

# 인공지능 개념을 이용한 공장 설비배치 알고리즘 개발 (Development of Facility Layout Design Algorithm Based on Artificial Intelligence Concept)

김 환 성 \* 이 상 용 \*

## ABSTRACT

The purpose of this study is to propose a facility layout design algorithm based on artificial intelligence concept, and then to develop a computer program which is more practical than any other conventional facility layout design systems.

The algorithm is composed of five step layout procedures; knowledge and data input, knowledge interpretation, priority determination, inference of layout design, and evaluation. In the step of priority determination, the algorithm is divided into single row and multi row layout problem. In the step of inference of layout design, alternatives are generated by constraints-directed reasoning and depth first search method based on artificial intelligence concept. Alternatives are evaluated by the moving cost and relationship value by interactive man-machine interface in the step of evaluation.

As a case study, analytical considerations over conventional programs such as CRAFT and CORELAP was investigated and compared with algorithm proposed in this study. The proposed algorithm in this study will give useful practical tool for layout planner. The computer program was written in C language for IBM PC-AT.

---

\* 건국대산업대학원

\*\* 건국대산업공학과 교수

## 1. 서 론

설비배치 알고리즘 들은 크게 2종류로 나눌 수 있는데, 하나는 수학적 최적화 알고리즘이고 다른 하나는 발견적(Heuristic)알고리즘이다. 수학적 최적화 알고리즘은 1957년 Koopmans와 Beckman[1]의 QAP(Quadratic Assignment Program)모델링을 시초로 해서 연구가 진행 되어왔는데, 방대한 계산량을 필요하기 때문에 현실적인 설비배치 작업에의 적용은 용이하지가 않다. Sahni와 Gonzalez (1976)[2]는 QAP가 NP-Complete문제라는 것을 증명하였고 Burkard(1984)[3]는 QAP가 FORTRAN에 의한 Branch and Bound해법을 적용하는데  $N^3 + 5.5N^2 + 17.5N$ 의 컴퓨터 메모리가 필요함을 보여 주었다. 수학적 최적화 알고리즘의 이러한 단점 때문에 발견적(Heuristic)해법에 대해 많은 연구가 집중 되어왔다. 1955년 R.Muther에 의해서 Relationship value에 대한 개념이 소개된 이후로 CRAFT (1963)[4], CORELAP(1967)[5], ALDEP (1967)[6]등의 컴퓨터를 이용한 발견적 해법 등이 연구되었고 이들 알고리즘은 설비배치 설계자들에게 유용한 대안들을 제시 해주는 역할을 해왔다. 그러나 아직까지도 생산현장에서의 설비배치 문제는 이론적 해법 보다는 경험있는 설계자의 감각에 의존하고 있는것이 현실이다. 이러한 이유중의 하나는 설비배치 문제의 다변량성과 다목적성에 있다. 설비와 설비간의 이동비용, 운반설비의 종류, 설비의 특성, 설비의 형태, 배치공간의 형태, 작업공간의 특성들이 설비배치 문제의 목적함수와 제한조건들로 작용한다. 이러한 변수들을 더 많이 고려하면 할수록 더 효율적인 해에 접근 할수있는데, 이와 비례해서 해법의 절차와 계

산량은 더욱 복잡해지고 방대해진다.

최근 들어 보다 많은 변수들을 효과적으로 고려하기 위한 하나의 방법으로 전문가(expert)시스템을 이용한 설비배치 해법의 연구가 주목받고 있다. 특히 최근 생산 시스템으로 크게 각광받고 있는 유연생산시스템(flexible manufacturing system)이나 F.A(factory automation)공장의 설비배치문제는 그 성격상 전문가 시스템의 적용이 기존 알고리즘들 보다 효율적이라 판단된다.

본 연구의 목적은 설비배치 문제 해결에 전문가 시스템적 개념을 도입하여 설비배치 알고리즘을 개발하고 이를 토대로 퍼스널 컴퓨터에서 중규모이상의 설비배치 문제를 해결할 수 있는 컴퓨터 프로그램을 개발하는데 있다.

## 2. 인공지능(artificial intelligen)을 이용한 설비배치 알고리즘의 기존연구 고찰

설비배치문제에 인공지능적 기법들을 적용하는 목적은 실질적인 배치문제에서 요구되는 다목적함수의 고려와 보다 지능적인 방법으로 배치안을 탐색하기 위해서이다. 따라서 기존의 알고리즘들보다 현실적인 대안을 제시해주는 반면 시스템의 구축이 어렵고 복잡하며 때때로 대용량의 컴퓨터 시스템을 요구한다. 또한 지식공학, 고도의 컴퓨터 프로그램 기술, 해당분야의 전문지식들 여러분야의 지식을 필요하므로 각분야에 대한 유기적인 결합이 없이는 성공하기 어렵다. 인공지능을 이용한 설비배치의 주요 기존연구의 특성은(표 2-1)와 같다.

(표 2-1) 인공지능을 이용한 알고리즘

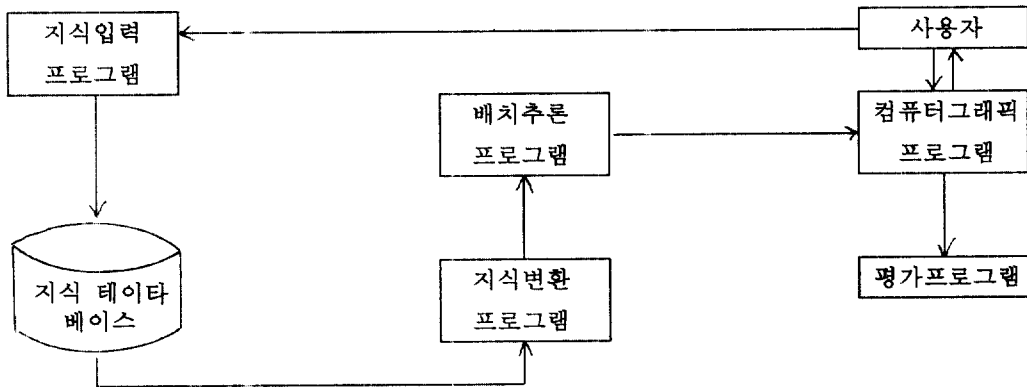
| 알고리즘           | 주요 특성   | 연구자   |
|----------------|---|---|
| FADES          | 지식(knowledge)base, data base, PROLOG를 이용한 forward chaining 깊이우선탐색 추론 구조를 구성하여 설비배치, 장비의 선택, 투자 분석 등을 행할수있음.   | Fisher E.L. & S.Y Nof(1984)[7]                  |
| IFLAPS         | pattern recognition system과 전문가 시스템을 이용하여 이동비용, 소음, 안전성의 다목적 함수를 고려 할수 있음<br>backward chaining추론 구조를 채택하고 있음. | S.R.T.Kumara, R.L.Kashyap & C.L.Moodie(1988)[8] |
| KBML           | 운반설비에 따른 배치형태의 지식 base를 구축하여 if-then 룰(rule)에의한 forward chaining 배치 추론을 행함.<br>단일라인과 복수라인으로 분류 배치를 수행.        | Heragu S.S & A.Kusiak (1988)[9]                 |
| 제약지향 설비배치 알고리즘 | 제약지향탐색(constraint-directed search)기법에 의한 배치안 생성.<br>각 설비의 제약정보를 지식 base 화만 하면 어떤 형태의 배치계획에도 폭넓게 사용할수 있음.      | S.Akagi & K.Fujita (1990)[10]                   |

### 3. 설비배치 알고리즘의 설계

System)의 구성 및 흐름도

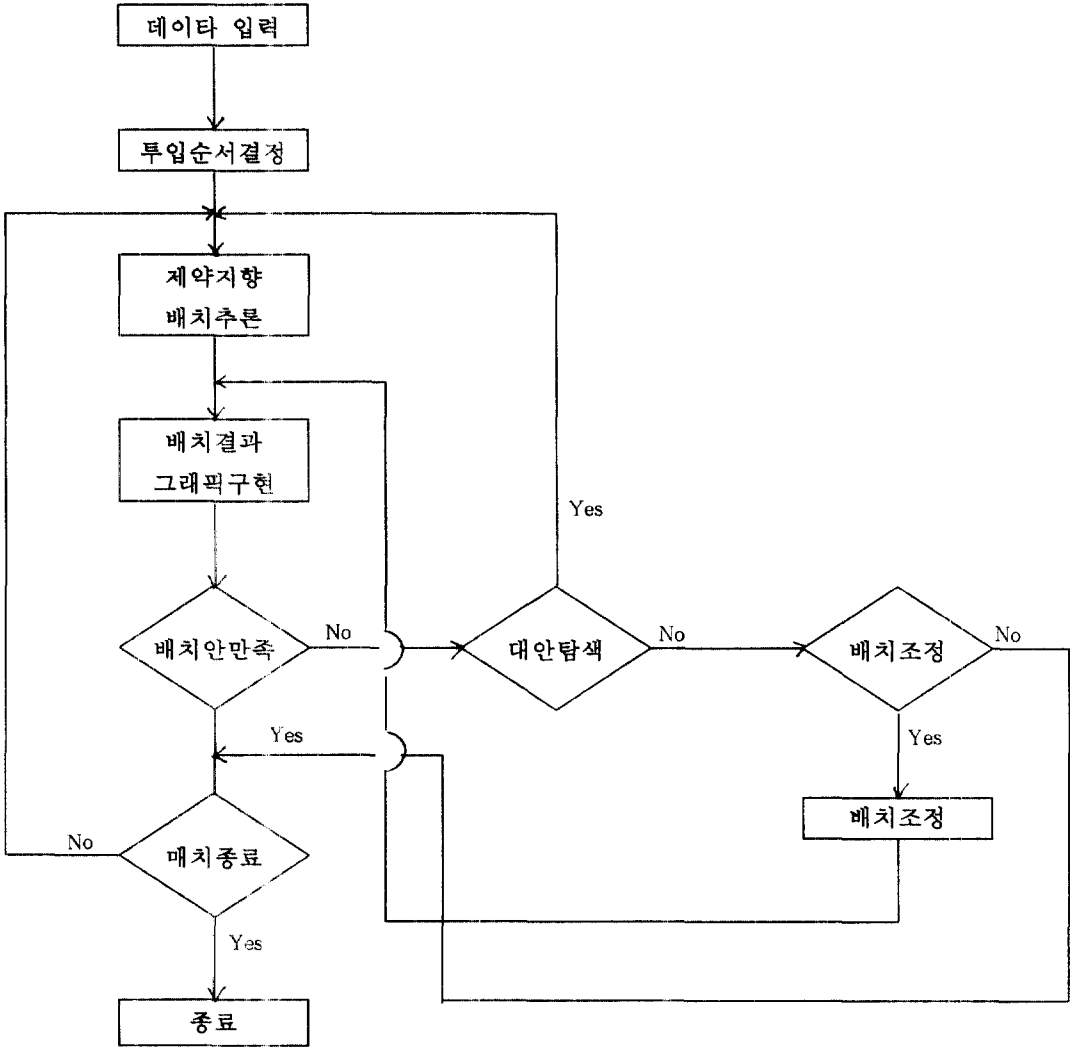
본 연구에서 개발된 EFLDS는 (그림 3-1) 과 같이 구성되어있다.

#### 1) EFLDS(Expert Facility Layout Design



(그림 3-1- EFLDS 구성

그리고 EFLDS의 알고리즘의 흐름도는 (그림3-2)와 같다.



(그림 3-2) EFLDS 배치 흐름도

(그림 3-2)에 의하면 사용자는 우선 지식입력(Knowledge Editor)프로그램에 배치 하고자 하는 설비들의 정보를 입력하게 되는데, 입력 데이터(data)와 지식 정보로는 다음과같은 사항들이 필요하게 된다.

- (1) 배치하고자 하는 설비의수
- (2) 각 설비간의 이동비용을 나타내는 From-To chart
- (3) 배치공간의 면적과 각 설비의 면적과 형태
- (4) 고정배치를 원하는 설비수와 고정위치 정보

- (5) 각 설비의 배치 형태의 유연성 여부
- (6) 설비와 설비 간의 운반수단의 종류
- (7) 설비 위치의 제약에 대한 방향정보
- (8) 각 설비간의 상호관련도(Relationship)

위와 같은 입력정보는 팝업(Popup)메뉴 방식과 대화형 입력방식에 의해서 손쉽게 처리할 수 있다. 데이터(data)입력이 끝나면 지식변환 프로그램(Knowledge interpreter)에 의해서 일반 Text 파일로 변환한다. 이 데이터를 이용하여 투입순서결정 프로그램에서 단일라인배치를 할 것인가, 복수라인배치를 할 것인가를 결정하고 이에 따라 각 설비의 배치우선 순위가 결정된다. 배치우선 순위가 결정되면 배치실행 부분에서 각 설비에 대한 정보를 토대로 배치안을 생성하게 된다. 배치안은 컴퓨터에 의해서 다른 대안을 계속해서 생성시켜 볼 수 있고 사용자에게 의해서 원하는 지점으로 배치안을 옮길 수도 있다. 이때 각 시점의 이동비용 평가치와 인접도 평가치에 대한 정보를 화면상에서 얻을 수 있다. 이동비용 평가치와 인접도 평가치는 다음과 같이 정의 하였다.

$$\text{이동비용 평가치} = \sum_{\text{모든설비}} \text{이동빈도} \times \text{단위}$$

당 이동비용 × 설비의 중앙에서 가장 짧은 이동거리

$$\text{인접도 평가치} = \sum_{\text{모든설비}} \text{인접도 가중치} \times$$

설비의 중앙에서 가장 짧은 이동거리

설비간의 이동거리는 설비의 중앙점에서부터 계산되는데, 이동비용 평가치는 CRAFT의 이동비용 평가치와 같은 방법이다. 이동비용 평가치와 인접도 평가치는 작을수록 우수한 대안이 된다.

## 2) 투입순서 결정 알고리즘

EFLDS는 운반수단의 형태와 설비의 수에 따라 단일라인과 복수라인의 설비배치 문제로 분류하여 배치 투입순서를 결정하고 있다. 운반수단의 형태로는 다음과 같은 10개의 운반수단을 고려하고 있다.

- (1) Over Head Conveyor
- (2) Loop Type Conveyor
- (3) Open Type Conveyor
- (4) Rotary Type Robot
- (5) Stacker Crane Type Robot
- (6) Gantry Type Robot
- (7) A.G.V (Automatic Guided Vehicle)
- (8) Over Head Cart
- (9) Other Type Device (forklift 등)
- (10) Unknown

위의 10가지 형태의 운반수단 가운데 (2),(3),(4),(5)의 운반수단을 사용하면서 배치하고자 하는 설비의 수가 10개를 넘지 않을 경우에는 단일라인 배치를 그렇지 않은 경우는 복수라인 배치를 하게 된다.

## 3) 단일라인(Single Row) 투입순서결정 알고리즘

단일라인의 경우 목적함수를 이동비용 최소화 문제로 둔다면 다음과 같이 수식화 시킬 수 있다.

$$\text{목적함수} \quad \text{Min} \sum_{i=1}^{n-1} \sum_{j=i}^n C_{ij} F_{ij} X_i - X_j$$

$$\text{제한식} \quad X_i - X_j > = 1/2(L_i + L_j) + D_{ij}, i=1,2,\dots,n \\ -1; j=1,2,\dots,n \quad X_i > = 0, i=1,2,\dots,n$$

여기서

- $n$  = 설비의 수(혹은 기계)
- $F_{ij}$  = 설비  $i$ 와  $j$ 사이의 물동량
- $C_{ij}$  = 설비  $i$ 와  $j$ 사이의 이동거리당 이동비용
- $L_i$  = 설비  $i$ 의 길이
- $D_{ij}$  = 설비  $i$ 와  $j$ 사이의 최소한의 여유 거리
- $X_i$  = 설비  $i$ 의 중간과 라인의 첫시작점 사이의 거리

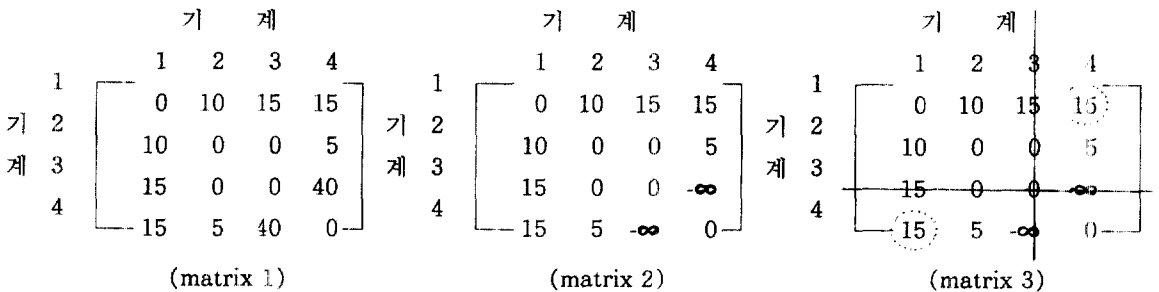
단일라인 문제를 매우 효과적으로 풀수 있는 발견적 알고리즘인 MST(Modified Spanning Tree)가 Heragu와 Kusiak(1988)[11]에 의해서 발표되었는데 EFLDS는 MST 알고리즘을 단일라인 순서 결정 알고리즘으로 채택하였다. 그러나 MST 알고리즘은 Heragu과 Kusiak이 의도한바의 결과를 가져오기 위해서는 알고리즘에 약간의 문제가 있는 것을 알수 있다. MST 알고리즘의 절차는 다음과 같다.

- (1) flow matrix  $F_{ij}$  에서,  
 $F_{i^*j^*} = \max\{F_{ij}; i=1,2,\dots,n, j=1,2,\dots,n\}$ 를 찾아낸다.  
 $i^*j^*$ 를 인접시키고 이를 부분해로 놓고  $F_{i^*j^*} = F_{i^*j^*} - \infty$ 로 놓는다.
- (2)  $F_{pk} = \max\{F_{i^*k}, F_{j^*k}; k=1,2,\dots,n, l=1,2,\dots,n\}$ 를 계산하고

- i)  $p$ 에  $q$ 를 인접시키고  $q$ 를 부분해에 추가한다.
  - ii) 행  $p$ 와 열  $q$ 를  $F_{ij}$ 에서 삭제한다.
  - iii) 만약  $p=i^*$ 이면  $i^*=q$ 로 놓고 그렇지 않으면  $j^*=q$ 로 놓는다.
- (3) 마지막 해가 구해질때 까지 (2)를 반복한다.

위 절차중 (2)부분이 빈약하다. 이는 Heragu와 Kusiak이 제시한 9개의 4machine배치문제 중 2번, 8번 문제를 풀어보면 알 수있다. 따라서 iii)뒤에  $F_{i^*j^*} = F_{i^*j^*} - \infty$ 을 추가하여야 한다. 왜냐하면 Heragu와 Kusiak이 제시한 4machine배치문제중 2번을 모델로한 경우 다음과 같은 문제점이 있기때문이다.

Heragu와 Kusiak이 제시한 4machine배치문제 2번의 flow matrix는 (matrix 1)과 같다. MST 절차(1)에 의해서  $F_{34}$ 는 찾고, 기계 3,4를 인접시킨다음  $F_{34} = F_{43} = -\infty$ 놓으면 flow matrix는 (matrix 2)와 같이된다. MST 절차(2)에 의해서  $F_{31}$ 이나  $F_{41}$ 를 선택하게 되는데  $F_{31}$ 을 선택 하였다고 하면 기계 3과 1를 인접시키면 부분해는 1,3,4가 되고 3행3열을 제외시키면 flow matrix는 (matrix 3)과 같이 된다. 절차(2)로 가서 계산하면  $\max\{F_{1k}, F_{4k}; k=1,2,\dots,n, i=1,2,\dots,n\}$ 를 구하면 다시  $F_{14}$ 와  $F_{41}$ 이 선택된다. 이는 이미 선택된 기계이기 때문에 선택될수 없다. 따라서  $F_{i^*j^*} = F_{i^*j^*} - \infty$ 를 절차(2)의 iii)뒤에 추가하여야 한다.



#### 4) 복수라인(Multi-Row) 투입순서결정 알고리즘

본연구의 복수라인 순서결정 알고리즘은 고정배치 설비를 고려할 수 있게 하였고 설비의 면적을 순서결정 순위가 같을 경우의 선택인자(select factor)로 고려하였다.

이는 면적이 큰설비가 실제 시스템에서 중요한 위치를 차지하고 있는 경우가 많을 뿐 아니라 면적이 큰설비를 먼저 배치하는것이 공간 활용에도 유리하기 때문이다. 본 알고리즘은 투입된 설비와 투입될 설비간에 이동빈도가 가장 큰 설비순서로 투입되므로 이상적인 설비위치를 알수만 있다면 이동비용을 최소로 할수 있다. 알고리즘은 다음과 같다.

- (1) 만약 고정설비  $i$ 가  $m$ 개 있으면 고정설비  $i$ 중에서 면적이 가장 큰 순서로 투입한다. ( $i=1,2,\dots,N$ ),  $m \leq N$

고정설비가 없으며 flow matrix  $F_{ij}$ 에서  $F_{ij}$ 가 가장 큰 설비  $i,j$ 중 면적이 큰순서로 투입한다.

선택된 행과열은  $F_{ij}$ 에서 삭제한다.

- (2) 투입된 설비들인  $i$ 와 남아있는 설비사이

에  $\sum_{i=1}^m F_{ij}$ 가 가장 큰 설비  $j$ 를 투입한다.

선택된  $j$ 행과  $j$ 열은  $F_{ij}$ 에서 삭제한다.

- (3) 투입순서가 모두 결정될때까지 (2)을 반복한다.

#### 5) 배치안 탐색(search)알고리즘

- (1) 제약지향적 추리(constraints-directed reasoning)에 의한 배치안 탐색

EFLDS는 다음과 같은 3가지 형태의 제약 정보를 고려하고 있다.

가) 설비인접에 관한 제약 :

각 설비간의 인접도를 A(absolutely necessary), O(ordinary), X(undesirable)의 3가지 형태로 분류하고, 기 배치된 설비와 배치될 설비와의 인접 특성이 A일 경우는 기 배치된 설비의 주위부터 공간을 탐색하고 O일 경우는 한단계 전에 배치된 설비의 주위부터 공간을 탐색해가며 X일 경우는 기 배치된 설비와 가장 먼곳에서 부터 탐색해 나간다.

나) 방향에 관한 제약 :

배치될 설비가 기 배치된 설비의 특정 방향에 위치한다는 제한이 있으면 방향제약이 주어진 공간 안에서만 배치가 이루어진다.

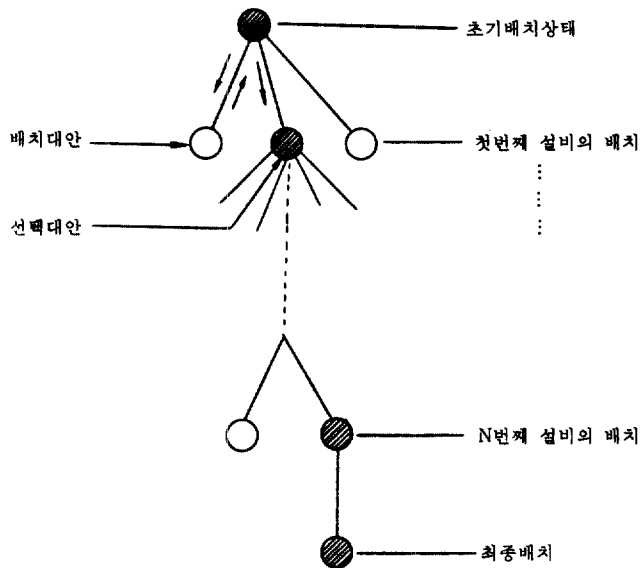
다) 설비 특성에 따른 배치 형태의 제약 :

설비의 특성상 설비의 형태를 변경시켜도 무방한 경우와 항상 주어진 형태로만 배치가 이루어져야하는 경우의 제약.

위와 같은 정보는 제약을 고려할 설비에 한해서만 입력하면 된다.

- (2) 계층적 탐색(hierachical search)에 의한 배치안 탐색

EFLDS는 (그림 3-3)와 같은 깊이 우선 탐색(depth first search)기법에 의한 배치과정을 전개 할수 있고, 각 노트(node)에서 사용자의 정의에 따른 배치 위치로 설비의 이동이 가능하며 그때마다 이동비용 평가치와 인접도 평가치를 평가해 볼수 있다. EFLDS의 배치안 탐색은 고정배치설비의 고려부터 시작되나 만약 고정배치설비가 없을 경우는 배치공간의 중앙에서부터 배치안탐색이 시작된다.



(그림 3-3) 깊이우선 탐색기법에 의한 배치안 탐색과정

### (3) 공간탐색 알고리즘

EFLDS의 공간 탐색 알고리즘은 나선구조와 직선구조를 갖고있다. 설비간의 인접도가 A(absolutely necessary), O(ordinary)일 경우는 나선구조에 의해서 탐색을 하고 인접도가 X(undesirable)인 경우는 인접대상이 되는설비와 가장 먼 곳에서 부터 직선구조 탐색을 한다.

터화면에서 요구하는 일정한 형태로 데이터를 입력한다. 만약 설비 A와 설비 B간의 인접관계가 매우 중요하고, 설비 A는 설비 B의 동쪽에 위치해야만 한다는 입력정보가 있다면 메뉴 화면에서 “control constraints”라는 메뉴를 선택하고 이에 대한 노트(note)화면이 나오면 (예 1)과 같은 형태로 입력하면된다. 이와 같은 정보는 (그림4-1)과 같은 프로그램 화면에서 얻을 수 있다.

(예 1) A B east A:

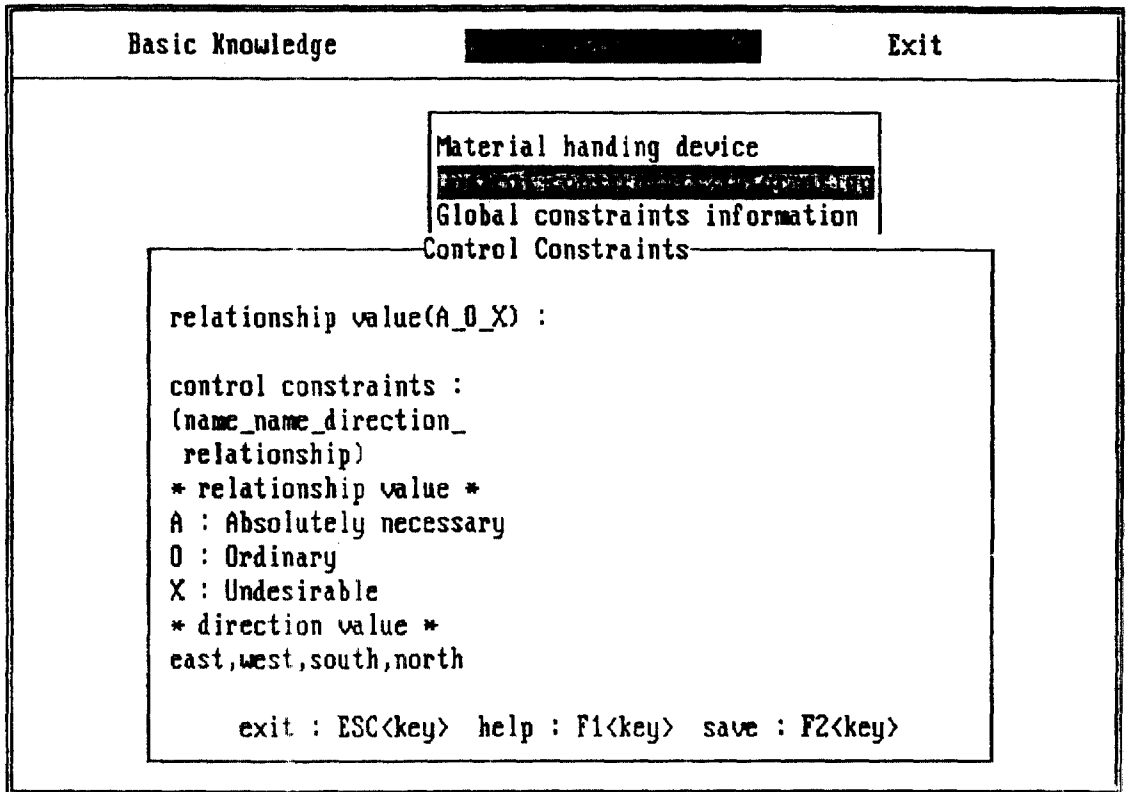
지식입력이 모두 끝나면 지식변환 프로그램에 의한 데이터변환, 투입순서 결정 프로그램에 의한 설비투입 순서결정들의 단계를 수행하고 배치 그래픽 화면을 통해 배치대안을 구할 수 있다. 모든 절차는 대화 형식으로 수행할 수 있다.

## 4. EFLDS의 적용 및 기존 알고리즘과의 비교분석

### 1) EFLDS 적용절차

EFLDS의 첫번째 작업은 입력정보를 토대로 (그림4-1)의 지식 입력 프로그램에 컴퓨터





W E L C O M E T O < E . F . L . D . S > S Y S T E M

(그림 4-1) EFLDS 입력화면

## 2) 기존알고리즘과의 비교분석

본 절에서는 EFLDS와 기존의 CRAFT와 CORELAP의 배치대안 결과를 비교해 보기로 하겠다.

CRAFT는 평가치로 이동비용만을 사용하고 있고 CORELAP은 인접도 평가치만을 고려하고 있기때문에 본 연구의 CRAFT에 대한 인접도 평가치와 CORELAP에 대한 이동비용 평가치는 EFLDS에서 사용하고 있는 평가치로 변환 계산하였다. EFLDS의 이동 비용 평

가치는 CRAFT와 같지만 인접도 평가치는 CORELAP과 같지는 않다. 그러나 쉽게 변환이 가능하므로 비교분석에 무리가 없을 것으로 판단된다. 또한 CORELAP은 고정설비를 고려할수 없기 때문에 본 비교분석에서는 고정설비를 고려하지 않았고 CRAFT의 초기배치안은 CORELAP에서 구한 배치안을 사용하였다. CRAFT와 CORELAP이 EFLDS에서 고려하고 있는 많은 정보들을 배치작업에 고려할수 없으므로 본 비교분석에서는 CRAFT와 CORELAP의 입력정보만을 고려하였다. 본 비

교평가의 의의는 EFLDS의 배치 알고리즘의 효율을 평가 해보기 위함이다. 비교분석을 위한 모델의 입력정보는 (표 4-1), (표 4-2), (표 4-3)와 같고 이에 대한 분석결과는 (표 4-4)와 같다.

(표 4-1) From-to Chart

|   |     |     |     |     |     |    |
|---|-----|-----|-----|-----|-----|----|
|   | 1   | 2   | 3   | 4   | 5   | 6  |
| 1 | 0   | 400 | 0   | 0   | 0   | 0  |
| 2 | 400 | 0   | 400 | 0   | 0   | 0  |
| 3 | 0   | 0   | 0   | 50  | 200 | 0  |
| 4 | 100 | 100 | 0   | 0   | 0   | 0  |
| 5 | 0   | 100 | 200 | 200 | 0   | 30 |
| 6 | 0   | 200 | 100 | 0   | 0   | 0  |

(표 4-2) 인접도

|   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|
|   | 1 | 2 | 3 | 4 | 5 | 6 |
| 1 | - | 0 | 0 | 0 | 0 | 0 |
| 2 | - | - | 0 | A | A | 0 |
| 3 | - | - | - | 0 | 0 | 0 |
| 4 | - | - | - | - | 0 | 0 |
| 5 | - | - | - | - | - | 0 |
| 6 | - | - | - | - | - | - |

(표 4-3) 배치될 설비의 형태

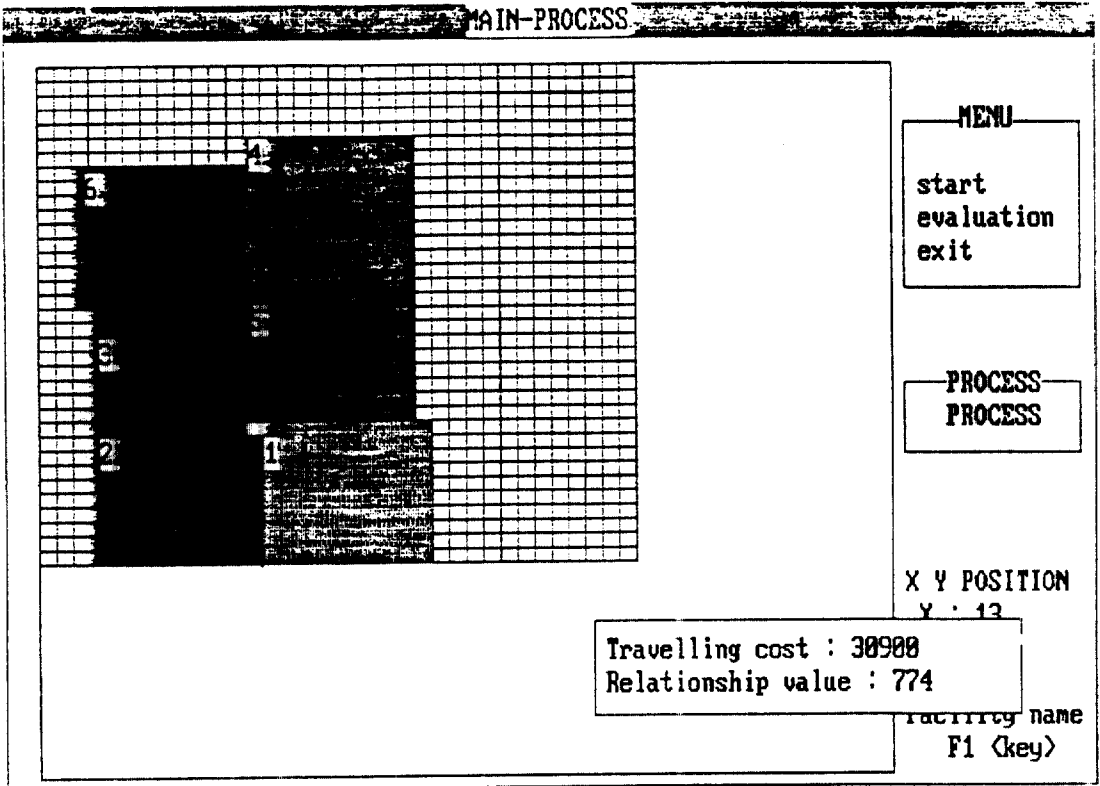
|       |    |    |   |    |    |    |
|-------|----|----|---|----|----|----|
| 설비 번호 | 1  | 2  | 3 | 4  | 5  | 6  |
| 가로 길이 | 10 | 10 | 9 | 10 | 10 | 10 |
| 세로 길이 | 10 | 10 | 9 | 10 | 10 | 10 |

(4-4) 비교 분석결과

| 알고리즘    | 이동 비용 평가치 | 최고치에 대한 효율 | 인접도 평가치 | 최고치에 대한 효율 |
|---------|-----------|------------|---------|------------|
| EFLDS   | 30900     | 96%        | 774     | 80%        |
| CRAFT   | 29600     | 100%       | 789     | 78%        |
| CORELAP | 35600     | 83%        | 617     | 100%       |

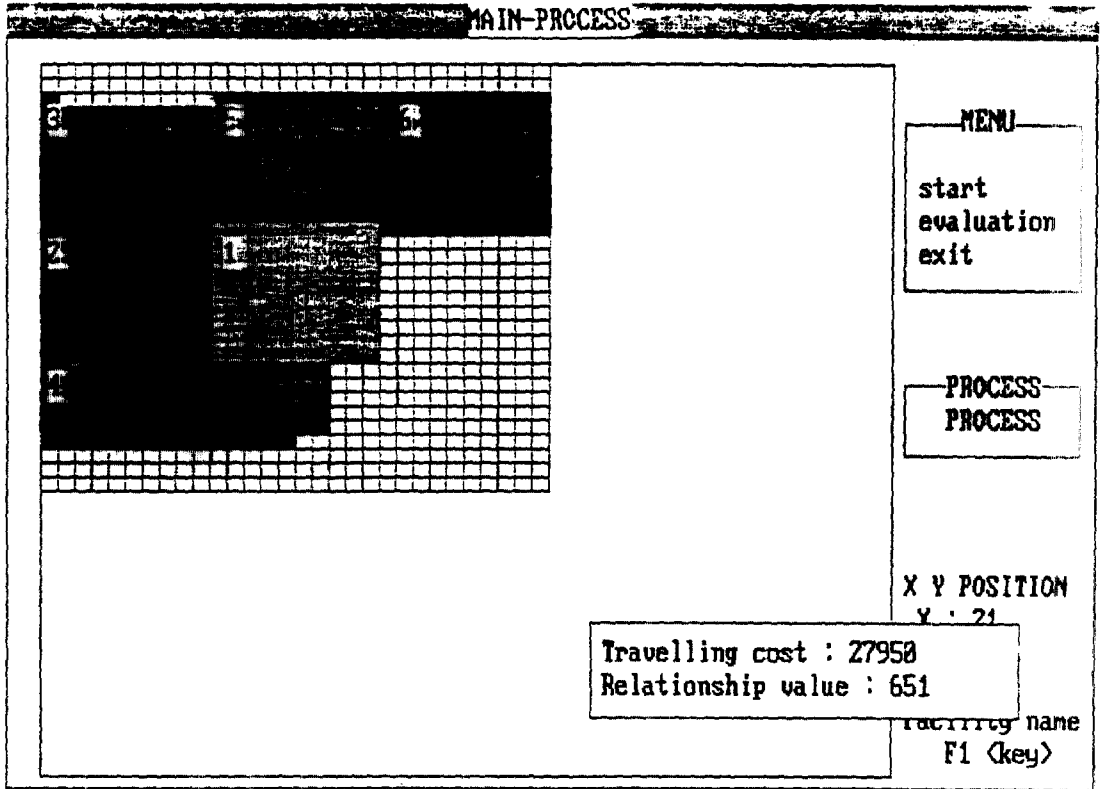
본 비교 분석의 EFLDS의 평가치는 (그림 4-2)와 같이 설비 형태를 고정형태로 놓고 구한 결과 이고, 설비 형태를 CRAFT나 CORELAP과 같이 변동시키고 이미 주어진 입력정보를 토대로 배치형태를 바꿔가면서 배치안을 구한 결과는 (그림 4-3)과 같고, 이동비용 평가치가 27950, 인접도 평가치가 651

인 아주 우수한 대안을 얻을 수 있었다. 본 비교 분석결과 EFLDS는 단순 평가척도에서도 우수한 대안을 생성시켜 줄 수 있음을 알 수 있다. 더구나 EFLDS는 기계배치와 같은 실질적 설비배치 환경까지 고려할 수 있으므로 배치 설계의 툴(tool)으로써 가치는 더욱 크다.



\*\* WELCOME TO E. F. L. D. S \*\*

(그림 4-2) EFLDS 배치결과화면 (최초안)



**\*\* WELCOME TO E. F. L. D. S \*\***

(그림 4-3) EFLDS 배치결과화면 (조정안)

## 5. 결 론

본 연구에서는 설비배치 문제의 해법에 전문가 시스템적 개념을 도입한 새로운 알고리즘을 개발하였다. 본 연구에서 개발된 알고리즘은 설비배치문제에 완벽한 해법을 줄수는 없지만, 설계자가 컴퓨터와 대화형식으로 배치안을 구성할 수 있으므로 배치계획에 유용하게 사용할 수 있고, 본 연구에서 논한 기존의 알고리즘들과의 비교 분석에서도 알 수 있듯이 효율적인 대안들을 생성해줄 수 있다.

또한 컴퓨터 프로그램을 범용 프로그램 언어인 C언어를 이용하여 퍼스널 컴퓨터에서 사용할 수 있게 개발함으로써 해법에 새로운 모델을 제시했다는데 그 의의가 있다. 그러나 본 연구에서는 운반설비에 대한 룰(rule)지식이 빈약하였고, 대안 선택에 대한 다양한 평가 기준이 부족하였다. 이부분에 대한 연구가 보완되어야 하겠고, 본 연구에서 개발된 알고리즘은 구성형(construction)알고리즘인데 이를 개선형(improvement)알고리즘과 연결시키는 연구도 시도되어야 할 것이다.