

# 분산제조환경 하에서 최소 비용을 갖는 공급자 선정\*

고창성\*\* · 김태운\*\*

## Selection of Suppliers with Minimum Costs in a Distributed Manufacturing Environment\*

Chang-Seong Ko\*\* · Tae-Woon Kim\*\*

### ■ Abstract ■

With every passing month, a number of new virtual enterprises emerge on the internet. Many companies have responded to the idea of distributed manufacturing in which the value adding activities comprising a finished product are dispersed geographically in a remote place. This study suggests an idea for selecting suppliers in a distributed manufacturing environment, which enables machining companies to share their capacities.

A mathematical model is constructed to show how to minimize the sum of the operation and transportation costs based on alternative process plans considering several kinds of operation characteristics in the real world. Four heuristic algorithms are developed to get an optimal or near-optimal solution using tabu search since the model is proven NP-hard. Example problems are tested to illustrate their procedure and performance.

## 1. 서 론

향후 세계 경제의 주요 활동들의 상당부분이 인터넷을 매개로 이루어질 것으로 예측되며 이 경우 인터넷은 모든 경제영역의 핵심수단이 될 것으로 예상된다. 따라서 미래 정보사회에서는 인터넷 핵심기술 및 서비스 영역에서의 우위 여부에 따라 산

업 및 국가경제의 성장과 발전이 좌우될 것으로 추정된다[1]. 이러한 인터넷 환경의 중요성에 부합되어 최근에 전통적인 제조시스템과 사이버 환경이 결합된 형태로 나타난 것이 가상제조 시스템(Virtual Manufacturing System: VMS)이다. VMS는 제품과 제조 시스템의 물리적 요소, 논리적 요소, 그리고 그 거동을 엄밀하게 모델링하고 프로그램

\* 본 연구는 1998년도 정보통신연구개발사업 지원에 의해 수행되었음.

\*\* 경성대학교 산업공학과

화함으로써 실제 제품의 생산 전에 문제점들을 미리 파악할 수 있도록 해주는 시스템으로서, 세분해 보면 설계중심(Design-centered) VMS, 생산중심(Production-centered) VMS, 제어중심(Control-centered) VMS로 나눌 수 있다[2].

본 연구에서 고려되는 분산제조시스템은 이상의 가상제조시스템 중 생산중심(Production-centered) VMS에 속하는 것으로, 최근에 제품 개발 및 설계와 생산에 필요한 가공기술 및 공정계획에 대한 전문 지식 만을 소유하고, 실제 제품 생산은 외주업체를 아웃소싱(Outsourcing)하는 제조시스템의 형태를 취한다[10].

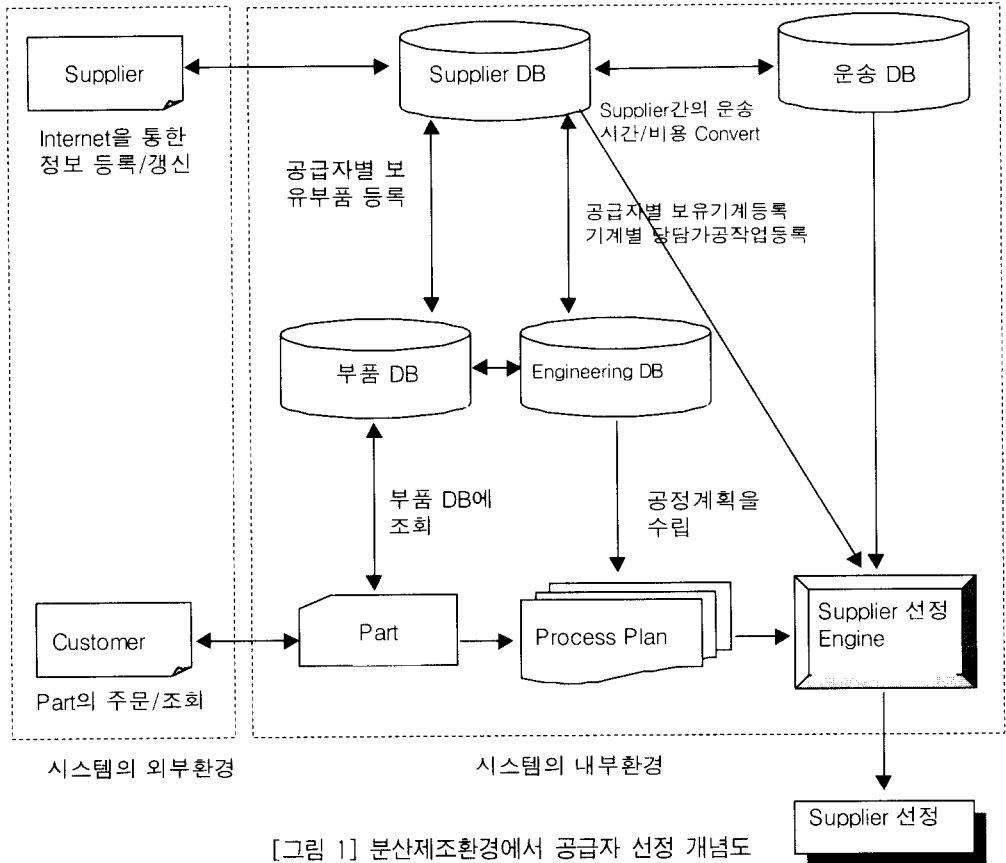
전통적인 생산시스템은 제품 가공에 필요한 모든 공정을 가능하면 자사에서 처리하고자 고가의 기계를 구입하였으나, 제품의 수요 변화에 따라 그 기계의 가동률이 현저히 떨어져 결국은 회사의 재무구조를 악화시키는 주요 요인으로 작용하고 있다. 최근에는 각 업체들이 생산기술 및 설비의 공유화를 통해 설비의 과잉 중복투자를 해결하고자 하는 것에 많은 관심을 갖게 되었다. 따라서, 본 연구에서 고려하고 있는 분산제조시스템은 각 공정을 담당할 기계들을 대상으로 이중 여유생산 능력을 가진 기계들을 보유한 업체들을 적절히 선정하여 제품 또는 부품을 생산하고자 하는 시스템이다. 즉, 대상시스템은 제품개발 및 공정계획 수립만을 전문적으로 취급하는 엔지니어링 역할만 담당하며, 실제 제품 가공은 여유 생산능력을 갖춘 업체가 본 시스템에 등록하여 인터넷 환경을 통하여 특정 제품 또는 부품 가공에 참여한다. 제조시스템의 환경을 인터넷기반으로 확장하게 되면 실제 환경과는 많은 차이가 발생한다. 인터넷 기반 분산시스템에서 정보교환과 관련된 문제로는 컴퓨터의 분산네트워크, 자료 공유, 도구의 공유와 자료흐름의 추적 등을 들 수 있다[4]. 분산환경 하에서 정보나 데이터를 통합하는 방법에는 이중 조직간의 연합은 느슨한 연합(Loosely-coupled Federation)이 되고, 조직내부의 기능간의 연합은 밀접한 형태(Tightly-coupled Federation)가 된다. 또한 분산환경과 관련

된 정보기술 인프라로서는 객체지향기법이 많이 이용되고 있다. OMG그룹에서 제안된 CORBA(Common Object Request Broker)는 플랫폼의 위치에 상관없이 분산된 환경에서 정보를 교환하는 수단으로 이용되고 있으며[6], 이의 사례로는 반도체를 위한 가상기업의 계획 및 통제를 위한 프로젝트인 X-CITTIC를 들 수가 있다[11].

[그림 1]은 본 연구에서 다루는 분산제조시스템 환경 하에서 공급자선정 모형의 개념도를 나타내 주고 있다. 시스템 외부에 위치한 고객이 특정한 공정을 위해서 인터넷을 이용하여 본 시스템에 접속하면 부품의 형태에 따라서 공정계획이 만들어 지고 이에 따른 모듈들이 작동하여 여러 가능한 외부의 가용 장비 중에서 최적의 가공 장비가 선정되는 과정을 개념적으로 도식화하여 나타내었다. 이를 위해, 본 연구에서는 인터넷을 이용하여 부품정보, 기계정보, 공정정보, 운송정보들을 데이터베이스화하여 각 지역에 산재되어 있는 고객과 기계를 보유하고 있는 공급자를 중개하여 부품의 검색과 주문이 가능하고 공급자의 다양한 기계정보를 알 수 있으며, 총 비용 즉 부품 가공비용과 운송비용의 합이 최소가 되도록 하는 공급자를 선정하는 엔진을 개발하는 것을 목표로 한다.

또한 보다 실제적인 생산환경을 반영시키기 위해 가공부품이 복수의 대체공정계획을 가질 수 있고, 각 기계에 따른 총가공시간과 기계의 능력을 비교하여 가공비용에 적절한 할인율과 벌금율을 적용시킨다. 따라서, 본 연구에서는 엔진 구축을 위해 수리적 정식화(Mathematical Formulation)를 구축하고 최적해(Optimal Solution) 또는 근사최적해(Near-optimal Solution)를 찾기 위해 Tabu Search에 기초한 탐색적 알고리즘을 개발한다.

Tabu Search[3]는 탐색과정 중 부분최적해(Local Optimum)에 빠질 가능성이 있는 조합최적화(Combinatorial Optimization)문제에서 최적해 또는 근사최적해를 찾는 Meta Heuristic으로서 그 동안 여러 분야의 연구활동에서 성공적 결과를 보여준 탐색기법으로 특히 유연제조시스템(Flexible Man-



ufacturing Systems)이나 셀형제조시스템(Cellular Manufacturing Systems)에서 부품 선정 문제, 공정계획 선정 문제 등을 해결하는 대표적 탐색기법으로 활용되고 있다[5, 8, 9].

## 2. 수학적 모형

본 연구에서는 인터넷을 이용하여 각 지역에 산재되어 있는 고객과 기계를 보유하고 있는 공급자를 증개하고 이러한 공급자를 선정하는 엔진을 개발한다. 이를 위해, 이 장에서는 부품이 복수의 대체공정계획을 가질 때 부품가공에 필요한 가공비용과 운송비용의 합이 최소가 되도록 사전에 정해진 여러 공정계획 중 하나를 선정하고, 선정된 공정계획에 따라 선택된 기계를 소유하고 있는 하나

혹은 그 이상의 공급자를 선정하는 수리적 모형을 구축하고자 한다. 이를 위해 본 연구는 다음과 같은 가정 하에 수행된다.

- (1) 각 부품에 대한 주요정보 즉, 수요, 공정특성, 크기 등을 알고 있다.
- (2) 기계당 가용 능력을 알고 있다.
- (3) 각 공정계획에 대해서 특정 기계에서 가공되는 부품의 가공시간 및 비용에 대해 알고 있다.
- (4) 복수의 대체공정계획은 엔지니어의 전문 지식과 경험에 의해 수립이 가능하다.
- (4) 기계간의 운송비용을 알고 있다.
- (5) 각 부품에 대한 수요는 납기일까지 평균화된다.
- (6) 대상 공급자는 품질 수준, 기술 능력 등의

기본 요건은 만족한다.

또한, 본 연구에서 활용되는 주요 기호 및 변수는 다음과 같다.

$i = 1, 2, \dots, m$  기계

$j = 1, 2, \dots, n$  부품

$p = 1, 2, \dots, p_j$  부품  $j$ 의 가능한 공정계획

$k = 1, 2, \dots, K(j, p)$  즉 부품  $j$ 가 공정계획  $p$ 를 선택했을 때의 해당공정

$(k) =$  공정을 순서화 시킨 것으로  $k$ 번째의 가공작업을 나타냄

$(j, p) =$  부품  $j$ 가 공정 계획  $p$ 에서 생산될 수 있는 조합

$d_j =$  부품  $j$ 의 일간 수요

$OC_{i(k)}(j, p) = (j, p)$ 에서  $k$ 번째 공정의 가공비용

$pt_{i(k)}(j, p) = (j, p)$ 에서  $k$ 번째 공정의 가공시간

$cu_j =$  부품  $j$ 의 단위 부피

$V =$  컨테이너의 부피

$N_j =$  부품  $j$ 가 적재된 컨테이너의 수,  $[\sum_j cu_j \cdot d_j/V]$

$c_1 =$  가공비용 할인율

$c_2 =$  가공비용 벌칙율

$e_1 =$  기계  $i$ 의 가용능력 적정 계수

$e_2 =$  기계  $i$ 의 가용능력 한계 계수

$Y_{i(k)}(j, p) = \begin{cases} 1 & \text{조합}(j, p)\text{에서 } k\text{번째 공정이 기계 } i\text{로 이루어 질 때} \\ 0 & \text{그 외의 경우} \end{cases}$

$Z_{jp} = \begin{cases} 1 & \text{부품 } j\text{가 공정계획 } p\text{로 생산될 때} \\ 0 & \text{그 외의 경우} \end{cases}$

$a_{i(k)} = \begin{cases} 1 & \text{기계 } i\text{에서 } k\text{번째 공정이 이루어 질 때} \\ 0 & \text{그 외의 경우} \end{cases}$

$b_{i(k)}(j, p) = \begin{cases} 1 & (j, p)\text{에서 } k\text{번째 공정이 이루어 질 때} \\ 0 & \text{그 외의 경우} \end{cases}$

$tc(i_1, i_2) =$  기계  $i_1$ 에서 기계  $i_2$ 까지의 컨테이너 단위 운송비용

$rt_i =$  기계  $i$ 의 총 작업요구시간

$at_i =$  기계  $i$ 의 가용능력

$s =$  부품 가공을 위한 원자재 공급지역을 나타내는 출발점

$t =$  가공작업이 완료된 부품이 납품되는 종착점

이상의 정의된 기호와 변수를 활용하여 총비용을 최소화시키는 수학적 모형은 다음과 같다. 여기서  $Y_{i(k)}(j, p)$ ,  $Z_{jp}$ 가 결정변수가 되며,  $a_{i(k)}$ ,  $b_{i(k)}(j, p)$ 는 모형 정식화에서 계수값을 나타낸다.

Minimize

$$Z = \sum_{i,j,p,k} d_j \cdot Y_{i(k)}(j,p) \cdot OC_{i(k)}(j,p) \quad (1)$$

$$+ \sum_{i,j,p,s} tc(s,i) \cdot Y_{i1}(j,p) \cdot N_j \quad (2)$$

$$+ \sum_{i_1, i_2, j, p} \sum_{k=1}^{K(j,p)-1} tc(i_1, i_2) \cdot Y_{i1(k)}(j,p) \cdot Y_{i2(k-1)}(j,p) \cdot N_j \quad (3)$$

$$+ \sum_{i,j,p,t} tc(i,t) \cdot Y_{i(K(j,p))}(j,p) \cdot N_j \quad (4)$$

where

$$OC_{i(k)} = \begin{cases} OC_{i(k)} : rt_i < e_1 \cdot at_i \\ c_1 \cdot OC_{i(k)} : e_1 \cdot at_i \leq rt_i < at_i \\ c_2 \cdot OC_{i(k)} : at_i \leq rt_i < e_2 \cdot at_i \\ M \cdot OC_{i(k)} : e_2 \cdot at_i \leq rt_i \quad i=1, 2, \dots, m \end{cases} \quad (5)$$

where

$$rt_i = \sum_{j,p,k} d_j \cdot Y_{i(k)}(j, p) \cdot pt_{i(k)}(j, p)$$

subject to

$$\sum_p Z_{jp} = 1 \quad j=1, 2, \dots, n \quad (6)$$

$$\sum_{i=1}^m a_{i(k)} Y_{i(k)}(j, p) = b_{i(k)}(j, p) Z_{jp} \quad (7)$$

$$Z_{jp} = (0, 1) \quad \begin{cases} (k) = 1, 2, \dots, K(j, p), \\ j = 1, 2, \dots, n, \\ p = 1, 2, \dots, p_j, \end{cases}$$

$$Y_{i(k)}(j, p) = (0, 1) \quad \begin{cases} j = 1, 2, \dots, n, \\ p = 1, 2, \dots, p_j, \end{cases}$$

$$Y_{i(k)}(j, p) = (0, 1) \quad \begin{cases} (k) = 1, 2, \dots, K(j, p), \\ j = 1, 2, \dots, n, \\ p = 1, 2, \dots, p_j, \\ i = 1, 2, \dots, m. \end{cases}$$

먼저 목적함수를 살펴보면, (1)은 부품가공에 소요되는 비용을 나타내며, (2)는 출발점에서 첫 번째 가공작업을 담당하는 기계까지의 운송비용을 나타내고, (3)은 기계간 운송비용을 나타내며, 또한 (4)는 최종공정을 마친 부품을 납품하기 위해 소요되는 비용을 나타낸다. 따라서 운송비용은 (2), (3), (4)의 합으로 표현할 수 있다.

본 연구에서는 보다 실제적인 생산 환경을 반영하기 위해 가공비용을 (5)과 같이 정의하였다. 예를 들면  $c_1 = 0.9$ ,  $c_2 = 1.5$ ,  $e_1 = 0.8$ ,  $e_2 = 2.0$ 이라고 가정하면 다음과 같이 설명 될 수 있다.

기계  $i$ 의 총 작업요구 시간이 기계능력의 80%일 때는 기계 가공비용( $oc_{ik}$ )을 그대로 적용하고 80% 이상 100%미만일 때는 10%의 할인을 적용한다. 100%이상 200%미만 일 때는 야근 등으로 인한 추가 비용이 들것을 예상하고 50%의 벌금을 적용한 것이다. 200%이상일 때는 야근 등으로도 운영이 불가능하기 때문에 기계가공비용에 큰 수  $M$ 을 곱하여 해의 대상에서 탈락시킨다. 다음으로 제약조건에 대해 살펴보면 (6)는 각 부품은 하나의 공정 계획만 선택된다는 것을 설명하며, (7)은 각 부품과 복수의 대체공정계획의 관계를 나타낸다.

상기 모형에서 목적함수는 총비용을 최소화하는 것으로, 여기서 총비용은 부품가공에 소요되는 가공비용과 운송비용의 합으로 구성된다. 이때, 제시된 모형은 Logendran의 2인[5]의 연구에서 알 수 있듯이 NP-hard에 속하는 문제이다. 따라서, 간단한 문제의 경우에는 수리적으로 최적해를 도출하는 것이 가능하나 문제 크기가 조금만 커져도 해 도출에 많은 어려움이 있기 때문에 현실적인 문제를 해결하기 위해서는 문제의 특성을 고려한 탐색적 알고리즘이 필요하다. 따라서, 본 연구에서는 Combinatorial 문제에서 최적해 또는 근사최적해를 도출하는데 성공적 결과를 보여준 Tabu Search에 기초한 4가지 탐색적 알고리즘을 개발한다.

### 3. Tabu Search에 기초한 탐색적 알고리즘

전 장에서 구축된 수리적 모형에서 최적 또는

근사최적해를 도출하기 위해 4가지 탐색적 알고리즘을 개발한다. 본 연구에서 개발된 알고리즘에 대해 개괄적으로 설명하면 각 부품에 대한 가능 공정 계획조합들 중에서 임의로 한 조합을 선정하고 이 조합에 대해 가공비용 및 운송비용의 합으로 표시되는 총비용이 최소가 되는 공정계획 및 공급자를 선정한다. 이때 조합 생성을 위해 활용될 주 논리는 Logendran의 2인의 연구[5]에서 사용된 Tabu Search에 기초하였다. 4가지 개발 알고리즘에 대해 구체적 설명을 위해 우선 ALG1-MET1이라 정의한 첫 번째 알고리즘에 대해 자세히 서술한 후, 나머지 3가지 알고리즘 ALG1-MET2, ALG2-MET1 및 ALG2-MET2를 ALG1-MET1과 비교를 통하여 설명할 것이다.

- Step 1. 각 부품에 대한 공정계획별 가공비용을 올림차순으로 정렬한다.
- Step 2. 각 부품에 대해 최소 가공비용을 갖는 공정계획을 초기해로 선정한다.
- Step 3. 대상 부품의 가공비용을 계산하고 또한 생산에 투입될 기계간의 부품 수송에 소요되는 운송비용을 계산한 후 이 두 가지 비용을 합하여 총비용으로 한다.
- Step 4. 초기해를 CL(Candidate List)에 등록한다. 이 때 CL은 장차 변동될 부품별 공정계획 조합을 형성한다. 그리고 이 때의 총비용을 AL(Aspiration Level)로 둔다. AL은 현재까지의 최소비용을 의미한다.
- Step 5. 각 부품에 대한 공정계획을 하나씩 변동시킨 후 각 공정계획조합에 대응되는 총비용을 Step 3에서와 같은 방법으로 계산한 후 최소의 비용을 갖는 공정계획조합을 찾는다. 이때 찾은 공정계획조합이 AL보다 작으면 '\*'를 할당하고 CL에 등록하며 AL을 이 비용으로 개정한다. 만일 최소비용을 갖는 공정계획조합이 복수 존재하면 모두 선택한다. 그리고, 이 때의 공정계획조합의 변동된 부품위치에 '\_'를 표시하여 Tabu로 등록한다. Tabu는 다음의 탐색과정에서 이 경우를 고려하지 말라는 의미가 된다.

- Step 6. 탐색과정 중 얻어진 최소 비용의 공정계획 조합이 AL보다 큰 값을 갖는 경우 새로이 AL을 이 값으로 교체하고 아울러 직전의 공정계획조합은 '\*\*'를 부여하여 부분최적해로서 IL(Index List)에 등록한다.
- Step 7. 지정된 횟수만큼 탐색한 후 IL에 등록된 부분최적해 중 최소값을 갖는 공정계획조합을 찾는다.

전술한 ALG1-MET1 외의 3가지 알고리즘에 대해 살펴보면, 우선 ALG1-MET2는 ALG1-MET1의 주 모듈과 부 모듈 Step 5 탐색과정 중 목적함수 값이 동일한 값을 갖는 경우 첫 번째 조합만을 CL에 등록시켜 수행하는 알고리즘이다. 즉, MET1과 MET2의 차이는 동일 목적함수 값을 갖는 경우에서의 처리과정 차이로 구분된다. 아울러, ALG2는 ALG1과는 달리 탐색과정을 조합의 다른 요소(Component)는 고정시킨 후 한 요소만을 변화시켜 가며 각 요소에서의 최소의 비용을 갖는 조합을 찾아나가는 과정을 Tabu Search를 활용한 것이며, 여기서 MET1과 MET2의 경우는 ALG1과 동일 과정을 채택한 것이다.

따라서, 본 연구에서는 전 장에서 제시한 수학적 모형의 해를 도출하기 위해 4가지 탐색적 알고리즘을 제시하며, 다음 장에서는 4가지 알고리즘 중 특히 ALG1-MET1을 예제를 통해 자세히 설명하며, 또한 현실적 문제 해결에도 적용이 가능한 지를 시험해보기 위해 다소 문제 크기가 큰 실험 문제를 통해 4가지 알고리즘을 평가한다.

#### 4. 알고리즘 평가를 위한 예제 수행

전 장에서 개발한 탐색적 알고리즘의 수행 과정 및 성능 평가를 위해 복수 공정계획을 갖는 셀형 제조시스템의 설계에 관한 Rajamani 외 2인[7]과 Logendran 외 2인[5]의 연구에서 도입한 문제를 대상으로 예제를 수행한다. 대상 예제에서는 4가지 부품, 3가지 기계, 3가지 공정(4P\*3M\*3O)의 경

우를 고려한다.

<표 1, 2, 3>은 4가지 부품과 3가지 기계에 대응되는 공정계획, 가공시간, 가공 비용에 대한 정보와 운송비용 등을 제시하고 있다. 예를 들어, <표 1>을 살펴보면 부품 4(P4)는 2개의 공정계획이 있으며, 그 첫 번째 공정계획은 공정 1, 2, 3으로 구성되어 있고 순서또한 1, 2, 3 순으로 이어진다. 두 번째 공정계획은 순서화된 공정 1, 2를 요구한다. <표 2>에서 공정 1은 기계 1이나 기계 3, 가공작업 2는 기계 2나 기계 3, 가공작업 3은 기계 1이나 기계 2를 이용하여 부품을 가공할 수 있다. 따라서, 부품 4에서 첫 번째 공정계획은 8(2\*2\*2)가지 조합의 기계로 가공할 수 있다. <표 4>는 각 기계간의 운송비용을 나타내고 있다. 여기서 운송비용은 컨테이너 한 개당 운송비용을 나타낸다. 예를 들면, 출발지는 울산에 있고 기계 1과 기계 2는 창원, 기계 3은 서울에 있는 경우를 생각하자. 같은 창원지역내에 있는 다른 공장의 경우는 운송비용을 2, 동일 공장의 경우는 운송비용을 0으로 둔다. 그리고 울산에서 창원까지의 운송비용은 2, 울산에서 서울까지는 5, 창원에서 서울까지는 7의 운송비용이 든다.

이상의 자료를 기초로 전 장에서 개발한 4가지 탐색적 알고리즘 중 ALG1-MET1은 다음과 같이 수행된다.

<표 1, 2, 3>의 자료를 기초로 각 부품에서 가능한 공정계획조합을 생성한다. 여기서, 공정계획 조합은 부품번호 가공비용만을 기준으로 하여 올림차순으로 정렬한다. 따라서, 실제 가능한 공정계획 조합은  $8 \times 16 \times 16 \times 12 = 24576$ 개의 조합을 생성할 수 있으며, 모든 조합이 고려대상이 된다. 우선 공정계획조합 (1,1,1,1)을 초기해로 선택하는데 여기서 (1,1,1,1)의 의미는 각 부품에 대해 가공비용이 최소가 되는 조합을 나타낸다. 일단 (1,1,1,1)이 선택되면 이 조합에서 운송비용을 계산한 후 가공비용과 합산하여 해당조합의 총비용을 계산한 후 이 조합을 CL에 등록한다. 이 값은 추후 부분최적해가 될 가능성이 있으므로 하나의 '\*'를 할당한다.

<표 1> 각 부품에 대응되는 공정계획 및 공정

	j = 1		j = 2		j = 3			j = 4	
	p=1	p=2	p=1	p=2	p=1	p=2	p=3	p=1	p=2
(k) = 1	2	2	1	2	1	3	3	1	1
(k) = 2	1	3	3	3	3	1	2	2	2
(k) = 3			2	1			1	3	
수요	10		10		10			10	
부피	10		15		5			20	

<표 2> 공정 k에 대응되는 기계 i 및 기계사양(기계 능력)

	i = 1	i = 2	i = 3
(k) = 1	1		1
(k) = 2		1	1
(k) = 3	1	1	
기계능력	100	100	100

<표 3> 각 부품의 공정계획, 공정, 기계에 대응되는 가공시간 및 가공비용

		$p_{ik}(j, p), oc_{ik}(j, p)$								
		j = 1		j = 2		j = 3			j = 4	
		p=1	p=2	p=1	p=2	p=1	p=2	p=3	p=1	p=2
(k)=1	i=1	5.3		3.4	4.2	2.2	1.2	8.1	1.2	9.7
(k)=1	i=3	7.2		4.3	7.5	2.2	2.4	9.2	2.1	8.9
(k)=2	i=2	3.5	9.8	7.8	3.3			5.9	2.3	9.8
(k)=2	i=3	4.3	7.9	7.7	2.3			3.10	2.4	10.9
(k)=3	i=1		8.8	10.9	6.5	4.7	11.7	7.4	3.5	
(k)=3	i=2		7.7	8.9	6.6	3.2	8.8	9.5	2.6	

<표 4> 기계간의 운송비용을 나타내는 From-to Chart

	a	i = 1	i = 2	i = 3
a	-	2	2	5
i = 1	2	0	0	7
i = 2	2	0	0	7
i = 3	5	7	7	0

다음 단계는 Tabu Search에 따라 근처의 공정 계획조합을 탐색한다. 먼저 (1,1,1,1)에서 근처의 (2,1,1,1)의 조합을 대상으로 한다. 여기서, 부품 1에 해당하는 요소 (2)의 '1'은 현재 Tabu 임을 표시해 준다. 따라서, 가능한 근처의 수요조합은 (2,1,1,1), (1,2,1,1), (1,1,2,1), (1,1,1,2)의 4가지 새로운 조합을 형성한다. 이 상태에서의 총기대비용은 각각 429, 465, 440, 425가 되며, 초기해인 (1,1,1,1)의 430과 비교하면 수요 조합 (1,1,1,2)가 최소 비용을 갖게 되어 다음 단계의 기준이 될 수 있어서 새로이 CL

에 삽입한다. 만일 (1,1,1,2)의 비용 425가 다음 단계의 값보다 작으면 부분최적해가 될 수 있다. 이런 경우에는 이미 하나의 '\*'를 할당받았으므로 각각 '\*\*'를 할당하고 IL에 등록되며, 아울러 AL은 425로 변경된다.

이상의 과정을 일정 개수의 부분최적해를 얻을 때까지 수행한다. <표 5>는 부분최적해 개수를 5개로 하여 ALG1-MET1을 수행한 결과를 보여주고 있다. 이 알고리즘에서 선택된 해는 주 수행횟수 4번째 즉 3번째 부분최적해에서 선택된 공정계획조합 (1,1,2,3)을 취하며, 이때의 총비용은 369이며 수행시간은 14초가 걸렸다.

<표 5> ALG1-MET1의 수행 과정

No.	조합	총 비용	비고
1	(1,1,1,1)	430	*
2	(1,1,1,2)	425	*
3	(1,1,1,3)	401	*
4	(1,1,2,3)	369	**
5	(2,1,2,3)	400	
6	(2,2,2,3)	420	
7	(3,2,2,3)	406	**
8	(4,2,2,3)	423	
9	(4,2,2,4)	464	
10	(4,2,2,5)	423	**
11	(4,3,2,5)	492	
12	(4,4,2,5)	485	**
13	(4,5,2,5)	552	
14	(4,6,2,5)	513	**
15	(4,7,2,5)	542	

\*: CL, \*\*: IL

또한, 동일한 입력자료로서 나머지 3가지 알고리즘을 수행한 결과 <표 6>에서 알 수 있듯이 모두 동일한 비용을 갖는 해를 도출하였으며, 참고적으로 전체탐색해(Enumeration)을 통해 구한 최적해와 동일한 해를 구하였다. 다만, ALG2-MET1과 ALG2-MET2를 적용한 경우는 ALG1-MET1과 ALG1-MET2를 적용한 경우보다 동일 부분최적해 개수에 대해 평균 약 6~10배 시간이 더 걸림을 알 수 있다.

본 연구에서 개발한 4가지 알고리즘은 Boland사의 Dephi 3.0으로 구현하여 PC 펜티엄 166MHz에서 수행되었다.

<표 6> 예제를 실행시킨 결과값

알고리즘	부분 최적해 개수		5개	10개	15개	20개	최적해		
	5개	10개							
ALG1-MET1	369	14초	369	21초	369	30초	369	50초	(1,1,2,3) 의 조합. 369
ALG1-MET2	369	13초	369	21초	369	30초	369	50초	
ALG2-MET1	369	98초	369	210초	369	291초	369	382초	
ALG2-MET2	369	74초	369	202초	369	138초	369	329초	

다음으로는 전술한 예제보다 보다 큰 문제에 대해 알고리즘을 수행하여 비교하였다. 여기서는 5개의 부품, 4개의 기계와 4개의 공정으로 구성된 즉  $5P \times 4M \times 4O$  구조를 가진 임의의 10개의 문제를 대상으로 4가지 알고리즘을 수행하였다. 이 때, 공정 계획을 수행하기 위한 공정 선택은 일양분포  $U(0, 1)$ 을 이용하여 임의로 생성하였다. 예를 들어, 4개의 공정이 있는 경우에 0부터 0.249999사이의 수가 나오면 첫 번째 공정이 선택되고, 0.25에서 0.49999 사이라면 두 번째 공정이 선택된다. 만일 두 번째 임의의 수가 첫 번째와 같은 구간이면 이 수는 포기하고 다시 선택하게 된다. 다른 모수에 대해서도 같은 방법을 사용하였다. 즉, 각 부품의 가공시간은  $U(10, 20)$ 에서, 각 부품의 가공비용은  $U(15, 30)$ 에서, 각 부품의 수요 발생은  $U(10, 50)$ 을 사용하였다.

그리고, 부품 1, 2, 3, 4, 5의 부피는 각각 10, 15, 5, 20, 10으로 하고 컨테이너의 부피는 100으로 한다. <표 7, 8, 9, 10>은 10개의 실험문제 중 하나의 입력자료 예를 보여주고 있으며, <표 11>에서는 임의로 생성된 10가지 문제에 대해 4가지 알고리즘을 수행한 결과가 나타나 있다. 각 문제에서 이상의 자료에 의해 생성되는 공정계획 조합은  $12 \times 8 \times 8 \times 8 \times 8 = 49152$ 개가 생성된다.

수행 결과를 분석하면 소요총비용에서는 ALG2가 ALG1에 비해 총비용이 평균 3.3% 나은 해를

<표 7> 각 부품에 대응되는 공정계획 및 공정( $5P \times 4M \times 4O$ )

가공 작업 순서(k)	j=1		j=2		j=3		j=4		j=5		
	p=1	p=2	p=3	p=1	p=2	p=1	p=2	p=1	p=2	p=1	p=2
(k)=1	2	2	3	1	4	2	2	1	3	4	2
(k)=2	1	4	1	4	2	4	3	2	4	1	4
(k)=3											
(k)=4											

<표 8> 공정 k에 대응되는 기계 i( $5P \times 4M \times 4O$ )

	i = 1	i = 2	i = 3	i = 4
k = 1	1	1		
k = 2			1	1
k = 3	1			1
k = 4		1	1	

<표 9> 각 부품의 공정계획, 공정, 기계에 대응되는 가공시간 및 가공비용( $5P \times 4M \times 4O$ )

		j=1			j=2		j=3		j=4		j=5	
		p=1	p=2	p=3	p=1	p=2	p=1	p=2	p=1	p=2	p=1	p=2
k=1	i=1	10.15			12.24	13.25				19.19		15.22
k=1	i=2	14.20			11.16	12.23				18.19		17.30
k=2	i=3	13.19	15.24				12.23	12.24	16.16	14.16		15.26
k=2	i=4	10.23	13.22				11.19	17.18	18.22	12.22		16.22
k=3	i=1			10.16					15.19		17.25	19.19
k=3	i=4			11.22				14.25		18.28		18.19
k=4	i=2		12.21		10.17	10.18	19.16				11.28	11.22
k=4	i=3		11.23		13.19	12.22	16.19			16.29	12.28	

<표 10> 기계 i간의 수송비용을 나타내는 From-to Chart( $5P \times 4M \times 4O$ )

	a	i = 1	i = 2	i = 3	i = 4
a	0	5	3	9	7
i = 1	5	0	8	14	5
i = 2	3	8	0	0	7
i = 3	9	14	0	0	8
i = 4	7	5	7	8	0

<표 11> 4가지 알고리즘의 비용과 시간 비교

	ALG1-MET1		ALG1-MET2		ALG2-MET1		ALG2-MET2		비고
	비용	시간	비용	시간	비용	시간	비용	시간	
1	6153	14	6153	14	6153	70	6153	70	
2	8053	23	8053	23	7374	117	7374	117	
3	8624	24	8624	23	8028	102	8028	102	
4	6772	16	6772	16	6677	94	6677	94	
5	9608	21	9608	21	9193	84	9608	78	
6	5626	16	5626	15	5478	85	5478	85	
7	4351	21	4351	21	4351	76	4351	76	
8	5190	11	5190	11	5121	78	5121	78	
9	5682	25	5682	25	5973	130	5973	130	
10	8475	14	8475	14	7656	98	7656	98	
평균값	6853.4	18.5	6853.4	18.3	6600.4	93.4	6641.9	92.8	

도출하였다. 그러나, 소요시간 상으로는 ALG2가 ALG1에 비해 평균 5배 더 걸렸음을 알 수 있다. 또한 ALG1과 ALG2의 경우 모두 MET1과 MET2의 경우에는 거의 차이가 없음을 알 수 있다. 이상의 실험 결과를 통해 빠른 시간내에 해를 도출해야 하는 상황에는 ALG1이 선호되며, 반면에 비록 장



시간이 소요되더라도 보다 나은 해를 원한다면 ALG2가 추천된다.

다음으로는 현실문제 적용에 대한 평가를 위해 기계의 댓수가 상당히 많은 보다 큰 문제 즉 (5P\*15M\*60)에 대해 ALG1-MET1을 수행하였다. <표 12>는 이 문제에 대한 가공시간과 가공비용에 대한 입력자료를 보여주고 있는데, 이 문제에서는 약  $2.1 \times 10^{11}$ 개의 공정계획조합이 생성된다. 부품 수요는 각각 40, 10, 20, 50, 30일 때 부분최적해가 5개 나올 때까지 수행하였다. 이 결과 도출된 공정조합은 (2,6,2,1,3)이고, 이때의 총비용은 6230, 수행시간은 104초가 소요되었다.

<표 12> 공정계획 p에서 부품 j와 공정 k로 구성된 기계 i의 요구 가공시간과 가공비용(5P\*15M\*60)

		j=1		j=2		j=3		j=4		j=5	
		p=1	p=2	p=1	p=2	p=1	p=2	p=1	p=2	p=1	p=2
k=1	i=1	17:25		14:20	16:25		15:15		19:15	16:30	20:25
k=1	i=2	10:20		19:15	14:20		17:20		16:20	11:20	13:25
k=1	i=3	20:25		12:15	19:25		14:20		19:20	14:25	17:15
k=1	i=4	13:25		14:30	16:30		18:25		18:25	15:25	17:25
k=2	i=2	17:15	19:15		12:15	11:25	11:25		19:15	18:15	18:30
k=2	i=3	17:25	16:20		16:15	14:25	19:30		17:15	19:25	10:25
k=2	i=4	18:30	19:20		14:30	15:30	15:15		18:30	19:20	16:20
k=2	i=5	10:25	18:25		10:30	16:30	16:30		15:15	17:30	14:30
k=3	i=1	16:20	19:15			11:25		11:15	15:30		
k=3	i=2	14:30	17:15			13:15		18:25	16:20		
k=3	i=10	10:30	18:30			14:15		17:30	11:25		
k=3	i=11	18:20	15:15			16:30		15:25	13:15		
k=4	i=4			17:20	13:30	11:20		17:20	14:15	19:20	
k=4	i=5			17:15	14:25	14:25		17:15	16:30	18:25	
k=4	i=6			18:30	15:25	15:25		18:30	11:20	19:15	
k=4	i=7			14:20	16:20	18:15		14:20	14:25	17:15	
k=5	i=2		12:15			19:25	18:30				14:15
k=5	i=3		14:20			19:20	14:25				16:30
k=5	i=8		19:25			17:30	17:20				11:20
k=5	i=9		13:20			11:30	19:25				14:25
k=6	i=3			12:15			11:20	17:15			
k=6	i=4			13:25				14:15	17:25		
k=6	i=7			14:30				15:15	18:30		
k=6	i=8			11:15			16:20	10:25			

## 5. 결론

본 연구에서 고려하는 분산제조시스템은 각 공정을 실제 처리할 기계들을 소유하지 않고, 사외에서 여유생산 능력을 가진 기계들을 보유한 많은 업체들이 위탁 처리할 수 있도록 하는 시스템이다. 즉, 대상시스템은 제품개발 및 공정계획 수립만을 전문

적으로 취급하는 엔지니어링 역할만 담당하며, 실제 제품 가공은 여유 생산능력을 갖춘 업체가 본 시스템에 등록하여 인터넷환경을 통하여 계속적으로 특정 제품 또는 부품 가공에 참여한다. 본 연구에서는 가공비용과 운송비용의 합을 최소화시키는 공급자를 선정하는 수리적 모형을 제시하였고, 4가지 알고리즘(ALG1-MET1, ALG1-MET2, ALG2-MET1, ALG2-MET2)을 개발하여 실험을 통해 각 알고리즘의 성능 및 소요시간을 비교하였다. 수행결과를 보면, ALG1-MET1, ALG1-MET2는 ALG2-MET1, ALG2-MET2와 비교해 볼 때 총비용 측면에서는 큰 차이가 없지만 수행시간 측면에서는 훨씬 적은 시간을 요구하기 때문에 짧은 시간내에 결정을 내야하는 상황에는 더 효과적이다. 그리고, 규모가 큰 문제를 통해서 ALG1-MET1을 수행함으로써 현실 문제 적용 가능성을 입증하였다.

비록 본 연구에서 다룬 분산제조환경 하에서 공급자 선정 모형은 가공비용과 운송비용 등 비용 요소만을 고려하였으므로 이 결과를 바로 현장에 적용하기에는 많은 무리가 따른다. 즉, 대상 업체의 생산 기술 능력이나 품질 수준 유지 및 납기일 고려 문제 등이 반드시 추가로 만족되어야 할 것이나, 본 연구에서는 수리적 모형 구축을 통한 해법 개발에 초점을 두었기에, 이상의 필수적 고려사항은 기본적으로 만족하는 업체를 대상으로 하여 비용을 최소화하는 목표 하에 선정 작업을 수행하도록 되어 있다. 따라서, 모형 수립에서 기본적으로 만족한다고 가정된 기술 능력, 품질 수준 및 납기 등을 고려한 모형 수립 및 해법 개발이 계속적으로 수행되어야 할 것이다. 그러나, 해법 개발에 있어서는 본 연구에서 제시했던 알고리즘에서 만족해야 할 기술, 품질 및 납기 등의 현실적 제약조건들은 프로그래밍 처리에 있어서는 큰 어려움 없이 해결할 수 있기 때문에 제시된 알고리즘을 토대로 이론적이 아닌 현장 지향적 시스템 개발은 용이할 것이다. 단, 실세계의 문제에 적용 시 분산환경에서 발생하는 정보의 교환, 데이터베이스의 공유 및 보안문제, 이기종 컴퓨터간의 통신 및 프로토콜과 다

른 기업문화의 차이에서 오는 업무방식의 차이 등이 해결되어야 할 과제로 판단된다.

## 참 고 문 헌

- [1] 정보통신부, 「인터넷 활성화를 위한 인터넷 비즈니스 발전대책」, 1998.
- [2] 한관희, “CIM의 발전 방향-VM(Virtual Manufacturing)”, 「산업공학」, 제11권, 제3호(1998), pp.219-225.
- [3] Glover, F., “Tabu search: Tutorial”, *Interfaces*, Vol.20, No.4(1990), pp.74-94.
- [4] Hardwick, M., D.L. Spooner, T. Rando and K. C. Morris, “Sharing manufacturing information in virtual enterprises”, *Communications of the ACM*, Vol.39(1996), pp.46-54.
- [5] Logendran, R., P. Ramakrishina and C. Sri-kandarajah, “Tabu search-based heuristics for cellular manufacturing systems in the presence of alternative process plans,” *International Journal of Production Research*, Vol.32, No.2(1994), pp.273-297.
- [6] OMG, “CORBA: The common object request broker”, *Revision 2.0*(1995).
- [7] Rajamani, D., N. Singh and Y.P. Aneja, “Integrated design of cellular manufacturing systems in the presence of alternative process plans”, *International Journal of Production Research*, Vol.28, No.8(1990), pp.1541-1554.
- [8] Srivastava, B. and W. Chen, “Part type selection problem in flexible manufacturing systems: tabu search algorithms”, *Annals of Operations Research*, Vol.41(1993), pp.279-297.
- [9] Veeramani, D., A.H. Stinnes and D. Sanghi, “Application of tabu search to process plan optimization for four-axis CNC turning centres”, *International Journal of Production Research*, Vol.37, No.1(1999), pp.207-218.
- [10] Virtual Manufacturing Technical Workshop, Compiled and Edited by Lawrence Associates Inc., (1994), pp.8-9.
- [11] Zhou, Q. and C.B. Besant, “Information management in production planning for a virtual enterprise”, *International Journal of Production Research*, Vol.37, No.1(1999), pp.207-218.