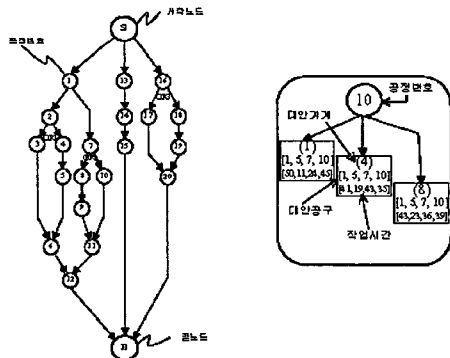


그림 3. FMS 공정계획문제의 대칭형 다계층 구조

본 연구에서 다루는 공정계획문제를 계층화하여 보이면, <그림 3>과 같은 대칭형 다계층 구조가 된다. 최상위 계층인 계층 3은 완전한 공정계획을 나타내는 모집단(Process Plan)이다. 계층 2는 공정계획의 대상이 되는 부품별 모집단(Part, $i=1,2,\dots,n$)으로 구성된다. 그리고 최하위 계층인 계층 1은 부품별 기계(M_i), 공구(T_i), 공정(P_i), 그리고 순서(S_i)를 나타내는 모집단으로 구성된다. 유연성은 최하위 계층을 구성하는 모집단의 수를 다르게 하므로, 공정계획문제의 구조에도 영향을 미친다.

3.3 표현 및 가점

본 연구에서는 기계, 공구, 공정, 순서유연성을 동시에 고려하면서도 기존 FMS 연구들과 다르게 유연성으로 인하여 형성되는 가능한 해공간을 인위적으로 축소하지 않고자 한다. 이를 위하여, 각 부품의 가공정보들은 Ho and Moodie(1996)에 의해 제안된 네트워크 표현(network representation) 형태로 주어졌다고 가정한다.



(a) 부품별 가공정보 (b) 한 공정의 세부정보

그림 4. 네트워크표현에 의한 부품의 가공정보

<그림 4>는 네트워크 표현으로 주어진 부품의 가공정보를 예로 보인 것이다. <그림 4>의 (a)에서 시작 및 끝노드는 가상(dummy)노드로, 제조공정이 있어 시작과 끝을 나타낸다. 중간노드들은 <그림 4>의 (b)에서 보는 바와 같이 공정번호, 각 공정을 수행할 수 있는 대안기계 및 대안공구, 그리고 이에 대한 작업시간을 포함하고 있다. 예를 들어, <그림 4>의 (b)는 <그림 4> (a)의 공정 10을 확대해 보인

것이다. 10번 공정은 기계 1, 4, 8중 하나의 기계에서 작업될 수 있고, 3대의 기계들은 공구 1, 5, 7, 10중 하나를 선택할 수 있다. 이때 작업시간은 기계에 따라 다를 수 있다. 중간노드에 포함된 내용은 기계 유연성과 공구유연성을 설명할 수 있다. 노드들을 연결하는 화살표는 공정간의 선후행관계를 나타낸다. 'OR'관계는 동일한 형상을 가공하는데 대안공정이 존재할 수 있다는 공정유연성을 표현한다. 만약, 노드의 연결이 'OR'기호에 나누어지면 어느 한 가지의 공정만을 따른다. 'OR'기호와 연결되지 않는 노드들은 반드시 수행되어야 함을 뜻하고, 다양한 가공순서를 포함한다. 네트워크표현은 각 부품의 가공순서가 유일한 선형순서(linear order)가 아닌 다양한 부분순서(partial order)를 가짐을 보일 수 있으므로, 순서유연성의 표현이 가능하다.

FMS는 유연성뿐만 아니라 여러 요소들에 의하여 문제를 다르게 정의할 수 있으므로, 다루는 문제의 범위를 명확히 규정해야 한다. 본 연구에서는 FMS가 다음 상황들에 의해 운영된다고 본다. 1) 생산계획기간은 주어졌고, 이 기간내에 생산해야 할 부품유형은 선택되어졌다. 2) 기계는 적절한 공구가 주어지면, 다양한 공정들을 수행할 수 있다. 3) 기계에 장착되는 공구매거진은 다른 한계용량을 갖는다. 4) 공구는 다양한 공정들을 수행할 수 있고, 공구유형별로 장착되는 기계에 무관하게 동일한 슬롯요구량을 갖는다. 5) 각 부품 및 지원장치에 준비시간 및 이동시간은 가공시간에 포함된다. 6) 선점(preemption)이 허용되지 않고, 한 공정이 완료될 때까지 기계는 변화되지 않는다. 7) 공구 수명은 고려하지 않는다.

4. FMS 공정계획문제를 위한 유전요소

4.1 적용도함수

FMS 공정계획문제는 다양한 제약들을 갖는다. 본 연구에서는 제약의 유형별로 다르게 처리하였다. 즉, 대안기계제약, 대안공구제약, 대안공정제약, 그리고 대안순서제약들에 대해서는 이들을 항상 만족하는 유전표현 및 유전연산을 개발하여 사용한다. 반면, 공구매거진의 한계용량제약과 각 공구의 가용공구수 제약은 벌금함수(penalty function)로 처리하였다. 또한, 본 연구에서는 각 기계부하량에 대한 상한제약을 추가하였다. 셋업비용 최소화 측면에서 공정을 기계에 할당하면, 특정한 몇개의 기계에 작업량이 집중될 수 있고, 불균형적인 작업할당은 전체적으로 기계의 유휴시간을 증가시킬 수 있다(Modi and Shanker, 2000). 따라서, 기계부하량의 상한을 이용하여, 기계의 작업량이 지나치게 불균형적으로 할당되는 것을 방지하고자 한다. 기계 부하량에 대한 제약도 제약의 위반 정도에 따라 벌금을 부과하였다.

식 (3)은 모집단의 위치 (a, b)에 있는 개체의 셋업비용과 계약어김에 대한 벌금을 합한 값을 계산하는 식이다. 식 (3)의 첫번째 항은 위치 (a, b)에 있는 개체의 셋업비용으로 식 (1)에 의해 구한다. 두번째 항부터 네번째 항은 각각 가용

공구수 제약(Tool Penalty: TP), 공구매거진의 한계 용량 제약(Magazine Penalty: MP), 그리고 기계 부하량의 상한제약(Load Penalty: LP)에 대한 벌금함수를 표현한 것이다. 함수 $TP(t)$ 는 위치 (a, b) 에 있는 개체에 할당된 공구의 개수가 그 공구의 가용 공구수를 초과하면, 초과된 값을 구하는 함수이다. 함수 $MP(m)$ 은 기계 m 에 장착된 공구매거진의 한계용량제약을 어긴 정도를 표현한 것이며, 함수 $TP(t)$ 와 같은 방법으로 계산한다. 마지막으로 함수 $LP(m)$ 은 기계 m 의 부하량이 상한 제약을 어긴 정도를 표현한 것이다. 벌금계수 $c_1, c_2, c_3, \alpha, \beta, \gamma$ 는 파라미터이다. 본 연구에서 다루는 문제는 최소화문제이므로, 개체의 적응도는 식 (3)의 값이 낮을수록 높은 적응도를 부여받을 수 있도록 정규화하여 사용하였다.

$$g_{19}(a,b) = SC(a,b) + c_1 \sum_i TP(t)^{\alpha} + c_2 \sum_m MP(m)^{\beta} + c_3 \sum_m LP(m)^{\gamma} \quad (3)$$

4.2 유전표현 및 초기모집단 생성

앞에서 언급한 바와 같이, SMCA에서는 계층 1의 각 모집단에 대한 표현방법만 결정하고, 계층 2부터 최상위 계층의 모집단들에 대해서는 직하위계층의 표현을 결합한 형태로 모집단을 구성한다. 따라서, 여기에서는 계층 1에서 사용한 유전표현과 초기 모집단 생성방법만을 언급한다.

기계모집단에서 각 개체의 j 번째 인자값은 공정 j 의 수행가능한 기계번호를 나타내고, 공구모집단의 개체도 j 번째 인자값은 공정 j 의 수행가능한 공구번호를 나타낸다. 두 모집단의 초기해는 각각의 공정별로 주어진 대안기계와 대안공구하에서 임의로 생성한다.

공정모집단의 개체는 대안공정을 표현하기 위하여 'OR'기호 개수만큼의 개체길이를 갖는다. 인자의 위치는 'OR'기호의 번호를 나타내고, 인자값은 그 위치에 해당하는 'OR'기호에 의해 분기되는 최대가치수를 B 라 한다면 $[1, B]$ 구간의 정수값을 갖는다. 만약, 그 값이 b 라면 왼쪽에서 b 번째 가지의 공정들을 수행하는 것으로 해석한다. 이 모집단의 초기해는 인자위치마다 취할 수 있는 정수값내에서 임의로 선택하여 생성한다.

순서모집단의 개체는 순열표현으로 각 부품을 완성하기 위한 가공순서를 표현한다. 이때, 개체에 나열된 공정들의 순서는 반드시 작업 선후행 관계를 만족한다. 이때, 이 순열은 모든 가능한 공정들을 포함하지만, 대안공정들로 인하여 실제로 수행되지 않는 공정들을 포함할 수 있다. 완전한 하나의 공정경로는 가공순서를 나타내는 순열에서 공정모집단의 개체가 나타내는 OR노드의 선택되어지지 않는 공정들을 삭제하면 결정된다. 순서모집단의 초기해는 순서제약을 만족하는 가공순서를 갖는 개체들을 임의로 생성하여 구한다. 그 절차는 다음과 같다. 먼저, 선행공정이 없는 공정들의 집합 A 를 생성하고, 집합 A 에서 임의로 한 공정을 선택하여 순서열에 첨부한다. 선택된 공정을 집합 A 에서 삭제하고, 그 공정의 직후행 공정을 집합

A 에서 삽입한다. 이 과정을 집합 A 가 공집합이 될 때까지 반복하면, 순서제약을 만족하는 가능해들이 생성된다.

4.3 유전연산자

SMCA에서의 유전연산은 계층 1에 있는 모집단에 적합한 연산자들만 개발하여 적용하고, 계층 2부터 계층 L 에 있는 개체들은 자신의 직하위 모집단들에 적용되었던 유전연산자들을 그대로 사용하던 된다. 따라서, 여기에서는 계층 1의 각 모집단을 위한 유전연산자들만을 설명한다.

기계 및 공구모집단은 이점교차를, 공정모집단은 일점교차를 사용한다. 세 모집단에 대한 돌연변이는 모두 각 인자에 돌연변이율을 적용하여 대안기계, 대안공구, 대안공정('OR'기호의 가지번호)중 임의로 선택하는 방법을 택하였다.

순서모집단의 개체들은 수정(modified)이점교차를 사용한다. 수정이점교차의 과정은 다음과 같다. 먼저 두개의 임의의 절단점을 선정한다. 두 절단점에 의하여 나누어진 앞부분, 중간부분, 뒷부분에 있는 인자중 앞부분과 뒷부분은 한 부모(P1)로부터 자손(O1)의 같은 위치에 복사하고, 복사된 인자들과 같은 다른 부모(P2)의 인자들을 지운다. P2에 남아있는 인자들은 P2에 나타나는 순서대로 O1의 채워지지 않은 위치에 순서대로 복사한다. 수정이점교차는 새롭게 생성된 자손개체가 선후행 순서제약을 만족하도록 한다. 이 모집단의 돌연변이 방법은 다음과 같다. 먼저, 부모개체로부터 한 인자를 임의로 선택한다. 돌연변이 후의 개체가 선후행 관계를 어기지 않도록 하기 위하여, 선택된 인자의 모든 직선행 작업과 직후행 작업을 조사하여 인자의 삽입가능 범위를 구한다. 이 범위내에서 선택된 인자를 임의의 위치에 삽입한다.

5. 실험 및 결과

5.1 실험문제

SMCA의 합리적인 성능 평가를 위하여 다양한 실험문제를 생성하였다. 특히, 각 유연성의 정도는 문제의 복잡도에 많은 영향을 미치므로, 4가지 유연성의 정도를 판단하는 척도를 제안하고, 이 척도에 따라 유연성의 수준이 다양한 실험문제를 생성하였다.

기계유연성의 척도는 공정당 평균 대안기계수로 하였다. 즉, 하나의 부품을 생산하기 위한 모든 공정의 대안기계수 총합을 공정수로 나눈 값을 사용한다. 공구유연성도 공정당 평균 대안공정수로 하였고, 기계유연성과 유사한 방법으로 계산된다. 공정유연성은 대안공정들로 인한 가능한 가지의 수로 평가하였다. 순서유연성은 조립라인밸런싱에서 선행관계의 유연성을 평가할 때 사용되는 방법으로 평가하였고, 구체적인 절차는 Kim *et al.*(2003b)의 연구를 참조하기 바란다.

<표 1>은 실험문제를 생성하기 위하여 다양한 유연성을 조합으로 갖는 18개 부품들의 가공정보를 보인 것이다. 본 논문에서는 지면제약상 네트워크를 보이진 않았으며, 구체적인 부품들의 가공정보는

Kim(2002)에서 참조할 수 있다. 실험문제는 <표 1>에서 보인 18개의 부품들을 적절히 조합하여 <표 2>와 같이 18개의 문제로 구성하였다. 실험문제들은 문제크기(부품 및 공정수)와 유연성 측면에서 다양한 범위를 갖도록 하여 실험문제가 한쪽으로 치우친 특성을 갖지 않도록 하였다. <표 2>에서 부품수는 공정계획의 대상이 되는 부품수를 의미하고, 총공정수는 각 문제마다 대안공정을 포함하여 수행되어야 할 총공정수를 순서대로 나열한 것이다. 각 실험문제에 포함된 구체적인 부품번호는 Kim *et al.*(2003a)의 연구를 참조하기 바란다.

표 1. 부품별 유연성

부품 공정 번호 수	유연성 척도값 기계 공구 공정 순서	부품 공정 번호 수	유연성 척도값 기계 공구 공정 순서
1	12 5.38 6.88 4 0.58 10	11 4.91 2.18 2 0.39	
2	14 5.00 4.88 2 0.61 11	15 9.78 3.22 3 0.80	
3	19 6.68 4.58 4 0.84 12	18 3.78 2.22 2 0.74	
4	16 3.94 1.94 4 0.38 13	18 6.50 3.33 5 0.33	
5	18 6.17 2.08 3 0.51 14	13 5.85 4.00 4 0.58	
6	20 4.65 2.35 3 0.72 15	15 3.33 3.40 4 0.73	
7	21 5.57 2.88 9 0.33 16	21 6.24 4.76 6 0.38	
8	20 3.50 3.05 10 0.41 17	22 2.86 4.55 12 0.69	
9	20 6.30 3.20 8 0.72 18	17 5.12 4.82 8 0.72	

표 2. 실험문제구성

문제번호	부품수	문제별 총공정수
P01~P06	6	100, 121, 99, 105, 76, 105
P07~P12	9	142, 168, 150, 160, 155, 151
P13~P16	12	189, 221, 201, 211
P17	15	262
P18	18	310

5.2 제약 및 실험설계

본 연구에서 다루는 FMS는 10대의 기계, 그리고 20개 유형을 갖는 공구로 구성되었다고 가정하였다. 공구매거진의 한계용량('용량'), 공구별 가용공구수('수') 및 슬롯요구량('슬롯')은 <표 3>과 같이 사용하였다. 이때, 공구매거진의 용량은 모든 유형의 공구가 갖는 슬롯요구량의 총합을 초과하지 않도록 하였고, 각 공구유형별 가용공구수는 다양한 값을 사용하여 제약의 강약을 표현할 수 있도록 하였다.

표 3. 기계 및 공구관련 제약

기계		공구	
번호	용량	번호	수(슬롯)
1	28	1	6(1)
2	31	2	7(2)
3	38	3	6(2)
4	28	4	6(2)
5	36	5	10(1)
6	37	6	6(2)
7	29	7	10(3)
8	28	8	9(2)
9	29	9	7(1)
10	26	10	5(2)

실험을 위하여 모든 알고리즘은 C++ 프로그램 언어로 구현되었으며, 1.7GHz Pentium CPU를 장착한 IBM-PC에서 수행되었다. 파라미터들은 예비실험을

통하여 우수한 성능을 보인 것으로 결정하였다. 식 (1)에서 기계의 단위당 셋업비용(MC)은 5, 공구의 단위당 셋업비용(IC)은 1로 하였다. 식 (3)에서 각 기계부하량의 상한은 실험데이터에서 각 기계의 평균작업시간을 구하고, 최대 평균작업시간의 2배를 부여하였다. 또한, 벌금계수 중에서 c_1, c_2, c_3 는 모두 30으로 두었고, α, β, γ 는 모두 0.5로 두었다. 모집단의 크기는 10×10 의 격자구조로 하였다. 모든 모집단에서 돌연변이는 0.15의 개체 돌연변이율에 따라 돌연변이시킬 개체를 선택한 후, 0.3의 인자 돌연변이율에 따라 해당된 인자를 돌연변이시켰다. 단, 공정모집단의 개체 길이는 다른 모집단의 개체 길이에 비해 짧으므로, 공정모집단의 인자 돌연변이율은 0.05로 하였다. 알고리즘의 종료조건은 각 모집단들의 개체 평가횟수로 하였고, 각 문제마다 해공간의 차이를 반영하기 위하여 달리 두었다. 즉, 공정계획의 대상이 되는 부품수에 따라 6개인 문제들은 각 모집단의 개체평가횟수가 $7.0E+03$ 이던 종료하였다. 부품수가 9개인 문제는 $1.0E+04$, 12개인 문제는 $1.1E+04$, 15개인 문제는 $1.2E+04$, 18개인 문제는 $1.3E+04$ 만큼 개체를 평가하면 알고리즘을 종료하였다.

5.3 실험결과 및 분석

SMCA의 탐색 성능은 다음 두가지 알고리즘과 비교한다. 첫번째 기법은 하나의 개체에 표현하고자 하는 모든 정보를 표현하고, 단일 모집단을 운영하는 알고리즘이다. 이를 TEA(Traditional Evolutionary Algorithm)라 부르기로 한다. TEA는 SMCA에서 최상위 계층인 계층 3만을 운영하는 것과 같다. 두번째 기법은 Kim *et al.*(2003b)이 제안한 SEA(Symbiotic Evolutionary Algorithm)이다. SEA는 복수개의 모집단을 구성하지만, 이들이 단일계층을 이루고 있는 형태이다. SEA는 SMCA에서 계층 1만을 운영하는 것과 같다. SEA에 대한 구체적인 절차는 Kim *et al.*(2003b)의 연구를 참조하기 바란다. 실험에서 모든 알고리즘에 사용된 진화전략, 파라미터 및 유전요소들은 제안한 알고리즘과 동등한 조건으로 설정하였다.

<표 4>는 TEA, SEA, SMCA에 의해 구한 각 문제의 총셋업비용을 나타낸 것이다. 아래 표에서 'Best', 'Mean', 'Std', 'Cnt'는 각각 실험문제마다 10회 반복실험을 통해 얻은 값에 대한 가장 좋은 값, 평균값, 표준편차, 그리고 10회 반복실험중 가능해 발견 횟수를 의미한다. 아래 표에서 '-'(dash)는 그 값을 찾지 못했거나 계산이 불가능한 경우를 뜻한다. 마지막 열은 각 문제마다 TEA와 SEA중 더 높은 성능을 보인 값에 대한 SMCA의 개선율을 $\{(TEA \text{ 또는 } SEA \text{의 mean} - SMCA \text{의 mean}) / TEA \text{ 또는 } SEA \text{의 mean}\} \times 100(\%)$ 로 나타낸 것이다.

실험결과, 모든 실험문제에서 SMCA가 가장 좋은 성능을 보였고, 그 다음으로 SEA가 좋은 결과를 보였으며 TEA가 가장 낮은 성능을 보였다. 또한, 반복실험에 대한 표준편차 측면에서도 SMCA는 다른 두 알고리즘에 비해 비교적 낮은 값을 보임에 따라 반복실험의 과정에서 안정적으로 해를 탐색해 나간다고 말할 수 있다. SMCA는

반복실험중 가능해를 발견한 횡수 측면에서도 다른 두 알고리즘보다 월등히 우수하다. 실험에 의하면, 본 연구에서 사용한 기계부하량 상한계약에 의해 TEA와 SEA가 발견한 가장 좋은 해들 중에는 비가능해로 분류되는 경우가 나타났다. 따라서 TEA나 SEA는 기계부하량의 평활화 측면에서 SMCA보다 낮은 성능을 보임을 확인할 수 있다. 이는 Kim *et al.*(2003a)의 연구결과에서도 확인할 수 있다. TEA는 문제크기가 커지면, 거의 대부분의 문제에서 기계부하량의 상한계약을 만족하는 해를 찾지 못하였다. SEA는 TEA와 SMCA의 중간정도 성능을 보이지만, 이 알고리즘 역시 문제크기가 커짐에 따라 가능해를 탐색하는 횡수가 적어지는 것으로 나타났다. SMCA는 유연성의 종류와 정도면에서도 크게 영향을 받지 않고 좋은 성능을 보였다.

TEA가 해의 질 측면에서 가장 좋지 않은 결과를 보인 것으로 보아, 복잡한 해공간을 갖는 문제 또는 여러 특성을 동시에 고려한 문제를 해결함에 있어 하나의 해들이 단일 모집단 내에서 운영되는 것은 SEA나 SMCA와 같이 몇 개의 부분해가 복수개의 모집단에 존재하면서 탐색하는 기법보다 좋지 않은 결과를 보인다고 말할 수 있다. 또한, SEA는 FMS의 공정계획문제를 단일계층으로 표현하였고, SMCA는 다계층으로 표현하였다. 문제의 정보를 다양하게 표현했다는 것은 해공간의 탐색을 다양하게 할 수 있다는 의미를 가질 수 있다. 이러한 이유로, SMCA는 SEA보다 탐색공간내에서의 이동 측면에서 다양하면서도 집중적이어서 좋은 해의 탐색 기회가 높을 것으로 예상된다. SMCA가 SEA보다 좋은 결과를 보이는 또 하나의 이유는 적용도를 기준으로 하는 계층간 상호작용에 기인한 것으로 보인다. 진화가 이루어지는 동안, 상위계층의 모집단들에는 하위계층에서 발견된 좋은 해들이 선별적으로 이동한다. 그리고 이들은 동일 계층 내에서 또다시 진화하므로, 좋은 해로의 탐색을 촉진하게 된다. 이는 SMCA의 기본 프레임이 되는 내공생 진화 알고리즘의 개념이기도 하다.

계산소요시간 측면에서 SMCA는 두 알고리즘에 비해 다소 불리하다. SMCA는 SEA보다 많은 모집단과 계층을 운영하고, 추가되는 모집단과 계층별로 유전연산과 진화를 위한 계산시간이 요구된다. 또한, SMCA는 TEA보다 SEA와의 비교에서 불리한 요소들과 함께 개체의 적응도 평가시 공생파트너를 선택하는데 필요한 계산시간이 요구된다. SMCA의 계산시간을 평균적으로 보면, P01~P06은 34초, P07~P12는 168초, P13~P16은 279초, P17은 511초, P18은 804초 정도 소요되었다. SEA는 SMCA보다 15%정도 덜 소요되었고, TEA는 SMCA보다 25%정도 덜 소요되었다. 그러나 <표 4>의 마지막 열에서 보듯이, FMS 공정계획문제와 같이 복잡도가 높은 문제에서는 해의 개선율이 계산소요시간의 증가에 따른 불리함을 상쇄시키는 효과로 제시될 수 있을 것이다.

6. 결론

본 연구에서는 대칭형 다계층 통합문제를 효과적으로 해결할 수 있는 일반화된 방법론으로 진화알고리즘의 특별한 형태인 SMCA를 제안하였다. 그리고 SMCA를 이용하여 FMS 공정계획문제를 해결함으로써 알고리즘의 적용성을 검증하였다. 실험결과, 유연성의 수준들을 다양하게 고려한 모든 실험문제에서 SMCA는 기존 알고리즘들보다 우수한 성능을 보였다. 특히, 다른 기존 알고리즘들은 실험문제의 해공간이 커짐에 따라 성능이 하락하였으나, SMCA는 이러한 경향을 보이지 않았다. 본 연구에서 제안한 SMCA는 계층수나 모집단수에 대한 제한을 받지 않고 다양한 대칭형 다계층 통합문제에 적용 가능하다.

감사의 글

본 연구는 한국과학재단 지역대학 우수과학자 지원 연구(R05-2002-000-00474-0) 지원으로 수행되었음.

표 4. SMCA의 탐색성능비교

문제	비교 알고리즘												개선율 (%)
	TEA				SEA				SMCA				
	Best	Mean	Std	Cnt	Best	Mean	Std	Cnt	Best	Mean	Std	Cnt	
P01	116.0	130.0	19.8	2	104.0	115.1	5.2	8	104.0	108.1	1.9	10	7.8
P02	138.0	147.0	12.7	2	116.0	124.9	5.6	8	105.0	110.4	3.4	10	11.6
P03	97.0	98.3	2.3	3	94.0	101.2	5.2	10	87.0	88.7	1.4	10	9.8
P04	185.0	188.0	4.2	2	124.0	128.9	3.1	8	115.0	117.3	1.9	10	7.5
P05	252.0	257.0	7.1	2	109.0	116.1	5.5	8	94.0	100.5	3.2	10	13.5
P06	234.0	245.0	12.8	3	110.0	114.2	4.7	9	98.0	102.2	2.1	10	10.5
P07	170.0	178.0	8.5	2	173.0	182.1	4.3	9	161.0	165.5	2.8	10	6.0
P08	342.0	349.5	10.6	2	165.0	177.6	6.7	10	154.0	157.6	2.9	10	11.3
P09	148.0	148.0	-	1	142.0	150.7	6.7	10	130.0	135.5	3.2	10	8.4
P10	212.0	212.0	-	1	203.0	209.7	4.0	6	183.0	186.9	3.3	10	10.9
P11	-	-	-	0	177.0	183.2	4.1	5	155.0	159.7	2.9	10	12.8
P12	199.0	199.0	-	1	158.0	167.0	5.4	10	144.0	147.5	2.1	10	11.7
P13	330.0	330.0	-	1	228.0	235.7	7.3	6	211.0	214.1	2.3	10	9.2
P14	-	-	-	0	233.0	245.6	9.8	5	210.0	215.6	5.2	10	12.2
P15	309.0	309.0	-	1	206.0	213.9	7.1	7	187.0	192.1	2.4	10	10.2
P16	-	-	-	0	255.0	266.2	8.2	5	225.0	234.1	3.7	10	12.1
P17	-	-	-	0	295.0	307.5	10.0	4	273.0	279.2	3.3	10	9.2
P18	-	-	-	0	384.0	393.0	12.7	2	327.0	332.8	4.0	10	15.3

참고문헌

- Guerrero, F., Lozano, S., Koltai, T. and Larraneta, J. (1999), Machine loading and part type selection in flexible manufacturing systems, *International Journal of Production Research*, **37**, 1303-1317.
- Ho, Y.C. and Moodie, C.L. (1996), Solving cell formation problems in a manufacturing environment with flexible processing and routing capabilities, *International Journal of Production Research*, **34**, 2901-2923.
- Kim, J.Y., Kim, Y. and Kim, Y.K. (2001), An endosymbiotic evolutionary algorithm for optimization, *Applied Intelligence*, **15**, 117-130.
- Kim, J.Y., Kim, Y.K., and Shin, T.H. (2000a), Analysis of partnering strategies in symbiotic evolutionary algorithms, *Journal of the Korean Operations Research and Management Science Society*, **25**, 67-80.
- Kim, Y.K. (2002), A set of data for the integration of process planning and scheduling in FMS, available at <http://syslab.chonnam.ac.kr/links/FMSdata-pp&s.doc>.
- Kim, Y.K., Kim, J.Y. and Kim, Y. (2000b), A coevolutionary algorithm for balancing and sequencing in mixed model assembly lines, *Applied Intelligence*, **13**, 247-258.
- Kim, Y.K., Kim, J.Y. and Lee, W.K. (2003a), A multi-level symbiotic evolutionary algorithm for FMS loading problems with various flexibilities, *Journal of the Korean Institute of Industrial Engineers*, **29**, 66-78.
- Kim, Y.K., Park, K. and Ko, J. (2003b), A symbiotic evolutionary algorithm for the integration of process planning and job shop scheduling, *Computers & Operations Research*, **30**, 1151-1171.
- Koza, J.R., (1992), *Genetic Programming*, The MIT Press, Cambridge, MA.
- Kuhn, H. (1995), A heuristic algorithm for the loading problem in flexible manufacturing systems, *International Journal of Flexible Manufacturing Systems*, **7**, 225-250.
- Kumar, N. and Shanker, K. (2000), A genetic algorithm for FMS part type selection and machine loading, *International Journal of Production Research*, **38**, 3861-3887.
- Liang, M. and Dutta, S.P. (1993), An integrated approach to the part selection and machine loading problem in a class of flexible manufacturing systems. *European Journal of Operational Research*, **67**, 387-404.
- Maher, M.L. and Poon, J. (1996), Modelling design exploration as co-evolution, *Microcomputers in Civil Engineering*, **11**, 195-209.
- Modi, B.K. and Shanker, K. (1994), Models and solution approaches for part movement minimization and load balancing in FMS with machine, tool and process plan flexibilities, *International Journal of Production Research*, **33**, 1791-1816.
- Nayak, G.K. and Acharya, D. (1998), Part type selection, machine loading and part type volume determination problem in FMS planning, *International Journal of Production Research*, **36**, 1801-1824.
- Potter, M.A. (1997), The design and analysis of a computational model of cooperative coevolution, Ph.D. dissertation, George Mason University.
- Stecke, K.E. (1983), Formulation and solution of nonlinear integer production planning problem for flexible manufacturing systems, *Management sciences*, **29**, 273-288.
- Stecke, K.E. and Raman N. (1995), FMS planning decisions, operating flexibilities, and system performance, *IEEE Transactions on Engineering Management*, **42**, 82-90.
- Tiwari, M.K. and Vidyarthi, N.K. (2000), Solving machine loading problems in a flexible manufacturing system using a genetic algorithm based heuristic approach, *International Journal of Production Research*, **38**, 3357-3384.
- Vidyarthi, N.K. and Tiwari, M.K. (2001), Machine loading problems of FMS: a fuzzy-based heuristic approach, *International Journal of Production Research*, **39**, 953-979.