

가공 순서 결정과 기계 선택을 위한 모형 개발

Model Development for Machining Process Sequencing and Machine Tool Selection

서 윤 호*

Seo, Yoonho

Abstract

Traditionally, machining process sequence was influenced and constrained by the design information obtained from CAD data base, i.e., class of operations, geometric shape, tooling, geometric tolerance, etc. However, even though all the constraints from design information are considered, there may exist more than one way to feasibly machine parts. This research is focused on the integrated problem of operations sequencing and machine tools selection in the presence of the product mix and their production volumes. With the transitional costs among machining operations, the operation sequencing problem can be formulated as a well-known Traveling Salesman Problem (TSP). The transitional cost between two operations is expressed as the sum of total machining time of the parts on a machine for the first operation and transportation time of the parts from the first machine to a machine for the second operation. Therefore, the operation sequencing problem formulated as TSP cannot be solved without transitional costs for all operation pairs.

When solved separately or serially, their mutual optima cannot be guaranteed. Machining operations sequencing and machine tool selection problems are two core problems in process planning for discretely machined parts. In this paper, the interrelated two problems are integrated and analyzed, zero-one integer programming model for the integrated problem is formulated, and the solution methods are developed using a Tabu Search technique.

* 울산대학교 산업공학과

1. 서론

현대 산업 사회는 다양한 소비자의 요구, 제품의 짧은 생명 주기, 제조 기술의 급속한 발전, 치열한 국내외 경쟁 등으로 특징 지워질 수 있다. 이러한 외부 환경의 급격한 변화에 대응하기 위하여 제조 시스템은 높은 유연성을 가져야 하며 이러한 유연 제조 시스템을 효율적으로 운용할 수 있는 방법이 동시에 설계 되어져야 한다. 수치 제어 장비(NC- 또는 Numerically Controlled machine)와 컴퓨터의 결합으로 제조 시스템의 하드웨어 유연성은 놀라운 정도로 발전 하였으나 이에 반하여 제조 시스템 운용의 유연성은 아직 자동화 장비의 다양하고 편리한 기능을 완전히 활용할 수 있는 수준 까지 도달하지는 못 하였다. 다양한 부품 조합(product mix)의 최적 생산을 위한 운용의 유연성을 얻기 위하여는 전통적으로 하나의 부품에 일정한 가공 순서와 방법을 고정 시키는 선형이며 정적인(linear and static) 공정 계획은 비적합하다. 유연 제조 시스템의 운용 유연성을 위하여는 부품을 가공하는 방법도 유연하게 설계되어야 하는 것이다.

공정 계획(process plan)이란 부품의 가공 방법, 가공 순서 등을 명시한 계획 문서로 공법 계획, operations sheet, 작업 계획 등 여러 명칭으로 불리운다. 전통적인 정적 공정 계획(static process plan)은 일반적으로 하나의 부품을 무한정 생산 한다는 가정 하에 설계되며, 공정 선택, 공정 순서 결정, 공작 기계의 선택, 최적 가공 조건의 결정, 가공 시간 추정, 공구 궤적 명시, NC 부품 프로그램 작성 등에 관한 정보를 포함하고 있다 [9]. 이

러한 전통적 공정 계획의 특징으로는 가공 방법(즉, 공작 기계 선택 과 방문 순서)이 고정 되어 있을 뿐만 아니라, 함께 가공 되어지는 부품 조합과 그들의 생산량 등 과 같은 생산 정보를 고려 하지 않는 다는 것이다.

그러므로 이러한 전통적 공정 계획은 묶음 생산을 근간으로 하는 현대의 생산 시스템의 효율 과 유연성을 저하 시키는 요인으로 지적 된다 [8]. Hancock [10, 11]에 의하여 간과 되었듯이 부품의 생산량이 변한 후에도 동일한 공적 계획이 반복적으로 이용된다면 더 이상 최적 절삭 조건(즉, 절삭 깊이, 절삭 속도, 이송 속도 등)을 제공하지 못할 뿐만 아니라, 부품 조합이 변할 경우 기계 간 부하의 불균형을 야기 시키게 될 것이다. 즉 생산 현장에서 동시에 가공 되는 구성 부품 과 각 묶음 생산량(lot size)이 결정 되기 전에 제품의 가공 방법을 고정 시킨다면 자동 제조 시스템의 운용 유연성을 활용하지 못하는 결과를 초래하기 때문이다.

이러한 전통적 공정 계획의 단점을 극복하기 위하여 CAD(Computer-Aided Design)로부터 설계 정보 뿐만 아니라 생산 현장에서 동적으로 변화하는 생산 정보를 적시에 반영할 수 있는 공정 계획 기법이 필요하다. 이를 위하여 유연 공정 계획(또는 동적 공정 계획) [8, 12], 다중 기계 고려 [10, 15], 다중 공정 계획 [16, 18, 19] 등의 방법들이 제안되어 있다.

본 연구는 그림 1에 굵은 선으로 표시되어 있듯이 부품 조합(part mix)과 각 생산량(lot size)의 다양한 변화에 대응하여 공작 기계와 가공 순서를 동시에 결정함으로써 자동 생산 시스템의 운용의 유연성을 높이기

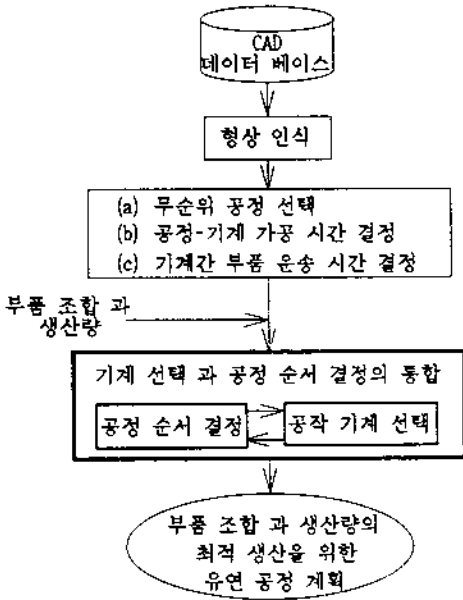


그림 1. 기계 선택과 공정 순서 결정 통합 문제의 범위

위한 방법을 제시하기 위함이다. 개별 부품 각각의 공정에 대하여 가장 효율적인 기계의 선택과 동시에 최단 거리의 가공 순서가 항상 부품 조합 전체를 가장 효율적으로 생산하지는 못 하는 것이다 [8]. 왜냐하면 몇몇 고성능 기계에만 편중된 선택은 작업량의 불균등 배분으로 인하여 병목 현상을 초래할 수 있을 뿐만 아니라, 순서가 정해지지 않은 공정에 대한 기계 선택은 부품 운송 시간을 고려할 수 없으므로 최적 생산을 보장할 수 없기 때문이다. 그러므로 부품 조합을 최적으로 가공할 수 있는 방법이란 개개 부품형의 최상의 가공 방법들 보다는, 각 부품형에 대해 최선의 방법은 아니더라도 전체 작업량이 각 기계에 균등하게 분포되어 전체 부품 조합을 가장 효율적으로 가공할 수 있는 방법들이어야 한다.

가공 공정 순서의 결정을 위하여 가공 기술적 요인을 일차적으로 고려 하여야 한다. 즉 가공 순서에 영향을 주는 가공 기술적 요인으로는 기하 형상, 공구의 종류, 기하 공차 (geometric tolerance) 또는 공정의 종류 (예, 황삭은 정삭 보다 선행 되어야 한다.) 등이 있으며 이에 관해 많은 연구가 진행되었다 [2, 17, 20, 21]. 본 연구에서는 가공 우선 순위에 영향을 끼치는 기술적인 방법들을 직접 고려하지 않았으며 이들은 입력으로 주어진다 가정하였다. 기술적으로 가능한 가공 순서들이 주어 진다면 다음 단계는 이들로부터 가장 경제적인 순서를 찾는 것이다. 본 연구는 기술적으로 가능한 가공 순서, 각 공정에 대한 각 기계에서의 가공 시간, 기계 간 부품 운송 시간이 주어진다는 가정 하에 동시에 가공되는 부품 조합과 그 생산량이 주어질 때 마다 전체 부품 조합을 가장 효율적으로 (즉 가공 시간과 운반 시간의 총합을 최소로 하는) 가공하기 위한 각 부품형에 대한 공정의 순서를 결정하며 동시에 공작 기계를 선택 하는 방법에 관하여 논한다.

각 부품형에 대해 최적의 공정 순서를 결정하기 위하여는 가능한 모든 공정 순서를 가공 시간과 운반 시간 합 의 척도로 평가할 수 있어야 하며 이를 위하여는 각 공정을 가공할 기계가 결정되어 있어야 한다. 왜냐하면 이는 각 공정 마다 가공할 기계가 정해져 있어야 가공 시간과 부품 운송 시간의 합을 구하는 것이 가능하기 때문이다. 즉 공정 순서의 평가를 위하여 각 공정에 기계가 할당되어 있어야 하기 때문에 기계 선택 문제는 공정 순서 결정에 선행되어야 한다. 반면에 공정 순서가 결정되기 전에 각 공정에 대하

여 공작 기계를 선택해야 한다면 기계 가공 시간과 기계간 부품 운송 시간을 동시에 고려할 수 없기 때문에 최적 기계의 선택은 어렵다. 그러므로 공정 순서 결정과 기계 선택 문제는 순차적으로 (즉, 어느 하나를 먼저 그 나머지를 후에) 접근해야 하는 별개의 문제라기 보다는 동시에 풀려야 하는 문제인 것이다.

본 연구의 목적은 가공 순서 결정 과 기계 선택 문제를 통합한 하나의 문제로 형식화하여 그 수학적 모형을 제시하고 이 문제를 위한 해법을 개발하는 것이다. 생산 현장에서 부품 조합이 바뀔 때마다 이 통합 문제는 풀려야 하기 때문에 해를 구하는 시간이 중요하며, 그러므로 최적해 보다는 근사해를 위한 발전적 기법의 개발이 타당하다. 이 문제의 발전적 알고리즘을 위하여 타부 탐색법을 응용하였다 [1, 5, 6, 7, 14].

다음 장에 이 공정 순서의 결정과 기계 선택을 위한 통합 문제를 기술하였으며, 제 3 장에는 수리 모형의 개발을 유도하였다. 제 4 장에는 해법 개발 과정이 설명되어 있으며, 그 다음에 간단한 예제를 풀어 알고리즘을 설명하였으며 결론으로 논문을 마친다.

2. 문제 기술

주문에 의한 묶음 생산 (batch production) 을 위주로 하며 공정 간 부품의 이동을 위하여 AGV (Automated Guided Vehicle)를 사용하는 자동 제조 시스템을 고려하자. 이 제조 시스템에서 부품 조합 과 그 생산량이 결정 될 때 마다 부품의 무순위 공정 집합으로부터 총 생산 시간을 최소로 하는 공정 순서

와 동시에 각 공정을 가공할 공작 기계를 선택하려 한다. 이때 각 부품의 가공을 위한 무순위 공정 집합, 각 공정을 서로 다른 기계에서의 가공할 때의 가공 시간과 기계 간 차량 운송 시간은 주어져 있다고 가정한다.

부품이 생산에 참여한 시간은 크게 가공 시간, 가공 대기 시간, 운반 시간, 운반 대기 시간으로 나뉘어 진다 [18]. 이 네 가지 구성 시간 중에서 가공 대기 시간은 부품의 스케줄링에 의하여 또한 운반 대기 시간은 운반 차량 즉 AGV (Automated Guided Vehicle) 의 스케줄링에 의하여 생산 시간 중에 결정되어 진다. 즉 이들 대기 시간은 스케줄링 의사 결정에 따라 그 양이 정해지는 것으로 공정 순서와 기계 선택을 결정하기 위한 본 연구의 범위를 벗어 나기 때문에 본 논문에서 가공 대기 시간과 운반 대기 시간을 각각 "0"으로 간주 한다. 본 논문에 사용된 생산 시간이란 가공 시간 과 운반 시간의 합으로 정의한다. 그러므로 총 생산 시간이란 생산에 참여한 전체 부품이 각 기계에서 가공에 소비한 시간의 합 과 기계 간에 이동을 위하여 소비한 운반 시간의 총합을 의미한다.

AGV에 의하여 한번에 운반되는 부품을 "적재 단위 (unit load)"라 하며 적재 단위에 포함된 부품의 개수를 "적재 단위 크기"라 한다. 적재 단위 크기는 물류 시스템을 최적으로 설계하기 위하여 고려되어 져야 하는 의사 결정 변수 중에 하나 이나 본 논문에서는 그 값이 주어졌다고 가정한다 [4].

3. 수리 모형

문제를 수리적으로 기술하고 그 모형을 개

발하기 위하여 기호와 의사 결정 변수를 다음과 같이 정의한다:

I 부품 조합. 즉, I 개의 상이한 부품형의 집합, 즉, $I = \{1, 2, \dots, i, \dots, I\}$.

q 부품 조합 내 부품의 생산 요구량

$$q = \{q_1, q_2, \dots, q_i, \dots, q_I\}.$$

G_i 부품 i 의 무순위 공정 집합.

$$G_i = \{g_{ij} | j=1, 2, \dots, J_i\} \quad \forall i=1, 2, \dots, I.$$

g_{ij} 부품 i 의 j 번째 공정. 단 이것은 가공 순서와는 무관하다.

J_i 부품 i 의 총 공정 개수, $J_i = |G_i|$.

M 공작 기계 집합, $M = \{1, 2, \dots, m, \dots, M\}$.

p_{ijm} 부품 i 의 j 공정을 m 기계에서 가공할 경우 가공 시간.

τ_{lmn} 공작 기계 m 에서 n 으로의 AGV의 운송 시간.

u_{ijl} 부품 i 의 j 번째 공정에서 l 번째 공정으로 이동 시 적재 단위의 크기, 즉, 적재 단위 내의 부품의 수.

\bar{W}_m 기계 m 의 이용 가능한 총 기계 시간.

$$y_{ijl} = \begin{cases} 1, & \text{부품 } i \text{가 } j \text{공정을 수행한 즉시} \\ & l \text{공정을 수행하는 경우,} \\ 0, & \text{그렇지 않은 경우.} \end{cases}$$

$$x_{ijm} = \begin{cases} 1, & \text{부품 } i \text{의 } j \text{공정을 수행하기} \\ & \text{위하여 } m \text{기계를 선택한 경우,} \\ 0, & \text{그렇지 않은 경우.} \end{cases}$$

무순위 공정 집합으로부터 공정 순서를 결정하기 위한 문제를 살펴 보자. 만약 어떤 부품의 무순위 공정 집합 내의 한 공정에서 다른 공정으로 이동할 때 발생하는 이동 비용을 알 수 있다면 전체 이동 비용을 최소화하는 공정 순서가 존재할 것이며, 이러한 유형

의 문제들은 TSP (Traveling Salesman Problem: 이동 판매원 문제)로 형식화 될 수 있다 [3, 13]. 즉 모든 부품형 $i=1, 2, \dots, I$ 의 모든 공정 짝 $j, l=1, 2, \dots, J_i$ 에 대하여 부품 공정 간 이동 비용 c_{ijl} 가 주어졌다면 전체 부품의 총 공정 이동 비용을 최소화 하는 각 부품의 공정 순서를 찾는 문제인 것이다.

부품 공정 간 이동 비용 c_{ijl} 는 부품 i 의 전체 생산 요구량 q_i 을 j 공정을 가공하는 기계에서 가공할 때 총 기계 가공 시간 과 l 공정을 가공하는 기계로의 이동 시간의 합으로 정의 될 수 있으며, 최소화 되어야 할 목적 함수 (전체 부품의 총 공정 이동 비용)는 다음 식(1)과 같이 표현 된다.

$$\sum_{i=1}^I \sum_{j=1}^{J_i} \sum_{\substack{l=1 \\ l \neq j}}^{J_i} c_{ijl} y_{ijl} \quad (1)$$

정의에 따라 각 공정을 수행 하기 위한 기계 선택이 선행 되어야 부품 공정 간 이동 비용을 계산할 수 있다. 부품형 i 의 j 공정에서 l 공정으로의 총 운송 시간을 구하기 위하여는 q_i 개의 부품 i 를 j 공정에서부터 l 공정까지 운송하기 위한 운반 횟수 v_{ijl} 를 알아야 한다. 적재 단위의 크기 v_{ijl} 이 주어 졌다면 n_{ijl} 는 다음 식(2)에 의하여 구할 수 있다. 단 $[x]^+$ 는 x 보다 크거나 같은 최소 정수를 의미한다.

$$v_{ijl} = \left(\frac{q_i}{u_{ijl}} \right)^+ \quad (2)$$

그러면 위 정의에 따라 부품 공정 간 이동 비용 c_{ijl} 는 다음 식(3)과 같이 표현 된다:

$$c_{ijl} = \sum_{m=1}^M \sum_{n=1}^M \{q_i p_{ijm} x_{ijm} + v_{ijl} \tau_{lmn} x_{ijm} x_{iln}\} \quad (3)$$

식 (1)과 (3)은 공정 순서 결정 과 공작 기계 선택 통합 문제의 목적 함수 이며, 의사 결정 변수의 비선형 결합으로 미루어 보아 공정 순서 결정 과 공작 기계 선택은 동시에 고려 되어야 하는 문제임을 알 수 있다.

이 두 문제를 풀기 위하여 목적 함수의 값이 최소화 되기만 한다면 각 공정을 수행하기 위한 가장 효율적인 기계들이 선택 될 것이고, 이는 몇몇 고성능 기계에 부하가 편중되는 결과를 초래할 수 있으며 또한 때때로 시스템 병목 현상을 유발할 수도 있다. 몇몇 고성능 기계에 걸린 과부하는 피가공품의 생산 대기 시간을 증가 시키게 되며 결국 생산 시간을 연장시키게 될 것이다. 그러므로 공작 기계 선정 시 부하를 적절히 배분하여 부하 편중 현상을 예방하는 것이 중요하다. 즉 각 공작 기계에 할당된 총 가공 시간의 합이 그 기계의 총 이용 가능한 시간 \bar{W}_m 보다 작거나 같아야 한다는 제약식을 부가함으로 이 목적을 달성할 수 있다. 뿐만 아니라 각 기계의 이용 가능한 총 시간 \bar{W}_m 을 적절히 변경 시킴으로 각 기계에 할당되는 부하를 임의로 조절할 수도 있다.

통합 공정 순서 결정과 기계 선택 문제의 정수 계획법을 이용한 수리 모형이 다음에 기술되어 있다.

Minimize

$$\sum_{i=1}^I \sum_{j=1}^{J_i} \sum_{l=1}^{J_i} \sum_{m=1}^M \sum_{n=1}^M (q_{ij} \phi_{ijm} x_{ijm} + v_{ijl} \tau_{mn} x_{ijm} x_{ijn}) y_{ijl} \quad (4)$$

subject to

$$\sum_{i=1}^I \sum_{j=1}^{J_i} q_{ij} \phi_{ijm} x_{ijm} \leq \bar{W}_m \quad \forall m \in M \quad (5)$$

$$\sum_{m=1}^M x_{ijm} = 1 \quad \forall i \in I \text{ and } j \in G_i \quad (6)$$

$$\sum_{j \in S_i} \sum_{l \in S_i} y_{ijl} \leq |S_i| - 1 \quad \forall S_i \subset G_i, 2 \leq |S_i| \leq J_i - 2 \text{ and } i \in I, \quad (7)$$

$$\sum_{j=1}^{J_i} y_{ijl} = \sum_{j=1}^{J_i} y_{ijj} = 1, \quad \forall i \in I \text{ and } l \in G_i \quad (8)$$

$$x_{ijm} \in \{0, 1\} \quad \forall i = I, j \in G_i \text{ and } m \in M, \quad (9)$$

$$y_{ijl} \in \{0, 1\} \quad \forall i \in I \text{ and } j, l \in G_i \quad (10)$$

제약식(5)는 각 기계에 걸리는 총 부하량이 그 기계의 최대 이용 가능한 시간을 초과하지 못하게 함으로써 부하량을 조절한다. 제약식(6)은 하나의 공정을 수행하기 위하여 하나의 기계가 선택 되도록 한다. 제약식(7)은 subtour elimination constraints로 각 부품의 공정이 모두 연결되어 순서가 결정될 수 있게 한다 [3]. 제약식 (8)은 각 부품의 공정들이 오직 한번씩 방문 되도록 한다. 식(9)와 식(10)은 의사 결정 변수는 이진 정수가 됨을 보장한다.

위 수리 모델은 공정 순서를 결정하기 위하여 각 부품 당 한 번의 TSP를 풀어야 하며 (즉 의사결정 변수 y_{ijl} 의 결정) 동시에 각 TSP를 풀기 위하여는 다시 기계 선택 문제를 풀어야 하는 (즉 의사결정 변수 x_{ijm} 의 결정) mixed nonlinear integer programming 모델이다. 생산 현장에서 부품 조합과 그 생산량이 변할 때 마다 위 모델을 이용하여 최적해를 구하는 것은 모델의 복잡성으로 미루어 보아 어렵다.

4. 해법 개발

위 모형 (또는 문제)의 구조는 도시 사이의 거리 (또는 비용)가 또 다른 최적화 문제 (즉 기계 선택 문제)의 해로 주어지는 여러 개의 TSP (즉 공정 순서 결정 문제)가 결합된 형태이다. 또한 제약식 (5)에 의해 주어진 균등 부하 제약이 있기 때문에 이 문제에서는 각 부품형에 대한 최적 공정 순서가 항상 전체 부품 조합에 대한 최적해가 되는 것은 아니다. 이러한 구조상의 특징과 타부 탐색 기법(Tabu Search Technique)을 결합하여 본 문제의 해법을 개발 한다.

Glover에 의하여 제안된 타부 탐색법은 많은 combinatorial optimization 문제에 성공적으로 적용되어 그 효율이 증명되었다 [5, 6, 7, 14]. 특히 Malek *et al.* [14]이 타부 탐색법을 TSP에 적용한 방법에 따르면 타부 탐색법은 임의의 초기 가능해로 시작하며, 현재해로부터 유도 가능한 모든 인근해를 구하고 이들 중 최상 인근해를 찾아 가는 과정을 정지 조건이 만족될 때 까지 계속 반복 하며, 더 나은 해를 찾아 가는 반복 개선 절차이다. 타부 리스트는 최근 몇 번의 반복에서 발견되어진 최상 인근해를 기록하고 있는 자료 저장소로, 현재해가 타부 리스트에 포함되어 있으면 고려 대상에서 제외 하는 역할을 하기 때문에 사이클링을 막고 지역해에서 빠져 나올 수도 있게 하는 장치인 것이다. 그러나 현재해가 타부 리스트에 속해 있더라도 타부 리스트에 속할 때 보다 궁극적으로 더 좋은 해를 유도할 가능성이 있다면, 이 현재해를 다시 고려 대상에 포함시키는 방법도 제시하고 있다. 더 좋은 해를 유도할 가능성이 있

는 판단은 희망 수준 (aspiration level)을 유지함으로 가능하다. 즉, 과거의 타부해가 타부 리스트에 속할 때의 어떤 속성 (예 비용)을 그 타부해의 희망 수준으로 기억시켰다가 타부 리스트에 속한 현재해의 속성이 타부 리스트에 속할 때의 희망 수준 보다 좋다면 (속성의 종류에 따라 높거나 낮을 수 있다), 현재해는 더 좋은 해를 유도할 가능성이 있다고 판단한다. 타부 탐색법은 여러 논문을 통하여 이미 소개 되었기 때문에 더 이상의 일반적 내용 설명은 생략 한다[1, 5, 6, 7, 14].

4.1. 공정 순서 결정과 기계 선택을 위한 알고리즘

본 알고리즘의 목적은 부품 조합 I , 생산 요구량 q , 와 각 부품의 무순위 공정 집합 G_i 가 주어졌을 때 총 생산 시간을 최소화 하는 최상해를 구하는 것이다. 해란 각 부품에 대한 공정 순서와 각 공정을 수행하기 위하여 선택된 공장 기계를 말하며, 선택된 기계와 순서에 따라 주어진 부품 조합의 생산을 위해 소요되는 총 기계 가공 시간과 운송 시간의 총 합은 그 해의 비용 또는 목적 함수 값이다.

다음에 알고리즘을 기술하기 위하여 필요한 기호들을 정의 하였다:

IFS_i 부품 i 의 공정 순서와 선택된 기계를 포함하는 초기 가능해.

π_i 부품 i 의 현재해.

$\pi_i(j, l)$ 부품 i 의 현재해로부터 공정 짝 (j, l)을 교환한 인근해.

π_i^* 부품 i 의 최상 인근해.

L_i 부품 i 의 최상 인근해 비용.

Π_i 부품 i 의 최상해.

B_i 부품 i 의 최상해 비용.

w_i 현재해에서 부품 i 의 해가 가능해가 되기 위한 가용 기계 시간.

T_i 부품 i 의 타부 리스트이며 그 원소는 공정 짝 (j, l) 이다.

$A(j, l)$ 공정 짝 (j, l) 의 희망 수준 (aspiration level).

$N(i, j)$ 부품 i 의 현재해에서 j 공정에 선택된 기계.

$$\eta_{ijm} = \begin{cases} 1, & \text{부품 } i \text{의 현재해에서 } j \text{공정을} \\ & \text{위하여 기계 } m \text{이 선택된 경우,} \\ 0, & \text{그렇지 않은 경우.} \end{cases}$$

일반적인 타부 탐색법을 본 문제 - 공정 순서 결정과 기계 선택 통합 문제 - 의 특성에 맞게 적용시킨 알고리즘이 개발 되었으며 그 개략적 흐름이 그림 2 와 3에 도시되어 있으며, 다음과 같은 특징으로 요약될 수 있다. 첫째 단계 4에서 현재해로부터 유도 가능한 모든 인근 공정 순서를 구한 후 단계 5.1에서 그 각각의 공정 순서에 대하여 기계 선택 문제를 풀기 때문에 기계 가공 시간과 부품 운송 시간 모두를 고려할 수 있다. 또한 여러 번의 공정 순서 결정 문제 (TSP)를 동시에 풀기 위하여 매 반복(단계 3~8 반복) 마다 모든 부품형에 대하여 다시 반복(단계 3~9 반복)하는 구조를 갖는 이중 반복 구조를 갖는다. 마지막으로 알고리즘이 초기 가능해로 시작하여 매 반복 마다 구해지는 각 부품형에 대한 현재해는 부하 균등 제약, 즉 수리 모형의 제약식(5)를 만족하는 가능해 이어야 한다. 그러므로 매 반복 마다 각각의 부품에 대하여 기계 선택 문제를 풀 때

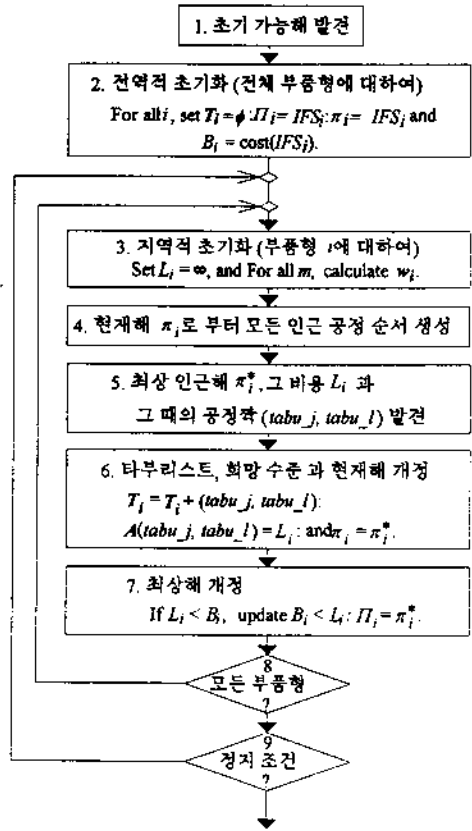


그림 2. 부품 조합의 공정 순서 결정을 위한 이중 반복 구조

현재해가 가능해가 되도록 유지하기 위하여 현재 부품형 i 에 대하여 남아 있는 기계 가용 시간 w_i 을 구하여 이 제약 하에 기계 선택 문제를 푼다.

본 알고리즘은 각 부품에 대한 공정 순서를 결정하기 위한 이중 반복 구조로 이루어져 있으며 (참조 그림 2) 매 반복마다 현재해의 공정 순서로부터 유도 가능한 모든 인근 공정 순서를 생성하고 (4.4절 참조), 이중 최상 인근해를 구하고 (그림 3 참조), 최상 인근해 중에서 가장 좋은 해를 최상해로

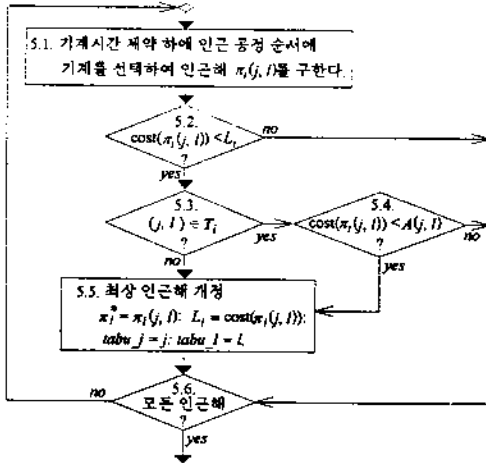


그림 3. 최상 인근해를 구하기 위한 반복 루프

기록하는 과정 (그림 2의 단계 7 참조)을 반복한다. 그림 3에 도시 되어 있듯이 최상 인근해를 구하는 과정은 현재해의 공정 순서로부터 유도 가능한 모든 인근 공정 순서 중에서 기계 가용 시간 제약 하에 기계 선택을 한 후 최상 인근해를 찾는 반복 구조이다. 최상 인근해란 타부 리스트 T_i 에 포함되어 있지 않거나 또는 타부 리스트에 포함되어 있다면 그 비용이 희망 수준 보다 작은 해 중에서 최소 비용을 갖는 인근해를 가리킨다.

4.2. 초기 가능해 (단계 1)

부품 조합에 대한 초기 가능해는 임의 반복에 기초한 탐색법에 의하여 구한다. 매 반복 마다 전체 부품에 대한 임의의 공정 순서를 구하고 각 공정에 가장 효율적인 기계를 선택한다. 구해진 임의의 초기해가 가능해 인지를 검사하기 위하여 선택된 기계 시간의 총합이 이용 가능한 총시간을 초과 하는 기계 집합 \bar{M} 과 미달 하는 기계 집합 \underline{M} 을 다음식 (11)과 (12)에 의하여 각각 구한다.

$$\bar{M} = \left\{ m \mid \sum_{i=1}^I \sum_{j=1}^{J_i} q_i \phi_{ijN(i,j)} \eta_{ijm} > \bar{W}_m \right\} \quad (11)$$

$$\underline{M} = \left\{ m \mid \sum_{i=1}^I \sum_{j=1}^{J_i} q_i \phi_{ijN(i,j)} \eta_{ijm} \leq W_m \right\} \quad (12)$$

\bar{M} 에 속하는 기계가 발견되지 않는다면 (즉 $\bar{M} = \emptyset$) 현재 초기해는 가능해임을 의미한다. 가능해는 아니나 가능해가 될 가능성이 있다면 (즉 $\bar{M} \neq \emptyset$ 이며 $\underline{M} \neq \emptyset$), 기계 시간 초과량이 가장 큰 기계를 이용하는 임의의 부품과 공정을 선택하여, 그때의 가공 시간과의 차가 가장 적은 다른 기계와 대체한다. 이를 위하여 다음 식 (13) (14) 와 같이 초과 기계 군에 대한 기계 시간 초과량 $\bar{Q}(m)$ 과 미달 기계 군의 기계 시간 미달량 $\underline{Q}(m)$ 을 유지한다.

$$\bar{Q}(m) = \sum_{i=1}^I \sum_{j=1}^{J_i} q_i \phi_{ijN(i,j)} \eta_{ijm} - \bar{W}_m, \quad \forall m \in \bar{M} \quad (13)$$

$$\underline{Q}(m) = \bar{W}_m - \sum_{i=1}^I \sum_{j=1}^{J_i} q_i \phi_{ijN(i,j)} \eta_{ijm}, \quad \forall m \in \underline{M} \quad (14)$$

이와 같은 과정을 임의의 번 반복 (outer_max, 와 inner_max)하는 초기 가능해 구하는 알고리즘이 다음에 자세히 기술되어 있다.

0. Initialize: input outer_max and inner_max. Set outer_iter = 0.
1. Set inner_iter = 0.
For all $i \in I$, generate a random initial sequence IS_i of operations in G_i .
For $j \in J_i$, assign machine: $N(i, j) = m$ such that $\min\{\phi_{ijm} \mid \forall m\}$.
2. (a) For the current sequence and machines selected, find \bar{M} , \underline{M} , \bar{Q} and \underline{Q} .
(b) If $\underline{M} = \emptyset$, stop: No initial feasible

solutions are found.

3. (a) If $\bar{M} = \emptyset$, stop: Initial feasible solutions are found, $IFS_i = IS_i, \forall i$.
 (b) If $\bar{M} \neq \emptyset$ and $\underline{M} = \emptyset$, go to Step 7.
 (c) If $\bar{M} \neq \emptyset$ and $\underline{M} \neq \emptyset$, go to Step 4.
4. For \tilde{m} such that $\bar{Q}(\tilde{m}) = \max\{\bar{Q}(m) \mid \forall m \in \bar{M}\}$,
 (a) Randomly select i, j such that $N(i, j) = \tilde{m}$.
 (b) Find m' such that $\min\{\phi_{ijm'} - \phi_{ijm} \mid \forall m' \in \underline{M} \text{ and } (\phi_{ijm'} - \phi_{ijm}) \geq 0\}$.
5. If no such m' exists, go to Step 6.
 Otherwise, (a) Replace \tilde{m} with m' for j operation of i part: $N(i, j) = m'$.
 (b) Update: $\bar{Q}(\tilde{m}) = \bar{Q}(\tilde{m}) - q_i \phi_{ij\tilde{m}}$:
 $\underline{Q}(m') = \underline{Q}(m') - q_i \phi_{ijm'}$.
 (c) If $\bar{Q}(\tilde{m}) \leq 0$, update $\bar{M} = \bar{M} - \tilde{m}$.
 (d) If $\underline{Q}(m') \leq 0$, update $\underline{M} = \underline{M} - m'$.
 (e) Return to Step 3.
6. If ($inner_iter < inner_max$), set $inner_iter = inner_iter + 1$: return to Step 4.
 Otherwise, go to Step 7.
7. If ($outer_iter < outer_max$), set $outer_iter = outer_iter + 1$: return to Step 1.
 Otherwise, stop. No initial feasible solutions are found.

4.3. 지역적 초기화 (단계 3)

지역적 초기화는 매 반복으로 인한 부품형의 변화 시 요구된다. 앞서 설명하였 듯이 알고리즘의 매 반복 마다 구해지는 현재해가 가능해가 되게 하기 위하여 현재 부품형 i 에 대하여 남아 있는 가용 기계 시간 $w_i = [w_{i1}, w_{i2}, \dots, w_{im}, \dots, w_{iM}]$ 을 구하고 이 가용 기계 시간 제약 하에 기계 선택 문제를 푸는

것은 본 알고리즘의 중요한 특징이다. 전체 기계 가용 총 시간 \bar{W}_m 중 부품 i 에 허용된 가용 기계 시간 w_{im} 은 다음 식 (15)에 의하여 구할 수 있다:

$$w_{im} = \bar{W}_m - \sum_{k=1}^I \sum_{j=1}^{J_k} q_k \phi_{kjN(k,j)} \eta_{kjm} \quad \forall m \quad (15)$$

4.4. 인근 공정 순서 생성 (단계 4)

현재해의 공정 순서로부터 가능한 모든 공정 짝 (j, l) 을 교환하여 인근 공정 순서를 생성한다. 부품 i 의 현재해 π_i 의 현재 공정 순서로부터 공정 짝 (j, l) 을 교환하여 생성된 인근 공정 순서의 각 공정에 기계를 선택한 인근해를 $\pi_i(j, l)$ 라 한다. 현재해 π_i 로부터 유도 가능한 인근 공정 순서의 총 개수는 부품 i 의 총 공정 수 J_i 개로부터 2개를 선택하는 방법의 수 $\frac{1}{2}(J_i^2 - J_i)$ 와 같다.

4.5. 제한된 부하량을 가진 기계 선택 (단계 5.1)

부품형 i 와 그 공정 순서가 주어 졌을 때 그 부품의 총 생산 시간 (즉 기계 가공 시간과 부품 운송 시간의 총합)이 최소화 되도록 각 공정에 기계를 선택하는 문제이다. 현재해가 항상 가능해가 되기 위하여 현재 부품형 i 의 가용 기계 시간 w_i 제약 하에서 문제를 푼다. 이 문제의 해법을 개발 하기 위하여 branch and bound 기법을 기초로 한 depth first search 알고리즘을 이용하였다. 설명을 위하여 필요한 기호를 정의 하였다.

S_j^i : j th 공정 까지 기계를 선택하고 $j+1$ 부터 J_i 까지 공정에 기계 선택을 남겨 둔 문제로, 현재해 σ 는 $1 \sim j$ th 공정까지 선택된 기계 집합이다.

LB_σ^j 문제 S_σ^j 의 lower bound.

UB 전역적 upper bound.

σ 현재해

σ^* 현재 최상해

문제 S_σ^j 의 lower bound란 현재해 σ 가 가질 수있는 최선의 값을 의미하며, 이는 j th 공정까지는 선택된 기계의 총 가공 시간과 운송 시간의 합과 기계를 선택하지 못한 공정들에 대하여는 그 최소 생산 시간의 합으로 표현되며 다음식 (16)에 의하여 정의 된다:

$$LB_\sigma^j = \sum_{j_i, l=1}^j (q_i \phi_{iN(i,l)} + v_{i+1} r_{N(i,l)N(i,l+1)}) + \sum_{l=j+1} \min\{q_i \phi_{ilm} + v_{i+1} r_{mn} \mid \forall m, n\} \quad (16)$$

push와 pop 연산을 위한 자료 구조 스택을 이용 하였으며 전체 알고리즘은 다음에 상세히 기술 되어 있다.

0. Initialize: Set $UB = \infty$; $j = 0$; $\sigma = \phi$, and Input w_i .

Push the original problem onto the stack.

1. (a) If the stack is empty and $UB = \infty$, stop. No solution is found.

(b) If the stack is empty and $UB < \infty$, stop. The solution σ^* with cost UB is found.

(c) Otherwise, pop a subproblem S_σ^j from the stack, and go to Step 2.

2. Lower Bounding:

(a) If $LB_\sigma^j \geq UB$, fathom subproblem S_σ^j and return to Step 1.

(b) If $LB_\sigma^j < UB$ and $j = J_i$, go to Step 4.

(c) If $LB_\sigma^j < UB$ and $j = J_i$, go to Step 3.

3. Upper Bounding:

(a) If $LB_\sigma^j < UB$, set $UB = LB_\sigma^j$: the

current best solution $\sigma^* = \sigma$.

(b) Return to Step 1.

4. Branching and Feasibility Test:

(a) For all $m \in M$, if $\sum_{j=1}^{J_i} q_i \phi_{ijm} \leq w_{im}$

(feasible), generate subproblems $S_{\sigma+m}^{j+1}$.

(b) Push subproblems into stack, according to a non-decreasing order of lower bound.

(c) Return to Step 1.

4.6. 타부 리스트와 희망 수준 개정

(단계 6)

타부 리스트는 타부 크기 (tabu size) 만큼의 최근 선택된 최상 인근해 들을 기록 유지한다. 실제 기록 되는 타부 리스트의 원소는 최상 인근해가 되기 위하여 그 이전 해로부터 자리 바꿈 한 공정 짝 (j, l) 이므로 $T_i = T_i + (j, l)$ 와 같이 표기된다. 타부 크기는 타부 탐색법의 효율에 영향을 끼치는 중요한 파라메터가 된다.

또한 최상 인근해 π_i^* 가 타부 리스트에 속하게 될 때 실제로는 공정 짝 (j, l) 이 기록되는 것이므로 그때의 희망 수준은 그 최상 인근해 π_i^* 의 비용 L_i 가 되며, 공정 짝 (j, l) 의 희망 수준은 $A(j, l)$ 로 참조되어 질 수 있다.

4.7. 정지 조건

알고리즘은 정지 조건을 만족 시킬 때 까지 반복된다. 정지 조건은 최대 반복 회수이거나 최대 허용 시간, 또는 현재 최상해가 발견된 후부터의 허용 시간 등이 가능하다 [5, 6]. 본 문제에는 최대 반복 회수 보다 큰 경

우를 정지 조건으로 하였다.

표 2. 기계 간 운송 시간

	1	2	3	4	5
1	31	7	27	26	17
2	19	19	15	19	10
3	5	17	37	36	27
4	8	20	4	39	30
5	18	18	14	5	28

5. 알고리즘 적용 예제

어느 공장에서 생산되는 여러 부품 중 부품 #1, #2, #3의 무순위 공정 집합이 $G_1 = \{g_{11}, g_{12}, g_{13}\}$, $G_2 = \{g_{21}, g_{22}, g_{23}, g_{24}, g_{25}\}$, $G_3 = \{g_{31}, g_{32}, g_{33}, g_{34}\}$ 와 같고, 이들을 가공하기 위하여 다섯 대의 머시닝 센터가 있으며, 각 부품의 공정을 각각의 머시닝 센터에서 가공할 때 소요되는 가공 시간과 머시닝 센터 간의 운송 시간은 표 1과 표 2에 각각 주어져 있다. 이와 같은 생산 시스템에서 부품 조합 $I = \{1, 2, 3\}$ 과 생산 요구량 $q = \{40, 70, 60\}$ 를 효율적으로 생산하기 위한 공정 순서와 각 공정을 가공하기 위한 기계를 선택하기 위하여 앞에 개발된 해법을 적용한다. 이때 적재 단위의 크기는 모든 부품형과 공정에 대하여 10으로 가정한다 $u_{ijt} = 10$.

표 1. 공정-기계 가공 시간

	g_{11}	g_{12}	g_{13}	g_{21}	g_{22}	g_{23}	g_{24}	g_{25}	g_{31}	g_{32}	g_{33}	g_{34}	\bar{W}_m	
$[p_{im}] =$	1	7	7	10	∞	3	8	12	4	∞	10	6	15	800
	2	4	3	6	9	5	7	5	3	14	9	5	∞	800
	3	12	9	5	4	6	12	∞	9	5	∞	9	7	800
	4	5	6	2	10	∞	8	6	6	2	10	4	8	800
	5	4	∞	8	12	6	6	10	14	8	7	12	6	800

5.1. 초기 가능해 발견

최대 이용 가능한 기계 시간은 $\bar{W}_m = 800, \forall m = 1, 2, \dots, 5$, 인 것으로 가정하여, 8 번의 반복 후에 다음과 같은 초기 가능해가 발견되었다:

$$IFS_1 = g_{11}(2) \rightarrow g_{13}(4) \rightarrow g_{12}(4),$$

$$[\tau_{mn}] =$$

$$IFS_2 = g_{24}(2) \rightarrow g_{23}(1) \rightarrow g_{21}(3) \rightarrow g_{22}(1) \rightarrow g_{25}(2),$$

$$IFS_3 = g_{33}(4) \rightarrow g_{32}(5) \rightarrow g_{31}(4) \rightarrow g_{34}(5).$$

이때의 총 기계 가공 시간의 합은 각 기계에 대하여 $W_1 = 770$ 분, $W_2 = 720$ 분, $W_3 = 280$ 분, $W_4 = 680$ 분, $W_5 = 780$ 분이며, 총 생산 시간 (총 기계 가공 시간 + 운송 시간)은 4258분이다.

5.2. 공정 순서 결정 과 기계 선택

가공 순서 결정 알고리즘을 이용하기 위하여는 정지 조건 (최대 반복 회수) 과 타부 크기를 결정 하여야 한다. 위 문제를 풀기 위한 정지 조건으로 최대 30 회 반복 과 타부 리스트의 크기를 3으로 하였다. 이 값을 알고리즘에 적용 시킨 결과 위의 초기 가능해로부터 다음과 같은 발견해를 얻었다.

$$\Pi_1 = g_{12}(2) \rightarrow g_{11}(5) \rightarrow g_{13}(4), B_1 = 420 \text{ 분},$$

$$\Pi_2 = g_{23}(1) \rightarrow g_{24}(2) \rightarrow g_{21}(3) \rightarrow g_{22}(1) \rightarrow g_{25}(2), B_2 = 1848 \text{ 분},$$

$$\Pi_3 = g_{31}(4) \rightarrow g_{32}(5) \rightarrow g_{33}(4) \rightarrow g_{34}(3), B_3 = 1343 \text{ 분}.$$

이때의 총 생산 시간은 3702 분으로 초기 가능해 보다 556분 (13 %) 줄었으며 총 기계 가공 시간의 합은 각 기계에 대하여 $W_1 =$

770분, $W_2=680$ 분, $W_3=700$ 분, $W_4=440$ 분, $W_5=580$ 분으로 최대 이용 가능 시간을 초과하지 않았다. 뿐만 아니라 초기 가능해 보다 각 기계 당 총 기계 가공 시간이 좀더 균등히 배분 되어 있음을 알 수 있다.

6. 결론

제조 시스템의 하드웨어 유연성은 컴퓨터 기술의 발달과 함께 획기적으로 발전한 반면, 자동 제조 시스템을 유연하게 운용하고자 하는 노력은 상대적으로 적었던 것이 사실이다. 자동 제조 시스템의 운용 유연성이란 부품 조합과 그 생산량이 계속해서 변화하는 혼류 생산 (production mix)을 얼마나 효율적으로 수행할 수 있는가의 척도이며, 전통적인 정적 공정 계획으로는 운영 유연성을 얻기에 적합하지 않다.

본 논문은 부품 조합과 생산량, 부품의 무순위 공정 집합이 주어 졌을 경우 공정 순서의 결정과 기계 선택을 동시에 고려하여 최적 가공 방법을 선택하는 “유연 공정 계획 방식”에 관하여 다루고 있다. 특히, 가공 순서의 평가를 위하여 총 가공 시간 뿐만 아니라 기계 간 부품 운반 시간을 함께 고려한 점은 공정 계획 과 물류 시스템 설계의 통합을 위한 핵심이라 생각된다. 본 연구에서는 공정 순서 결정과 기계 선택의 통합된 문제의 수리 모델의 개발뿐만 아니라, 문제 구조의 분석을 통하여 발견적 해법을 개발하였다. 해법은 타부 탐색법을 근간으로 하였으며, 초기 가능해 구하는 방법 과 부품의 공정 순서를 평가 하기 위하여 또 다른 최적화 문제인 기계 선택 문제의 해법이 소개 되었다.

그러나 서론에 언급하였듯이 기술적 제약에 따른 가공 순서의 우선 순위 결정 문제 또는 비용 함수에 피가공품의 품질을 고려하는 문제 등은 본 논문에서 명시적으로 다루지 않았으며 차후 연구 과제로 남겨 놓는다.

참 고 문 헌

- [1] Bland, J. A. and Dawson, G. P., 1991, “Tabu search and design optimization”, Computer-Aided Design, Vol. 23, No. 3, pp 195-201.
- [2] 조 규갑 (Cho, K. K.), 김 인호 (I. H. Kim), 노 형민 (H. M. Rho), 1989, “자동 공정 설계 에서 가공 작업의 선정 및 순서 결정 기법의 개발”, 대한 산업 공학 회지, Vol. 15, No. 2, pp 45-55.
- [3] Dantzig, G. B., Fulkerson, D.R., and Johnson, S.M., 1954, “Solution of a large-scale traveling salesman problem”, Operations Research, Vol. 2, pp 393-410.
- [4] Egbelu, P. J., 1987, “Integration of unit load selection and in the design of an AGV-based automated manufacturing shop,” Proceedings of 9th International Conference on Production Research, Vol. 2, pp 2159-2166, Cincinnati, Ohio.
- [5] Glover, F., 1989, “Tabu search - Part I”, ORSA Journal on Computing, Vol. 1, No. 3, pp 190-206.
- [6] Glover, F., 1990, “Tabu search - Part II”, ORSA Journal on Computing, Vol. 2, No. 1, pp 4-32.
- [7] Glover, F., 1990, “Tabu search : a

- tutorial", Interfaces, Vol. 20, No. 4, pp 74-94.
- [8] Halevi, G., Weill R., Technion, Haifa, 1980, "Development of flexible optimum process planning procedures", Annals of the CIRP, Vol. 29, No. 1, pp 313-317.
- [9] Ham, Inyong and Liu, S. C.-Y., 1988, "Computer-aided process planning: the present and the future," Annals of the CIRP, Vol. 37, No. 2, pp 1-11.
- [10] Hancock, T.M., 1989, "Effects of alternative routings under variable lot-size conditions," International Journal of Production Research, Vol. 27, No. 2, pp 247-259.
- [11] Hancock, T. M., 1988, "Effects of adaptive process planning on job cost and lateness measures," International Journal of Operations and Production Management, Vol. 8, No. 4, pp 34-49.
- [12] Hutchinson, G. K. and Pflughoeft, K. A., 1994, "Flexible process plans: their value in flexible automation systems", International Journal of Production Research, Vol. 32, No. 3, pp 707-719.
- [13] Lawler, E. L., Lenstra, J. K., Rinnooy Kan, A.H.G. and Shmoys, D.B., 1985, The traveling salesman problem: a guide tour of combinatorial optimization, John-Wiley & Sons.
- [14] Malek, M., Guruswamy, M., Pandya, M. and Owens, H., 1989, "Serial and parallel simulated annealing and tabu search algorithms for the traveling salesman problem", Annals of Operations Research, Vol 21, pp 59-84.
- [15] Nagi, R., Harhalakis, G. and Proth, J. M., 1990, "Multiple routing and capacity considerations in group technology applications," International Journal of Production Research, Vol. 28, No. 12, pp 2243-2257.
- [16] Rajamani, D., Singh, N. and Aneja, Y. P., 1990, "Integrated design of cellular manufacturing systems in the presence of alternative process plans", International Journal of Production Research, Vol. 28, No. 8, pp 1541-1554.
- [17] Rho, H. M., Geelink, R., vant Erve, A. H., Kals, H. J. J., 1992, "An Integrated Cutting Tool Selection and Operation Sequencing Method", Annals of the CIRP, Vol. 41/1.
- [18] Seo, Yoonho, 1993, Integrated Manufacturing Systems Design Through Process Plan and AGV Guidepath Selection, Ph.D. Thesis, Department of Industrial and Management Systems Engineering, The Pennsylvania State University, University Park, PA 16802.
- [19] Seo, Yoonho and Egbelu, P. J., 1995, "Process plan selection based on product mix and production volume", submitted to International Journal of Production Research.
- [20] Spur, R. W. and Eversheim, W. 1982, "Survey of Computer-Aided Process Planning Systems", Annals of the CIRP,

Vol. 31/1.

- [21] Wang, H.-P., 1986, Intelligent Reasoning for Process Planning, Ph.D. Thesis, Department of Industrial and Management Systems Engineering, The Pennsylvania

State University, University Park, PA 16802.

94년 9월 최초 접수, 95년 8월 최종 수정