韓國經營科學會誌
第21卷 第1號
1996年 4月

FMS 설계시 성능분석 및 팔렛수 결정에 관한 연구*

강석호** · 김철영** · 민학기***

The performance analysis and the determination of the number of pallets in FMS design*

S. H. Kang** · C. Y. Kim** · H. K. Min***

Abstract

We formulate the problem and propose a new heuristic method for determining the number of pallets of each type of component parts produced in FMS. The proposed method uses the Scheweitzer's mean value analysis algorithm of which robustness and fitness for this problem is shown through simulation experiments. This method finds a better solution more efficiently than Solot's PANORAMA, which is a well-known method to determine the number of pallets in FMS.

1. 서 론

FMS는 고전적 제조시스템에 비해서 생산성 향상, 납기단축의 장점을 가지며, 다품종 소량생산에 필요한 유연성을 가진 시스템이지만, 도입시 초기 투자비용이 크므로, 설계에 있어서 세심한 고려가 필요하다.

FMS의 설계단계에서 첫번째로 결정해야 할 것은 대상 부품이다. 결정된 대상부품에 대해서 제조기술이 파악되면 가공장비의 종류가 결정되며, 요구되는 생산용량에 따라 각 장비의 갯수와 자재운반시스템, 저장장소의 형태가 결정된다. 이렇게 FMS의 형태를 결정짓는 주요한 투자결정이 내려진 다음에, 생산률과 공정중 재고를 고려한, 팔렛(pallet)과 치공구(fixture)의 종류와 갯수가 결정되어야 한다.

FMS에는 여러 종류의 부품들이 동시에 가공되며, 팔렛의 종류는 부품의 종류에 따라 달라질 수 있다. 따라서 FMS의 설계시 종류별 팔렛수가 결정되어야 하며, 특히 최종제품의 여러가지 부품을 생산하는 FMS의 경우, 종류별 팔

* 본 연구는 한국과학재단 1992-1993년 연구비 지원에 의해 수행되었음
** 서울대학교 산업공학과
*** 현대전자 시스템소프트웨어실

렛수의 결정문제는 중요한 문제가 된다. 이런 종류의 FMS에서는, 각 부품의 재고를 낮게 유지하면서 최종제품의 원활한 생산이 주요한 목표이며, 이를 위해서는 각 부품의 적절한 생산비가 유지될 수 있도록 종류별 팔렛수가 결정되어야 한다.

FMS의 설계문제를 해결하기 위해서, FMS의 성능평가모형과 최적설계대안을 선정하기 위한 방법에 관한 많은 연구가 행해졌다. 일반적인 FMS의 성능 평가를 위해서는 시뮬레이션 모형과 대기망모형이 주로 사용된다. 시뮬레이션 모형은 정확한 결과를 제공해주지만, 결과를 얻는데 상당한 시간과 노력을 필요로 하는 반면, 대기망모형은 일단 구축이 되면, 빠른 시간내에 결과를 구할 수 있다. 실제 FMS의 설계에서는 고려해야 될 설계대안이 많으므로, 설계문제를 위한 성능평가방법으로서 대기망 모형이 주로 연구되었다. 대표적인 것으로 CAN-Q, MVA(Mean Value Analysis) 알고리즘이 있다.

CAN-Q([10])는 가장 고전적인 모형으로서, 승법해(product form solution) 조건을 만족하는 Central Server 폐쇄 대기망모형을 기초로 하여, 여러 종류의 팔렛이 존재하는 경우를 고려하지 못한다. MVA 알고리즘은 여러 종류의 팔렛이 존재하는 FMS의 분석을 위한 것으로, 각 종류의 팔렛은 각 station에서 같은 가공시간의 분포를 가지는 경우에 적용된다. Suri와 Hilderbrant([8]), Seidmann, Oren, Schweitzer([6])는 팔렛 종류별 가공시간이 다른경우에 적합한 휴리스틱 MVA 알고리즘을 제안하였다.

FMS의 설계는 여러 단계를 거쳐서 행해지며, 각 단계에 대해서, 여러가지 의사결정모형이 개발되어왔다. Vinod와 Solberg([13]), Dalliry와 Frein([1])은 주어진 생산률을 만족하면서, 투자비용을 최소화하는 기계와 팔렛의 수를 결정하는 FMS 설계문제를 다루었다. 그들은 단일 팔렛종류를 가정하고, 종류별 최적 기계수와 최적 팔렛의 수를 발견적기법과 내재열거법(Implicit Enumeration)을 이용하여 구하였는데, 각 대안의 생산률은 CAN-Q를 수행하여 확인하였다.

De Luca와 Perotto([5])는 FMS의 형태와 생산비율이 결정되었을때, 최적 팔렛갯수와 종류별 팔렛의 비를 구하는 방법을 제시했다. 그들은 일단 단일종류의 팔렛으로 가정하고 전체 팔렛수를 결정한다음, 각 종류별 팔렛의 비를 구하는 방법을 제안했다.

Solot([8])은 최종 완제품으로 조립될 여러 종류의 부품이 생산되는 FMS에 대해서, 주어진 평균 산출시간(throughput time)을 만족하면서 팔렛비용을 고려한 최종제품의 생산률을 최대화하는 팔렛수를 결정하는 알고리즘인 PANORAMA를 제안했다. 그는 부품종류별 생산률과 평균산출시간을 계산하기 위해서, 다유형 팔렛을 고려할 수 있게 CAN-Q를 개량한 Multi-Q([9])를 사용하였다. 이 방법은 좋은 해를 제공해주는 것으로 나타났으나, 각 팔렛 종류당 하나씩의 팔렛을 할당한 초기해를 사용함으로써, 많은 수의 팔렛이 존재할 수 있는 경우, 알고리즘의 수행시간이 긴 단점이 있다. 또한 문제의 모형화에서 직관적이지 않은 상수를 도입함으로써, 실제 적용에 어려움이 있다.

본 논문에서는 Solot의 연구에서와 같이, 고유의 팔렛을 필요로 하는 여러 부품을 조립하여 완제품을 생산하는 FMS의 경우에, 재고를 낮게 유지하면서, 최종 완제품의 생산율을 최대화하는 팔렛수를 결정하는 방안을 제시한다. 본 논문에서는 목적함수에 평균산출시간을 반영하고 전체팔렛수의 상한을 제약식으로 설정

한 모형을 제시하고, 근사해를 구하기 위한 해법을 제시한다. 이 해법은 각 부품의 생산률과 평균산출시간의 계산을 위해서 Schweitzer의 MVA 알고리즘([6])을 사용하고, 전체최적을 추구하며 문제의 특성을 반영한 타부탐색 알고리즘을 사용한다.

이하 논문의 구성은 다음과 같다. 2장에서는 MVA 알고리즘의 견실성(robustness)를 알아보기 위한 실험과 결과를, 3장에서는 팔렛수 결정모형과 해법을 제시하고, 알고리즘의 수행도에 관한 분석결과와 기존연구와 비교결과를 기술했다. 마지막으로 4장에서는 결론과 추후연구과제가 제시되어있다.

2. MVA알고리즘의 견실성

2. 1 실험계획

본 논문의 팔렛수 결정문제의 해법에서는 가능한 전체팔렛수와 각 종류별 팔렛수를 변화시켜가면서 해를 찾는다. 이 때, 각 종류별 팔렛수에 대해서 시스템의 여러가지 성능척도는 MVA 알고리즘을 통해서 계산된다. 따라서 MVA 알고리즘의 정확도는 전체 팔렛수와 종류별 팔렛수의 분포의 변화에 따른 영향이 없어야 한다. 예를 들어, 팔렛수가 증가함에 따라서 제품의 생산률을 과대평가한다면, 팔렛수 결정문제의 해법은 실제 최적보다 많은 수의 팔렛을 해로 선택할 것이다.

전체팔렛수와 종류별 팔렛수의 분포에 따른 영향을 조사하기 위해서, MVA 알고리즘의 결과와 시뮬레이션 결과를 비교하였다. 비교를 위한 성능척도로는 본 논문의 팔렛수 결정문제에서 사용되는, 각 팔렛종류별 생산률과 시스템에서 부품의 평균산출시간으로 하였다. 실험은 각각의 팔렛종류가 필요한 3종류의 부품과, 1개의 Load/Unload 스테이션과 4개의 가공스테이션으로 구성된 FMS([12])를 대상으로 하였으며, 전체 팔렛수가 6, 12, 18, 24, 종류별 팔렛의 분포가 1:1:1, 1:2:3, 2:3:1, 1:1:4인 16가지의 경우로 나누어서, 전체팔렛수와 종류별 팔렛의 분포에 의한 영향을 평가하기로 하였다.

시뮬레이션은 16가지 경우 각각에 대해서, 1주(10,000분)를 단위로 10번을 행하여 10개의 표본을 구하였다. 이 시스템은 10,000분내에 안정상태에 도달하는 것으로 관찰되었다.

2. 2 실험결과

전체 팔렛수와 종류별 팔렛의 분포에 따른 팔렛종류별 생산률의 오차를 〈표 1〉에, 평균산출시간의 오차를 〈표 2〉에 나타내었다. 오차는 시뮬레이션과 MVA 알고리즘으로 구한 성능척도값의 차이의 절대값으로 정의하였으며, 오차를 시뮬레이션값에 대한 백분율로 표시하였다. 표에서 X_1, X_2, X_3는 각각 팔렛종류 1, 2, 3의 생산률을 나타낸다.

〈표 1〉과 〈표 2〉를 살펴보면, 팔렛의 분포에 대해서 오차의 영향이 거의 없으며, 전체팔렛수가 적을때 다소 큰 오차가 발생하는 것을 알 수 있다. 일반적으로 생산률은 팔렛의 수에 비례하므로, 생산률의 최대화를 목적함수로 하는 팔렛수 결정문제의 경우, 적은 팔렛수에서 오차는 해법에 큰 영향을 미치지 않을 것이므로, MVA알고리즘은 본 문제의 해법에 적용가능한 성능평가방법이라고 볼 수 있다.

〈표 1〉 전체팔렛수와 팔렛의 분포에 따른 팔렛종류별 생산률의 오차(%)

분포	1:1:1			3:1:2			2:3:1			1:1:4		
팔렛수	X_1	X_2	X_3	X_1	X_2	X_3	X_1	X_2	X_3	X_1	X_2	X_3
6	8.8	8.4	4.4	7.1	13.6	2.5	9.7	5.3	6.5	12.8	9.5	6.0
12	0.0	0.9	3.5	2.3	0.3	6.8	1.0	0.3	2.3	1.4	0.1	3.6
18	0.8	0.4	4.4	0.5	3.1	2.5	1.4	4.3	6.5	2.2	4.0	6.0
24	4.3	9.5	8.9	1.3	4.0	5.6	0.1	0.1	5.6	3.0	1.4	1.9

〈표 2〉 전체팔렛수와 팔렛의 분포에 따른 평균산출시간의 오차(%)

팔렛수/분포	1:1:1	3:1:2	2:3:1	1:1:4
6	9.2	8.9	8.5	10.0
12	1.2	0.7	2.0	0.4
18	1.7	3.5	0.6	2.3
24	1.8	5.0	3.0	3.9

3. 팔렛개수 결정문제

최종제품을 구성하는 여러 부품에 대해서 각 부품의 개수의 상대적 비율을 production-mix 라고 한다. FMS에서 여러 종류의 부품이 한 종류의 팔렛에 장착될 수 있는데, 이 경우에 하나의 팔렛에 할당된 부품의 production-mix 값의 합을 이 팔렛의 pallet-mix라 하고 각 부품의 production-mix에 대한 pallet-mix의 비를 그 부품의 part-mix라고 한다. 예를 들어 최종제품이 4개의 부품 A, B, C, D 각각 1개씩으로 구성되면, A, B, C, D의 production-mix 값은 각각 0.25가 된다. 이 경우, A, B는 팔렛1에 장착되고, C는 팔렛2에, D는 팔렛3에 장착된다고 하면, pallet-mix의 값은 팔렛 1, 2, 3에 대해서 각각 0.5, 0.25, 0.25가 되며, 부품A와 B의 part-mix는 각각 0.5가 되며, 부품 C, D의 part-mix는 각각 1.0이 된다.

FMS에서 재고를 낮게 유지하면서, 원활한 최종제품의 생산을 위해서는, 각 부품의 생산률이 production-mix의 비로 유지되어야 하는데, 이는 각 종류별 팔렛의 생산률을 pallet-mix의 비로 유지하고, 각 팔렛에 대해서, 장착되는 부품의 비를 part-mix로 유지함으로써 이루어질 수 있다. FMS의 운영시에 한 종류의 팔렛에 장착되는 부품들의 생산비는 Loading/Unloading station에서 가공이 끝난

부품을 제거하고, part-mix에 맞도록, 적절한 종류의 부품을 팔렛에 장착함으로써 유지될 수 있다. 그러나 각 팔렛종류별 부품군의 생산률은, 각 팔렛에 장착되는 부품들이 시스템에서 가공되는 시간과 팔렛의 수에 관계된다. 즉 각 팔렛종류에 할당된 부품군의 평균생산률을 pallet-mix에 맞게 유지하기 위해서는, 종류별 팔렛수가 적절하게 결정되어야 한다.

3. 1 팔렛개수 결정문제의 모형화

팔렛개수의 결정문제를 모형화하기 위해서 다음과 같은 기호의 정의가 필요하다.

R : 전체 팔렛 종류의 수

N_r : 팔렛 종류 $r=1,\cdots,R$

$\vec{N}$: $(N_1, N_2, \cdots, N_R)$

N : 전체팔렛수 $N=\sum_{r=1}^{R} N_r$

$\hat{N}$: 전체팔렛수의 상한값

d_r : 팔렛종류 r의 pallet-mix 값, $r=1,\cdots,R$

X_r : 팔렛종류 r에 할당된 부품군의 평균생산율, $r=1,\cdots,R$

$\bar{T}$: 모든 부품종류의 평균산출시간

Z : 목적 함수

K : 목적함수에 사용되는 시스템의 정책에 관련된 가중치

c : 최종제품 생산률에 대한 재공품재고의 반영비율

목적함수의 설정에서 가장 우선적으로 고려되어야 할것은 최종제품의 생산률이다. 최종제품의 생산률은 $\min_{1\le r\le R}\{X_r/d_r\}$ 비례하며, 이 값은 시간당 최종제품 조립에 사용된 총 부품수이다. 예를 들어, 2개의 부품1과 하나의 부품2로 구성된 제품을 생산하는 경우라면, $d_1=2/3$, $d_2=1/3$,이 되며, 부품1이 시간당 6개 부품2가 시간당 2개가 생산된다고 하면, $X_1=6$, $X_2=2$이다. 이 때 시간당 조립가능한 최종제품은 부품2의 생산률에 의해 2개가 되며, 조립에 사용된 부품의 총갯수는 6개가 된다. 즉, $\min\{X_1/d_1, X_2/d_2\}=\min\{9, 6\}=6$은 최종제품의 조립에 사용된 총 부품의 수이며, 이를 하나의 제품에 필요한 총부품수 3으로 나누면 시간당 조립가능한 최종제품의 수가 된다.

재고는 완성부품 재고와 재공품 재고로 나누어 볼 수 있다. 시간당 완성부품재고는 총생산부품수에서 완제품 조립에 사용된 수만큼을 뺀 값이다. 즉, 부품종류 p의 시간당 완성부품 재고는 $X_p-d_p\cdot\min_{1\le r\le R}\{X_r/d_r\}$가 되며, 총 완성부품 재고는 $\sum_{p=1}^{R} X_p-\min_{1\le r\le R}\{X_r/d_r\}$가 된다. 이 식을 살펴보면, 최종제품의 생산률이 최대화되면, 완성부품의 재고가 줄어드는 효과가 나타남을 알수있다.

시스템내의 재공품의 개수는 전체 팔렛수에 비례한다. 이는 또한 부품들의 평균산출시간과 관계가 있다. 일반적으로 부품의 평균산출시간은 시스템내의 부품수에 비례하여 증가한다. 또한 평균산출시간은 새로운 주문에 대처할 수 있는 시스템의 유연성과 밀접한 관계가 있다. 따라서 재공품재고의 경우, 팔렛수를 직접 고려하는 것보다, 전체 부품의 평균산출시간을 고려하는 것이 더 효과적일 수 있다.

이상의 사항을 고려하여, 팔렛갯수 결정문제를 다음과 같은 형태로 모형화하였다.

$$\text{Maximize } Z(\vec{N}) = \min_{1\le r\le R}\{X_r/d_r\} + K\cdot(1/\bar{T}) \quad (1)$$

$$\text{s.t. } N_r \ge 1,\ 1\le r\le R \quad (2)$$

$$\sum_{r=1}^{R} N_r \le \hat{N} \quad (3)$$

$$X_1, \cdots, X_R, \overline{T}) = f(\vec{N}),$$

$$\text{여기서, } f : \mathbb{Z}^R \rightarrow \mathbb{R}^{R+1} \quad (4)$$

목적함수식 (1)은 최종제품의 생산률에 관련된 항 $\min_{1 \le r \le R} \{X_r/d_r\}$과 재공품 재고에 관련된 항 $K \cdot \{1/\overline{T}\}$의 합으로 구성되었다. K는 재공품 재고의 가중치를 나타내는 상수로서 시스템의 정책에 따라 결정되어야 한다. K를 결정하기 위해서는, 우선 최종제품의 생산률에 대한 부품의 재공품재고의 반영비율 c가 결정되어야 한다. 리틀의 법칙에 의하면, $N/\sum_{r=1}^{R} X_r = \overline{T}$이므로, $1/\overline{T}\ \sum_{r=1}^{R} X_r = \overline{T}$이다. $\min_{1 \le r \le R} \{X_r/d_r\}$은 최종조립에 사용되는 부품수를 나타내며, $\sum_{r=1}^{R} X_r$은 생산되는 총부품수를 나타내며 이들은 근사한 값을 가진다. 그런데 전체 팔렛수는 미리 결정될 수 없으므로, 대신 전체 팔렛수의 추정치에 c를 곱해서 K를 구할 수 있다. 예를 들어 최종제품의 생산률에 대해, 평균산출시간의 중요도를 $c = 0.1$로 결정하고, 팔렛수의 추정치를 $\hat{N}/2$로 잡으면, $K = 0.1 \cdot \hat{N}/2$로 결정할 수 있다.

제약식 (2)는 모든 종류의 부품을 생산해야 하기 때문에 팔렛종류별로 최소한 1개씩은 할당해야 한다는 의미이다. 제약식 (3)은 팔렛수는 주어진 상한을 넘지못한다는 것을 나타낸다. 팔렛수의 상한은 FMS의 버퍼의 용량등 물리적인 형태와 관련되어서 결정될 수 있다. 제약식 (4)는 생산율과 평균산출시간은 종류별 팔렛수의 함수라는 의미이며, 실제로 이 관계는 각 $\vec{N}$에 대해서 대기망모형을 통한 해석을 통해서 추정되어져야 한다.

3. 2 팔렛개수 결정문제의 해법

3.1절에서 모형화된 문제에서 제약식 (4)는 명시적인 관계식 형태(closed form)로 표현될 수 없으며, 대기망모형을 통해 계산되어야 한다. 따라서 수리계획법을 사용하는 것이 가능하지 않으며, 또한 대기망모형은 다소간의 오차가 존재하므로, 모든 가능팔렛조합을 나열하여, 엄밀한 최적해를 구하는 방법은 바람직하지 못하다. 본 논문에서는 전체 최적을 추구하는 알고리즘중 하나인 타부탐색방법을 사용한다.

타부탐색방법은 기본적으로 타부리스트에 포함되지 않은 현재해의 이웃해를 탐색해 나가는 방법으로, 현재해를 개선하지 못하는 이웃해를 방문하는 것을 허용하고, 한 번 방문한 해를 타부리스트에 저장함으로써 국소최적에 빠지는 것을 방지하는 알고리즘이다. 일반적으로 타부탐색 알고리즘을 특정문제에 적용하기 위해서는 문제의 특성에 맞는 초기해, 이웃해의 정의, 타부리스트의 유지방법, 종료기준이 결정되어야 한다([2][3]).

알고리즘을 정의하기 위해서 다음과 같은 기호의 정의가 필요하다.

k 알고리즘의 단계수

$\vec{N}^k = (N_1^k, \cdots, N_R^k)$ 알고리즘의 k단계에서 해(현재해)

r^* 최종제품생산률을 결정짓는 팔렛종류, $X_{r^*}/d_{r^*} = \min\{X_r/d_r\}$

L 타부리스트

w 연속해서 해를 개선시키지 못한 단계수

$\hat{w}$ 허용가능한 w의 상한

$S(\vec{N}^k)$ 이웃해의 집합

($\vec{N}^{opt}$, Z^{opt} 현재까지의 최적해와 목적함수값

($\vec{N}^{*}$, Z^{*} 현재해의 최적이웃해와 목적함수값

t_r 팔렛종류 r의 총가공시간

l_r 팔렛종류 r이 시스템에 끼치는 상대적 부하

팔렛갯수 결정문제의 경우, 각 해에 대한 목적함수값의 계산을 위해서는 수행시간이 긴 MVA 알고리즘을 수행하여야 하므로, 탐색할 이웃해를 주의깊게 선정하여야 한다. 즉 목적함수값을 개선할 수 있는, 가능한 적은 수의 이웃해를 탐색해야 한다. 본 논문의 타부탐색 알고리즘은, 다음과 같은 FMS에 대한 두가지 가정을 바탕으로 하고있다. FMS에는 모든 종류의 팔렛이 방문하는 Loading/Unloading station, 세척, 검사 station등이 존재하기 때문에, 이 가정들은 대체로 성립한다고 볼 수 있다.

가정1 한 종류의 팔렛개수가 증가하면, 이 형태의 팔렛에 장착되는 부품의 생산률이 증가한다.

가정2 한 종류의 팔렛개수가 감소하면, 다른 종류의 팔렛에 장착되는 부품의 생산률이 증가한다.

현재해에 대해서 r^* 종류의 팔렛의 개수를 증가시키면 이 종류 팔렛의 생산률이 증가한다. 마찬가지로 r^* 종류가 아닌 다른 팔렛의 개수를 감소시키면 r^* 종류의 팔렛의 생산률이 증가한다. 따라서 현재해 $\vec{N}^k$에 대해서 이웃해 $S(\vec{N}^k$는 다음과 같이 정의할 수 있다.

$$\vec{N}^k = (N_1^k,\cdots,N_r^k,\cdots,N_R^k) \text{에 대해서}$$

$$S(\vec{N}^k = \{(N_1^k,\cdots,N_r^k+1,\cdots,N_R^k)$$

$$\cup\{\vec{N}\,|\,\vec{N}_p = (N_p^k-1,\ p\neq r^*\}$$

팔렛수 결정문제의 경우 해는 R개의 정수로 이루어진 벡터로 표현되므로, 타부리스트는 단계수가 많아도 별로 크지 않다. 따라서 타부리스트는 계속 증가시켜도 큰 문제가 없다. 알고리즘의 종료기준으로는 연속적으로 현재해를 개선시키지 못하는 단계의 수를 사용한다. 즉 $w > \hat{w}$ 이면 알고리즘을 종료하고 현재까지 최적해인 N^{opt}를 해로 제시한다. 초기해는 다음과 같은 휴리스틱 알고리즘을 사용해서 구한다.

R에서 $\hat{N}$ 까지의 전체 팔렛수에 대해서, 각 종류별 팔렛을 시스템에 미치는 부하의 비에 따라 할당하여, 2분할법으로 탐색한다. 각 종류의 팔렛이 시스템에 미치는 상대적인 부하는 팔렛종류 p에 대해서 $l_p = d_p \cdot t_p / \sum_{r=1}^{R} d_p \cdot t_p$로 정의할 수 있는데, 이 값은 실수이므로 라운딩의 문제가 발생한다. 따라서 전체팔렛수 N에 대해서 종류별 팔렛수를 다음과 같이 할당한다. 이하에서는, 팔렛수 N에 대해서 이 할당을 편의상 함수 $f_1 : \mathbb{Z}\rightarrow\mathbb{Z}^R$로 나타내기로 한다.

$$\vec{N} = (N_1,\cdots,N_R) = f_1(N) \text{ such that}$$

$$\mathrm{N_p} = \begin{cases} N_p = 1, & \text{if } N\cdot l_p = 0 \\ [N\cdot l_p] + I_p, & otherwise \end{cases}$$

여기서 I_p는 정수값을 갖는 변수로, $\sum_{p=1}^{R} N_p = N$, $N_p \geq 1$이 될 때까지, $N\cdot l_p - N_p$가 큰 순서대로 $I_p = I_p + 1$을 할당하거나 $N_p > 1$인 p에 대해서 $N\cdot l_p - N_p$가 작은 순서대로 $I_p = I_p - 1$을 할당한다.

리틀의 법칙에 따르면 $X_r = N_r/T_r$이고, T_r은 t_r에 비례하므로, 종류별 팔렛을 시스템에 미치는 부하의 비로 할당하는 것은 휴리스틱적인 관점에서 타당하다고 볼 수 있다. 초기해를 구하는 알고리즘은 다음과 같다.

초기화 N = $\hat{N}$ /2, k=1, $\vec{N}(1)$에 대해서, $Z(\vec{N})$ 계산

Step 1 k=k+1 $N_1 = N - \hat{N}/2^k$,

$N_2 = N + \hat{N}/2^k$

$\vec{N}_1(l) = f_l(N_1)$, $\vec{N}_2(l) = f_l(N_2)$에 대해, MVA 알고리즘을 수행하여,

$Z(\vec{N}_1)$, $Z(\vec{N}_2)$를 계산하여, N을 N, N_1, N_2 중 최적팰렷수로 수정.

Step 2 $\hat{N}/2^k < \hat{w}$이면 *Step 3*으로, 아니면 Step 1로

Step 3 $Z(N_1) < Z(N) < Z(N_2)$이면,

$\vec{N}_l(l) = f_l(N)$

$Z(N_1) < Z(N)$이고 $Z(N_2) < Z(N)$이면,

$\vec{N}_l(l) = f_l(N_1)$

$Z(N_1) > Z(N) > Z(N_2)$이면,

$\vec{N}_l(l) = f_l(N_1 - \hat{N}/2^k)$

알고리즘에 의해 구해진 초기해는 팔렛수를 $\hat{w}$개 증가시키면 비교적 좋은 해에 도달할 수 있는 해이다. 이는 팔렛수를 증가시키면서 해를 탐색하는 것이 유리한 문제의 특성을 반영한 것이다. 알고리즘은 초기화단계에서 *MVA* 알고리즘을 *1*번, 각 단계에서 *MVA* 알고리즘을 두 번 수행하고, 단계를 $\lceil \log_2 | \hat{N}/\hat{w} | \rceil$ 번 수행하므로, *MVA*알고리즘을 $2 \cdot \lceil \log_2 | \hat{N}/\hat{w} | \rceil + 1$번 수행하는 것으로 매우 효율적이다. 예를 들어, $R=3$, $\hat{N} = 40$, $\hat{w} = R$ 이면, MVA 알고리즘을 9번 수행하는 것으로, 이는 주 알고리즘에서 3번의 단계에 해당하는 것이다.

주 알고리즘은 다음과 같다.

초기화 $\vec{N}^1 = \vec{N}_1$, $k=1$, $w=0$

Step 1 제약식 (2)를 만족하는 $S(\vec{N}^k)$를 구하고, $S(\vec{N}^k) = S(\vec{N}^k) - L$,

$S(\vec{N}^k) = \phi$ 면 종료

$S(\vec{N}^k)$의 각 원소에 대하여 MVA 알고리즘을 수행하여 Z 값을 계산

$Z^* = max_{\vec{N} \in S(\vec{N}^k)} \{Z(\vec{N}^k)\}$, $\vec{N}^*$는 Z값을 최대로 하는 $\vec{N}^k$

Step 2 $Z^* \geq Z^{opt}$이면 $w=0$, $Z^{opt} = Z^*, \vec{N}^{opt} = \vec{N}^*$, $L = L \cup \vec{N}^*$

$Z^* < Z^{opt}$이면 $w = w+1$

Step 3 $w \leq \hat{w}$이면 Step 1로,

$w > \hat{w}$이면 종료

3. 3 알고리즘 분석

알고리즘의 수행도를 분석하기위해, 3종류의 팔렛, 1개의 L/U station과 6개의 가공 station으로 구성된 FMS에 대해, 확률적으로 발생된 공정자료에 대해 최적해와 알고리즘의 해를 비교하였다. 알고리즘의 수행도(performance ratio)는 알고리즘의 해와 최적해의 비로 측정하였으며, 최적해는 모든 가능팔렛조합을 나열하여 구하였다.

모형화과정에 결정되는 상수 K와 $\hat{N}$, 그리고 알고리즘 자체의 계수인 $\hat{w}$을 변화시켜가면서 알고리즘의 수행도에 미치는 영향을 조사하였다. K를 결정하기 위한 c값을 0.05에서 0.3까지, $\hat{N}$ 값을 24에서 36까지, $\hat{w}$값을 1에서 2R까지 변화시켜가면서, 4종류의 팔렛믹스에 대해 확률적으로 발생된 공정자료 10개씩, 40개의 표본에 대해서 알고리즘의 수행도를 평균수행

도와 최저수행도를 계산하였다.

$\hat{w}$값에 대한 수행도의 영향은 다음 〈표 3〉에 나타나있다. 알고리즘은 $\hat{w}=1$인 경우에도 비교적 높은 수행도를 보이며, $\hat{w}$값이 증가할수록 평균수행도와 최저수행도가 조금씩 증가하는 것을 알 수 있다. 대체로 $\hat{w} \geq R$인 경우에는 안정된 수행도를 나타낸다고 볼 수 있다.

〈표 3〉 $\hat{w}$ 값에 대한 수행도의 변화

	1	2	$3=R$	4	5	$6=2R$
평균수행도	0.9624	0.9683	0.9706	0.9723	0.9733	0.9745
최저수행도	0.8015	0.8449	0.8449	0.8449	0.8576	0.8576

$\hat{w} = R$에 대해서 $\hat{N}$, K값의 변화에 따른 수행도의 변화는 다음 〈표 4〉와 〈표 5〉에 나타나있다. 대체로 알고리즘의 평균수행도는 $\hat{N}$, K값에 별로 영향을 받지 않는것으로 나타났다. 비교적 큰 K값(c=0.25이상)에서 최저 수행도가 낮아지는 것을 관찰할 수 있는데, 이는 최종제품 생산률의 개선을 지향하는 알고리즘의 특성에 기인한다고 볼 수 있다.

〈표 4〉 $\hat{N}$ 값의 변화에 따른 알고리즘의 수행도

$\hat{N}$	24	27	30	33	36
평균수행도	0.9705	0.9702	0.9774	0.9745	0.9662
최저수행도	0.8404	0.8866	0.8454	0.8316	0.8110

〈표 5〉 K 값의 변화에 따른 알고리즘의 수행도

c	0.05	0.1	0.15	0.20	0.25	0.30
평균수행도	0.9646	0.9663	0.9666	0.9691	0.9680	0.9669
최저수행도	0.8110	0.8426	0.8404	0.8589	0.7775	0.7357

3. 4 기존 방법론과의 비교

본 논문에서 다루는 다유형 팔렛개수 결정문제에 대한 기존의 해법으로는 Solot이 제안한 PANORAMA가 있다. Solot의 해법과 본 논문의 해법의 특징에 대한 비교는 아래 [표 6]과 같다.

〈표 6〉 Solot의 해법과 본 논문 해법의 특징 비교

	Solot의 해법	본 논문의 해법
목적함수	생산율 최대화(팔렛비용고려)	생산율 최대화(평균산출시간고려)
중요제약식	평균산축시간 상한	전체팔렛수 상한
초기해	(1, … ,1)	휴리스틱알고리즘(이분할탐색)
해법	타부탐색 알고리즘	타부탐색 알고리즘

Solot의 해법은 문제의 모형화과정에서 팔렛의 한계비용(marginal cost)에 관한 상수와, 평균산출시간의 상한을 결정해야 한다. 그러나 이 값들을 결정하는 방안이 제시되어있지 않으며, 이 값들은 알고리즘의 수행도에 영향을 미치는 것으로 나타나있다([8]). 그러나 본 연구에서 제시한 해법은 모형화과정에서 직관적으로 구할 수 있는 전체팔렛수의 상한과 평균산출시간에 관한 상수를 결정하는 규칙을 포함하며, 앞 절의 〈표 4〉와 〈표 5〉에서 보듯이, 이 값들에 대해서 알고리즘이 영향을 받지 않는것으로 나타났다.

그리고 초기해를 최소할당에서 시작하는 Solot의 타부탐색 알고리즘과 본 논문의 알고리즘과의 비교가 아래 〈표 7〉에 나타나 있다. 〈표 7〉에서 본 논문의 알고리즘의 단계수는 초기화 알고리즘의 단계수를 반영하여 계산한 것으로서, 본 논문의 알고리즘이 더 적은 단계를 거쳐서 해를 찾으며, 해의 품질을 나타내는 알고리즘의 수행도에 있어서도 비교적 우월한 것으로 나타났다.

〈표 7〉 Solot의 타부탐색 알고리즘과 본 논문의 알고리즘의 수행도 비교

$\hat{u}$		1	2	3	4	5	6
Solot의 알고리즘	평균수행도	0.8932	0.9435	0.9520	0.9588	0.9633	0.9676
	최저수행도	0.5808	0.8133	0.8133	0.8133	0.8133	0.8133
	MVA 실행횟수	24	36	45	51	57	60
본 논문의 알고리즘	평균수행도	0.9624	0.9684	0.9706	0.9724	0.9733	0.9745
	최저수행도	0.8015	0.8445	0.8445	0.8445	0.8576	0.8576
	MVA 실행횟수	27	33	39	42	45	48

4. 결론 및 추후연구과제

본 논문에서는 최종제품을 구성하는 여러종류의 부품을 생산하는 FMS의 설계시 팔렛수를 결정하는 방안을 제시하고 있다. 해법은 FMS의 성능평가를 위해 Schweitzer의 MVA 알고리즘을 사용하고 있으며, 시뮬레이션과 비교실험결과 MVA 알고리즘은 팔렛수 결정문제의 해법에 적용가능한 것으로 나타났다. 본 논문에서 제시한 방안은 기존의 방법에 비해서 명확한 모형화 방법을 제시하고 있으며, 해법은 다음과 같은 특징을 가진다.

(1) 알고리즘의 수행도가 모형의 상수에 민감하지 않다.
(2) 짧은 수행시간으로 비교적 높은 평균수행도를 나타낸다.

해법의 최저수행도는 대체로 80%정도이며, 다양화된 해공간의 타부탐색방법([4][7])을 사용한다면, 최저수행도의 개선이 가능할 것이다. 가장 단순한 다양화 타부탐색기법은 종료기준이 만족되면, 임의의 새로운 초기해에서부터 다시 타부탐색을 계속하는 방법이며([7]), 현재까지 모든 해의 이동을 기억하는 타부리스트를 이용하여, 탐색되지 않은 해공간의 부분에서 새로운 초기해를 선택하는 복잡한 방법([4])이 연구되었다. 본 팔렛수 결정문제의 경우, 해의 표현이 간단하여 타부리스트의 크기가 큰 문제가 되지 않으므로, 후자의 방법이 적용가능하다. 추후에 이와 관련된 연구가 행해져야 할 것이다.

참 고 문 헌

[1] Dalliry, Y., Frein, Y., "An efficient method to determine the optimal configuration of a flexible manufacturing system," *Proceedings of the 2nd ORSA/TIMS conference on FMS*, 1986, pp. 269-282.

[2] Glover, F., "Tabu Search - Part I," *ORSA Journal on Computing*, Vol. 1, No. 3, 1989, pp. 109-206.

[3] Glover, F., "Tabu Search - Part II," *ORSA Journal on Computing*, Vol. 2, No. 1, 1990, pp. 4-32.

[4] Kelly, J. P., Laguna, M., and Glover, F., "A study of diversification strategies for the quadratic assignment problem," *Computers and Operations Research*, Vol. 21, No. 8, 1994, pp. 885-893.

[5] De Luca, A. and Perroto, G., "FMS simulation: advanced mathematical models and fast simulation algorithms for system optimization and performance evaluation," *Internal Report*, *EICAS Automazione S. p. A.*, Torino, Italy, 1986

[6] Seidmann, A., Oren, S. S., and Schweitzer, P. J. "An analytical review of several computerized closed queueing network models of FMS," *Proceedings of the 2nd ORSA/TIMS conference on FMS*, 1986, pp. 369-380

[7] Skorin-kapov, J., "Tabu search applied to the quadratic assignment problem,"

ORSA Journal of Computing, Vol. 2, No. 1, 1990, pp. 33-45.

[8] Solot, P., "A heuristic method to determine the number of pallets in a FMS with several pallet types," *International Journal of Flexible Manufacturing Systems*, Vol. 2, 1990, pp191-216

[9] Solot, P., Bastos, J. M., "Multi-Q: a queueing model for FMS with several pallet types," *Journal of Operations Reserach Society*, Vol. 39, No. 9, 1988, pp. 811-821

[10] Solberg, J. J., "A mathematical model of computerized manufacturing systems," *Proceedings of the 4th International Conference on Production Research*, Tokyo, Japan, 1977, pp. 1265-1275.

[11] Suri, R., Hildebrant, R. R., "Modelling flexible manufacturing systems using mean-value analysis," *Journal of Manufacturing Systems*, Vol. 3, No. 1, 1984.

[12] Templemeier, H., H. Kuhn, "Flexible manufacturing system: decision support for design and operation," *John Wiley*, 1993, p 102.

[13] Vinod, B., Solberg, J. J., "The optimal design of flexible manufacturing system," *Internal Journal of Production Research*, Vol. 23, No. 6, 1985, pp. 1141-1151.