

# 시스템記述技法의 開發에 관한 研究 \*

(Development of a System Description Algorithm)

崔 仁 壽\*\*

姜 錫 昊\*\*\*

## Abstract

The purpose of this study is to provide a System Description Algorithm based on the need to describe the system more systematically than the System Dynamics does.

The algorithm, which comprises 14 steps, is described in detail and demonstrated through an example of the dynamic market networks.

## 1. 序 論

시스템分析에 있어서 항상 우선적으로 요구되고 있는 事項은 어떻게 하면 시스템을 간단하고, 농도있게, 또한 쉽게 理解시킬 수 있도록 記述할 수 있을까 하는 方法을 찾고자 하는 것이다.

시스템動態論(System Dynamics, 以下 SD로 略記)은 1950年代 후반부터 概念이 定立되고 開發되기 시작했는데[20], 이 때의 初期에 있어서는 주로 經營分野에 관해 研究가 이룩되었다[10].

그러나 실제로 SD가 알려지기 시작하고, 都市模型 및 地球模型과 같이 巨視的인 水準의 模型 研究가 시작된 것은 1960年代에 들어서고부터였다[1, 2, 4~9, 12, 13, 14]. 1970年代에 들어서서는 60年代의 巨視的 水準으로부터 탈피해 微視的 水準으로 活用되기 시작했으며 SD에 관한 哲學도 再定立되기 시작했다[3].

시스템圖(System diagram)가 잘 표현되면 模型作成者의 마음을 傳達받을 수 있고, 文語(written language)로서는 도저히 표현할 수 없는

\* 본 논문은 서울대학교 공과대학 박사학위 논문중의 일부임

\*\* 崇田大學校 産業工學科

\*\*\* 서울大學校 産業工學科

事項도 說明받을 수 있게 된다. SD에서 채택하고 있는 시스템圖는 影響圖(influence diagram)라고 제일 많이 불리고 있는데, 이러한 影響圖의 作成技法에 대해서 Wolstenholme [19,20]을 필두로 하여 많은 研究를 현재까지 수행하고 있다.

以下 시스템記述技法의 최근 진전사항인 Wolstenholme의 影響圖 作成 알고리즘을 段階別로 간략히 소개하면 다음과 같다.

- (1 段階) 問題와 관련되어 있는 主要變數를 確認.
- (2 段階) 主要變數와 관련된 初期의 시스템資源(system resource)의 確認.
- (3 段階) 시스템資源의 初期狀態(initial states) 確認.
- (4 段階) 各 資源의 各 狀態에 대한 物的 흐름 모듈(physical flow modules)의 作成.
- (5 段階) 4 段階에서 말하는 資源의 狀態를 둘 이상 취급할 때는 4 段階에서 作成한 各 狀態에 대한 物的 흐름 모듈을 直列化(cascade) 시킨다.
- (6 段階) 各 모듈(혹은 直列化 모듈) 內에 있어서의 作用(behaviour), 情報(information) 및 統制(control) 分野의 연결사항을 確認.
- (7 段階) 異種資源 모듈간의 作用, 情報 및 統制分野의 연결사항을 確認.
- (8 段階) 資源이나 資源의 狀態 중에서 새로 添加할 필요가 있을 경우에는 이를 1 段階 및 2 段階에서 確認한 主要變數 및 資源에 연결시킨다. 필요하다면 지금까지의 各 段階를 반복시킨다.
- (9 段階) 이렇게 作成한 影響圖를 再檢討한다.

본 研究에서는 前述한 바와 같은 최근에 이룩되고 있는 影響圖 作成技法이나 作成 알고리즘보다 더 体系的인 간단하게 影響圖를 作成할 수 있는 「시스템 記述 알고리즘(System Description Algorithm)」을 開發하는 것을 目的으로 하고 있

다.

二進從屬行列(binary subordination matrix)의 諸特性[15~18]을 活用하여 本 알고리즘을 개발하였는데, 먼저 알고리즘을 記述하고 다음으로 알고리즘 各 段階의 妥当性を 論한 다음, 事例研究을 하는 順으로 本 論文을 전개시키고자 한다.

## 2. 시스템記述 알고리즘

本章에서는 다음과 같은 14 段階로 構成된 시스템記述 알고리즘을 제시하고자 한다.

1~6의 段階는 시스템記述을 体系的으로 遂行하게끔 하는 基礎段階이며, 7~14의 段階는 시스템의 因果關係를 水準(level) 別로 分類하여 시스템의 分析 및 統制에 有效活用케 하는 階層分類圖 作成用 應用段階라고 할 수 있다.

### 1 段階

컨트롤러, 環境 및 컴플리먼트를 살펴보아 分析의 対象이 되는 시스템의 構成要素를 규명하고  $E = \{e_1, e_2, e_3, \dots, e_n\}$  를 시스템의 構成要素群으로 定義) E 중에서 2 個씩의 要素를 골라 이들간의 從屬關係를 규명하여 正方形行列 MF 를 作成한다.

### 2 段階

#### 2. 1

MF 에 있어서 行(혹은 列)의 칸이 모두 "0"으로 記入된 行(혹은 列)을 가려내어 이 該當要素의 行과 列(혹은 列과 行)을 MF 에서 除去시켜 正方形行列을 作成한다.

#### 2. 2

2. 1의 過程을 더 이상 除去시킬 行과 列이 없을 때까지 반복한다. 이 경우 殘存하게 되는 正方形行列을 PF 라 定義한다.

### 3 段階

#### 3. 1

PF의 1행에 해당되는 요소의上位要素를 確認한다. 예를 들어 i, j, k 列에 해당되는 요소가 1행에 해당되는 요소의上位要素가 된다고 한다면  $R^* = R_i + R_j + R_k$ 를 PF에서의  $R_i$  과 대체시킨다.

#### 3. 2

PF의 1列에 해당되는 요소의下位要素를 確認한다. 예를 들어 l, m, n 行에 해당되는 요소가 1列에 해당되는 요소의下位要素가 된다고 한다면  $R_l^* = R_l + R_1$ ,  $R_m^* = R_m + R_1$ ,  $R_n^* = R_n + R_1$ 을 각각 PF에서의  $R_l, R_m, R_n$ 과 대체시킨다.

PF의 모든各行 및 列에 대해서 3. 1 및 3. 2의 段階를 밟는다. 以上の 段階를 완전히 밟아서 얻게되는 正方向行列을 RPF라 定義한다.

#### 3. 3

$RPF = \{e_{ij}\}_{m \times n}$ 에서 모든 i, j에 대해  $e_{ij}$ 의 값을 조사하여 同時に 모든  $e_{ij} = e_{ji} = 1$ 을 만족시키는 요소의 群을 별도로 묶어 各各 피드백 블록(feedback block)으로 삼는다.

### 4 段階

MF에서의各構成要素를 살펴보자 3. 3 段階에서 확인한 피드백 블록에 속하게 되는 요소들을 하나의 요소로 간주하는 方法을 택함으로써 MF를 再作成한다. 이렇게 再作成한 MF를 M 正方向行列이라 定義한다.

### 5 段階

M의 構成要素중에서 임의로 選定한 하나의 構成要素  $e_r$ 을 基準으로 하여 M의 構成要素를 다

음과 같이 分類한다.

#### 5. 1

$e_r$ 로부터 到達可能한 즉  $e_r$ 를 下位 構成要素로 거느리고 있는 모든 構成要素의 群. 이를 上位群(Lift set)  $L(e_r)$ 라 定義한다.

#### 5. 2

$e_r$  및  $L(e_r)$ 을 除外한 M의 나머지 構成要素들을  $e_r$ 의 下位要素가 아닌 要素들로 構成되는 空百群(Vacancy set,  $V(e_r)$ )과  $e_r$ 의 下位要素가 되는 要素들로 構成되는 下位群(Drop set,  $D(e_r)$ )으로 分類한다.

### 6 段階

$L(e_r), V(e_r), D(e_r)$ 의 順으로 M 行列의 構成要素를 다음 그림과 같이 再整理하고 ①의 칸에는 모두 "1"로, ②, ③, ④의 칸에는 모두 "0"을 記入한다.

$M_{LL}$ 에서 더 以上 上位群, 空百群, 下位群등의 어느 하나까지도 分類할 수 없을 때까지 5 段階에서 以上の 6 段階에 이르는 M의 再整理 方法을 同一하게 적용시킨다.

		$L(e_r)$	$e_r$	$V(e_r)$	$D(e_r)$
			0		
			0		
$L(e_r)$		$M_{LL}$		②	③
			0		
	$e_r$	1 1 .....	1 0	0 0 ... 0	0 0 ... 0
			0		
			0		
$V(e_r)$		$M_{VL}$	⋮	$M_{VV}$	④
			0		
$D(e_r)$		①	⋮	$M_{DV}$	$M_{DD}$

(그림 1) M 行列의 分類

$M_{VV}$  및  $M_{DD}$  에서도  $M_{LL}$  과 同一한 方法을 적용시킨다. 이렇게 최종적으로 作成한 正方形行列을 從屬行列  $S$  라 定義한다. 단  $S$  의 作成時  $V(e_r) \rightarrow L(e_r)$  의 從屬關係는 手作業으로 규명해야 한다.

## 7 段階

$S$  行列의 行과 列을 살펴보아 그 行과 列의 各 칸이 모두 “0”으로 채워져 있는가를 확인한다. 最上水準群(行의 各 칸이 “0”으로 채워져 있는 要素의 群)과 基底水準群(列의 各 칸이 “0”으로 채워져 있는 要素의 群)에 同時に 포함되어 있는 構成要素가 있으면 이를  $S$  行列에서 除去 시킴으로써  $B$  正方形行列을 作成한다.

## 8 段階

$B$  行列의 最上水準群을 가지(branch) 別로 分類한다. 첫번째 가지에 해당되는 最上水準群의 要素를 중심으로하여 9 段階의 作業을 수행한다.

## 9 段階

### 9. 1

該當가지에 소속되는 要素의 列을 골라내어 이들 列의 부울합(Boolean sum)을 計算 함으로써 列벡터를 구한다.

### 9. 2

이 列벡터를  $B$  行列의 各 列과 곱하고, 여기서 行과 列이 모두 “0”으로 채워진 것을 除去시킴으로써  $C$  行列을 作成한다.

### 9. 3

該當되는 가지에 소속되는 要素를 記入함으로써 階層分類圖의 作成을 시작한다.

## 10 段階

$C$  行列의 行중에서 “1”이 記入된 것이 하나뿐인 行을 골라 다음과 같은 方法으로 이들 行을 除去시킨다.

### 10. 1

$i$  번 行에 있어서  $j$  번 列에 해당되는 칸만이 “1”로 記入되어 있다고 한다면  $C_j^* = C_j \bar{C}_i$  를 계산하여  $C_j$  대신에 이를 대체시킨다.  $C_j$  가 없는 경우는  $C_j^* = C_j$  로 한다.

### 10. 2

$C$  行列에서  $i$  번 行을 除去시키고 난다음, “0”으로만 채워져 있는 行과 列이 있다면 이를 除去시킨다.

### 10. 3

階層分類圖에  $e_i$  를 記入하고, 이를  $e_i$  에 從屬시킨다.

殘存  $C$  行列에서 “1”이 하나만 記入된 行이 發見되면 上記 10 段階 作業을 되풀이 하고 아니면 11 段階로 넘어간다.

## 11 段階

10 段階의 行 除去를 거친 行列에 대하여 다음과 같은 方法으로 列을 除去시킨다. 물론 “1”이 記入된 것이 하나뿐인 列을 대상으로 한다.

### 11. 1

$i$  번 行에 있어서  $j$  번 列에 해당되는 칸만이 “1”로 記入되어 있다고 한다면  $R_j^* = R_j \bar{R}_i$  를 계산하여  $R_j$  대신에 이를 대체시킨다.  $R_j$  가 없는 경우는  $R_j^* = R_j$  로 한다.

### 11. 2

行列에서 j번 행을 除去하고 난 다음, “0”으로만 채워져 있는 行과 列이 있다면 이를 除去시킨다.

### 11. 3

階層分類圖上에  $e_j$ 를 記入하고,  $e_i$ 를  $e_j$ 에 從屬시킨다.

以上の 列 除去가 可能할때까지 11段階의 作業을 되풀이하고 난 다음 12段階로 넘어간다.

### 12段階

“1”이 하나만 記入된 行도 列도 없는 行列의 경우는 10段階 및 11段階를 밟을 수 없게 된다. 이 경우는 “1”의 個數가 가장 작은 行을 골라 (k個의 “1”이 記入된 경우) 이를 k個의 “1”이 하나씩 記入된 行으로 分割시켜 10段階로 돌아간다.

### 13段階

9段階로 돌아가 두번째 以下 모든 가지에 해당하는 最上水準群의 要素에 대한 分析을 되풀이한다.

### 14段階

階層分類圖上에 탈락되어 있는 M行列의 從屬關係가 있다면 이를 추가시키고, 階層分類圖를 再點檢한다.

## 3. 「시스템記述 알고리즘」의 妥当性

本章에서는 2章에서 서술한 바 있는 「시스템記述 알고리즘」을 구성하고 있는 各段階의 妥当性에 대해서 기술하고자 한다.

### 3. 1 시스템의 構成要素群(1段階)

시스템의 内部構造를 살펴보면 시스템境界내에

포함되어 있는 다음과 같은 컨트롤러(controller), 環境(environment) 및 컴플리먼트(complement)로 구성되어 있다고 할 수 있다. [3]

컨트롤러는 認識可能하고 變化可能한 시스템의 主体를 말하고, 環境은 컨트롤러와 相互作用은 하지만 그 돌아가는 狀況(working)은 명백하게 分辨할 수 없는 것을 말하며, 컴플리먼트는

〈표 1〉 從屬關係

要 素	上 位 要 素
$e_1$	$e_2$
$e_2$	$e_3$
$e_3$	$e_4$
$e_4$	$e_5$
$e_5$	$e_6$
$e_6$	$e_7$
$e_7$	$e_8$
$e_8$	$e_9, e_{10}$
$e_9$	$e_4, e_{10}$
$e_{10}$	없 음

	$e_1$	$e_2$	$e_3$	$e_4$	$e_5$	$e_6$	$e_7$	$e_8$	$e_9$	$e_{10}$
$e_1$	0	1	0	0	0	0	0	0	0	0
$e_2$	0	0	0	0	1	0	0	0	0	0
$e_3$	0	0	0	1	0	0	0	0	0	0
$e_4$	0	0	0	0	1	0	0	0	0	0
$e_5$	0	0	0	0	0	1	0	0	0	0
$e_6$	0	0	0	0	0	0	0	1	0	0
$e_7$	0	0	0	0	0	1	0	0	0	0
$e_8$	0	0	0	0	1	0	0	0	1	1
$e_9$	0	0	0	1	0	0	0	0	0	1
$e_{10}$	0	0	0	0	0	0	0	0	0	0

〈그림 2〉 MF 正方形列

컨트롤러나 환경에 영향을 미칠수 있으면서도 반면에 이들 컨트롤러나 환경에 의해서는 영향을 받지 않는고, 컨트롤러나 환경에 속해있지 않는 其他의 것을 말한다.

따라서 시스템의 構成要素群  $E$  는 컨트롤러, 環境, 컴플리먼트에 관련되어 있는 모든 要素를 包括해야 한다.

예를 들어 10個의 構成要素로  $E$  가 構成되어 있다고, 즉  $E = \{e_1, e_2, e_3, \dots, e_{10}\}$  이라고 하고, 이들간의 從屬關係 [22] 는 <표 1> 과 같다고 하여  $MF$  (Matrix with Feedback Loops) 를 作成해보면 <그림 2> 와 같다.

### 3. 2 入出力 構成要素의 除去 (2 段階)

$E$  의 構成要素 중에서 入力 및 出力部分에 해당되는 要素를 除去시키기 위하여, 또한 피드백 루프에 해당되는 要素를 區分하기 위하여 2 段階가 필요하게 된다.

<그림 2>의  $MF$  에 대하여 2 段階의 內容을 遂行시켜  $PF$  (Partial Feedback Loops) 正方形行列을 作成해 보면 <그림 3> 과 같다.

	$e_1$	$e_2$	$e_3$	$e_4$	$e_5$
$e_1$	0	1	0	0	0
$e_2$	0	0	1	0	0
$e_3$	0	0	0	1	0
$e_4$	0	1	0	0	1
$e_5$	1	0	0	0	0

<그림 3> PF 正方形行列

### 3. 3 피드백 블록의 確認 (3 段階)

<그림 3>의  $PF$  正方形行列에 대해 3. 1 및 3. 2 의 段階를 수행하여  $RPF$  (Revised Partial Feedback loops) 行列을 作成해 보면 <그림 4> 와 같다.

	$e_1$	$e_2$	$e_3$	$e_4$	$e_5$
$e_1$	1	1	1	1	1
$e_2$	1	1	1	1	1
$e_3$	1	1	1	1	1
$e_4$	1	1	1	1	1
$e_5$	1	1	1	1	1

<그림 4> RPF 正方形行列

3. 1 및 3. 2 의 段階를 채택한 것은 피드백 루프를 가장 경제적으로 記述하기 위해서 임이며, 3. 3 段階를 채택한 것은 <그림 4> 의  $RPF$  중에서 어느 要素가 피드백 루프에 해당되는 가를 파악하기 위해서 이다. <그림 4> 를 3. 3 段階의 內容으로 살펴보면 다섯개의 構成要素 모두가 단 하나의 피드백 블록에 소속되게 됨을 알 수 있다.

### 3. 4 完全階層式 行列의 作成 (4 段階)

피드백 블록에 속하게 되는 構成要素를 하나의 構成要素로 간주하여 完全階層式 行列  $M$  을 作成하기 위해서 이 段階가 필요하게 된다. 4 段階를 수행시켜 作成한  $M$  行列은 <그림 5> 와 같다.

	$e_1$	$e_2$	$e_3$	FB1	$e_7$	$e_{10}$
$e_1$	0	1	0	0	0	0
$e_2$	0	0	0	1	0	0
$e_3$	0	0	0	1	0	0
FB1	0	0	0	0	0	1
$e_7$	0	0	0	1	0	0
$e_{10}$	0	0	0	0	0	0

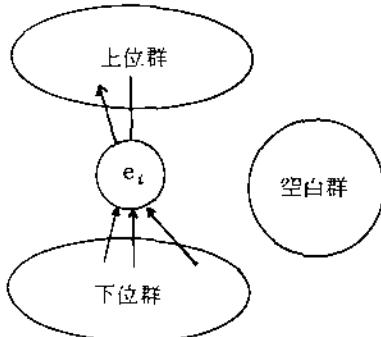
<그림 5>  $M$  行列

### 3.5 시스템構成要素의 分類(5段階)

수 많은 시스템構成要素가 서로 얽혀 복잡한 從屬關係를 이루고 있는 경우에 시스템 分析을 어느 構成要素부터 시작하여 遂行해야 하는가를 규명하는 것은 매우 어려운 일이다.

本 알고리즘에서는 시스템分析의 出發點에 제한을 두고있지 않다. 어떠한 構成要素든 임의로 하나의 構成要素  $e_i$ 를 골라 시스템分析을 시작하면 된다.

임의로 선정한 要素  $e_i$ 를 基準으로하여 <그림 6>과 같이  $e_i$ 의 上位群, 下位群, 空白群 등으로  $M$ 의 構成要素를 分類할 수 있음을 自明하다.



<그림 6> 從屬關係

註: 화살표의 화살이 향하고 있는 쪽에 속해있는 要素는 꼬리쪽에 속해있는 要素의 主人이 된다. 즉, 꼬리쪽의 要素가 화살쪽의 要素에 從屬(行列上에 1로 表記)되어 있다.

<그림 5>의  $M = \{e_i, e_2, e_3, FBI, e_7, e_{10}\}$  중에서 임의로  $e_i$ 를 選定하여 5段階를 遂行해 보면 다음과 같은 上位群, 下位群, 空白群을 얻게 된다(<표 2> 參照).

<표 2>  $e_i$ 를 基準으로 한 各 群의 分類

上位群, $L(e_i)$	FBI, $e_{10}$
下位群, $D(e_i)$	$e_7$
空白群, $V(e_i)$	$e_2, e_3$

### 3.6 從屬行列(Subordination matrix)의 作成(6段階)

$M = \{e_i, e_2, e_3, FBI, e_7, e_{10}\}$ 을  $L(e_i), e_2, V(e_i), D(e_i)$ 의 順으로 再整理하여  $M$ 行列을 作成해 보면 <그림 7>과 같다.

	FBI	$e_{10}$	$e_2$	$e_3$	$e_7$	$e_i$
FBI		1				
$e_{10}$					②	③
$e_2$	1	⑤				
$e_3$	1					
$e_7$	1					④
$e_i$	①	1				

<그림 7> 새로운  $M$ 行列

이제부터는 <그림 7>의 새로운  $M$ 行列에 移行性(transitivity) 概念을 導入시키려고 한다. <그림 6>을 例로 들면,  $e_i$ 가 上位群에 속한 모든 要素에 從屬되어 있고, 반면에 下位群에 속해있는 모든 要素는  $e_i$ 에 從屬되어 있음을 알 수 있다. 따라서 下位群에 속해있는 모든 要素는 上位群에 속해있는 모든 要素에 從屬된다고 말할 수 있다는 것이 바로 移行性이다. 이러한 移行性을 <그림 7>에 導入하자면 <그림 7>의 ①에 해당되는 칸은 모두 "1"로 記入하고, ②, ③, ④의 칸에는 모두 "0"을 記入하면 되는데, 이 理由는 다모음과 같다.

1)  $D(e_i)$ 의 모든 構成要素는  $L(e_i)$ 의 모든 要素에 從屬된다. 따라서 ①의 칸에는 모두 "1"이 記入되어야 한다.

2)  $L(e_i)$ 의 어떠한 構成要素라도 이는  $V(e_i)$ 에 從屬될 수 없다. 따라서 ②의 칸에는 모두 "0"이 記入되어야 한다.

3)  $L(e_i)$ 의 어떠한 構成要素라도 이는  $D(e_i)$ 의 어떠한 構成要素에도 從屬될 수 없다. 따라서 ③의 칸에는 모두 "0"이 記入되어야 한다.

4)  $V(e_r)$ 의 어떠한 構成要素라도 이는  $D(e_r)$ 의 어떠한 構成要素에도 從屬될 수 없다. 따라서 ④의 칸에는 모두 "0"이 記入되어야만 한다.

마지막으로 ⑤의 칸에는 모두 "1"이 記入되어야 함은 분명하다.

이와같은 移行性概念을 적용시키는 6段階의 作業을 例로 들은 <그림 5>의 **M**行列에 대해서 最終遂行한 **S**(Subordination) 行列은 <그림 8>과 같다.

	$e_{10}$	FB1	$e_1$	$e_2$	$e_3$	$e_4$
$e_{10}$	0	0	0	0	0	0
FB1	1	0	0	0	0	0
$e_1$	1	1	0	0	0	0
$e_2$	1	1	0	0	0	0
$e_3$	1	1	0	0	0	0
$e_4$	1	1	1	0	0	0

<그림 8> S行列

### 3.7 가지(Branch) 式 行列의 作成 (7段階)

<그림 8>에서 行이 모두 "0"으로 채워져 있는 것을 보면 1行 즉  $e_{10}$ 이 있음을 알 수 있고, 모두 "0"으로 채워진 列을 살펴보면 4, 5, 6列 즉  $e_3, e_2, e_1$ 이 있음을 알 수 있다. 따라서  $e_{10}$ 이 最上水準群에 속해있는 構成要素가 되고,  $e_1, e_2, e_3$ 이 基底水準群에 속해있는 構成要素들이라고 할 수 있다. 同時에 最上水準群과 基底水準群에 속하게 되는 構成要素 즉 고립된 要素(isolated element)는 없기 때문에 <그림 8>의 S行列 자체가 바로 가지式 行列 B가 된다. B行列을 作成하는 理由는 行列의 構成要素 全部가 일련의 가지로 서로 연결되어 있음을 나타내기 위함이다.

### 3.8 最上水準群의 가지별 分離 (8段階)

<그림 8>의 **B**(=S) 行列의 例에서는 最上水準群에 속하는 要素가  $e_{10}$  하나뿐이기 때문에 별

문제가 없지만, 일반적으로 最上水準群에 속하는 要素들이 基底水準群에 속하는 어떤 構成要素들과 서로 가지로 연결되어 있는가를 확인해야 하는데, 이를 위해서 8段階를 設定한 것이다.

例로 들은 **B**行列의 基底水準群에 속하는  $e_1, e_2, e_3$ 을 살펴보면 이들 모두 最上水準群의 構成要素  $e_{10}$ 에 從屬되어 있음을 알 수 있다. 즉,  $e_{10} \leftarrow \text{FB1} \leftarrow e_2 \leftarrow e_1$  여기서 "0"으로만 記入된 行列을 除去함으로써 <그림 9>와 같은 **C**(Chained) 行列을 얻게 된다. 또한 9.3段階를 遂行하여 <그림 10>과 같이 階層分類圖의 作成

$$\begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 \end{bmatrix} \begin{matrix} e_{10} \\ \text{FB1} \\ e_2 \\ e_3 \\ e_1 \\ e_1 \end{matrix}$$

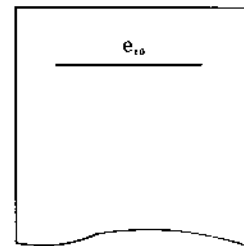
을 (基底水準群  $e_1$ 의 경우) 이고,  $e_{10} \leftarrow \text{FB1} \leftarrow e_2$  ( $e_2$ 의 경우)이며,  $e_{10} \leftarrow \text{FB1} \leftarrow e_3$  ( $e_3$ 의 경우)이기 때문이다. 따라서 이 경우, 한가지 가지로만 最上水準群을 分離하는 것이 가능하다.

### 3.9 單一階層 從屬行列의 作成 (9段階)

最上水準群의 첫가지에 해당되는 要素  $e_{10}$ 의 列 벡터  $e_{10} = [0, 1, 1, 1, 1, 1]^T$ 과 **B**의 Boole 積 구해보면 다음과 같다.

	$e_{10}$	FB1	$e_2$
FB1	1	0	0
$e_2$	1	1	0
$e_3$	1	1	0
$e_1$	1	1	0
$e_1$	1	1	1

<그림 9> C行列



<그림 10> 階層分類圖 (1)



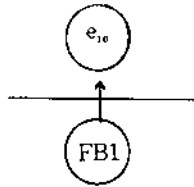
시작한다.

### 3.10 階層分類圖의 作成 (10~13段階)

〈그림 9〉의 C行列을 볼 것 같으면 “1”이 하나만 記入된 行이 FB1에 해당되는 行이 있음을 알게 된다. 따라서 10段階의 FB1行 除去 作業을 遂行할 수 있게 되는데, 遂行結果는 〈그림 11〉 및 〈그림 12〉와 같다.

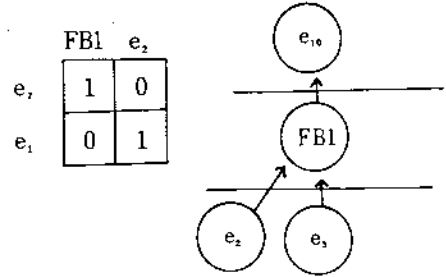
	FB1	e <sub>2</sub>
e <sub>2</sub>		
e <sub>3</sub>		
e <sub>7</sub>		
e <sub>1</sub>		

〈그림 11〉 C(1)行列

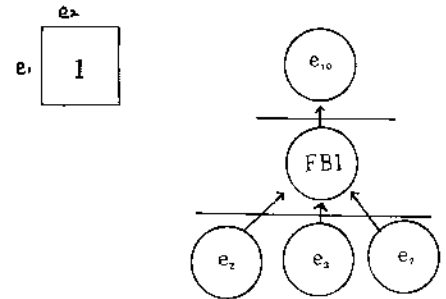


〈그림 12〉 階層分類圖(2)

〈그림 11의 C(1) 行列을 볼 것 같으면 “1”이 하나만 記入된 行이 e<sub>2</sub>, e<sub>1</sub>, e<sub>7</sub>이 있음을 알 수 있다. 〈그림 13〉은 e<sub>2</sub>, e<sub>3</sub>, e<sub>7</sub>行을 차례차례 除去시키고 난 다음에 남게 되는 C行列 및 階層分類圖의 狀態를 나타낸 것이다.



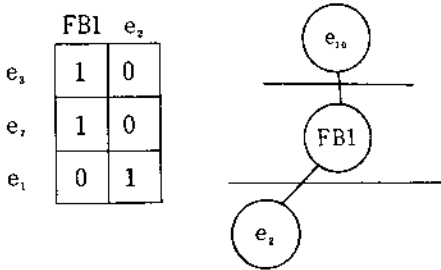
(b) e<sub>2</sub> 行 除去後



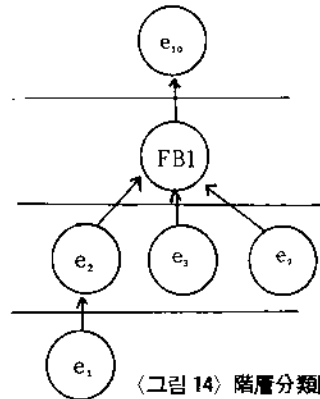
(c) e<sub>7</sub> 行 除去後

〈그림 13〉 行의 除去作業

〈그림 13(C)〉를 바탕으로 최종적으로 e<sub>1</sub> 行을 除去시키면 〈그림 14〉와 같은 最終階層分類圖를 얻게 된다.



(a) e<sub>1</sub> 行 除去後



〈그림 14〉 階層分類圖(完)

때로는 行除去가 불가능하면 列除去作業(11段階)을 해야하고, 이것도 불가능을 경우는 行의 分割作業(12段階)을 遂行해야 한다. 이렇게 行除去, 列分割 등을 거치면 첫번째 가지에 해당되는 最上水準群의 要素에 대한 分析이 끝난 셈이다. 以後 13段階로 넘어가 두번째 以下 모든 가지에 해당되는 最上水準群의 要素에 대한 分析을 되풀이 하면 된다. 本例에서는 行의 除去만으로 모든 作業이 끝나고, 또한 最上水準群에서 分離된 가지수도 하나뿐이기 때문에 11~13段階를 설명할 기회가 없었음을 밝혀둔다.

### 3.11 階層分類圖의 最終點檢(14段階)

n개의 構成要素의 從屬關係가 가장 복잡한 경우는 <그림 15(a)>라고 생각한다.

本 알고리즘에 있어서는 大體 시스템인 경우를 가장하여 가장 복잡한 從屬關係에 있는 <그림 15(a)>를 리던던시가 최대로 줄어있는 <그림 15(b)>로 변환시키고 있다. 그러나 경우에 따라서 꼭 필요한 리던던시도 존재할 경우가 있다. 예를 들어 <그림 15(a)> ③→①이 필수적이라 할 것 같으면 <그림 15(b)>에서는 ③→①의 리던던시가 삽입되어야 한다는 뜻이다. 즉 <그림 15(c)>로 명시된 影響圖을 얻을 수 있어야 한다는 뜻이다. <그림 15(c)>에서와 같이 꼭 필요한 리던던시를 階層分類圖上에 첨가시키고자 하는 의도하에서 14段階를 本 알고리즘에서 채택한 것이다.

## 4. 事例研究

本章에서는 C 絹織物会社를 対象事例(211)로 삼아 이 會社의 生産-在庫시스템을 2章에서 제시한 바 있는 「시스템記述 알고리즘」을 活用하여 記述하고자 한다.

### 4.1 C 絹織物会社의 生産-在庫 시스템

우선 이 會社의 生産-在庫시스템 模型作成에 관련된 主要事項을 分野別로 記述하고자 한다.

#### 4.1.1 驅動力 分野

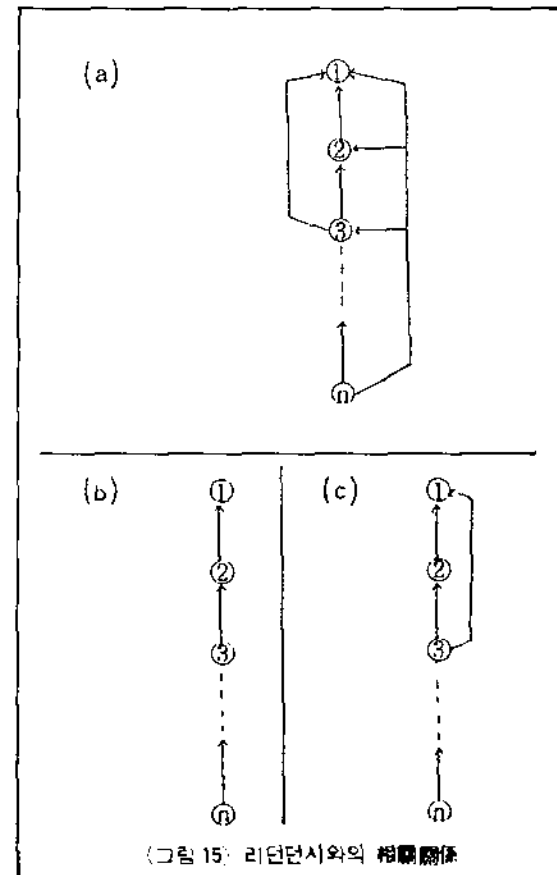
本 模型에 있어서 驅動力은 生産完製品에 대한 顧客의 需要活動이라고 볼 수 있다. 그런데 顧客의 需要活動은 다음과 같은 여러가지 要因에 의해 결정된다.

i) 生産製品의 價格, 品質, 구매가능도 및 自社의 在庫활동과 같은 컨트롤러의 統制下에 있는 要因들

ii) 競争社의 판매유요도와 같은 環境에 서의 變化要因들

iii) 정부와 규제등과 같은 컴플리먼트가 영향을 미치는 요인들

이러한 上記 要因들에 의해 生産製品에 대한 顧客의 需要率이 決定되고, 이와같은 需要情報은 各 中間商에게 전달된다.



#### 4.1.2 中間商 分野

顧客의 需要(ORCT)는 製品價格(PE 1), 製品의 시장점유도에 의해 결정되고 ORCT에 따라서 中間商은 자기의 目的值(OVT)와 견주어보아 工場倉庫로의 注文量決定(DMT)을 하게 된다. 이 DMT는 약간의 時間이 지난 후 실행(DMCT)된다. DMCT後 어느정도의 지연뒤에 製品이 工場倉庫로부터 도착(AFRW)하고 결과적으로 中間商의 在庫가 늘어나게 된다.

#### 4.1.3 工場倉庫 分野

DMCT에 따라 工場倉庫의 目的值(OVFW)와 工場倉庫의 完製品在庫(INVFW)를 견주어 工場倉庫에서의 注文量決定(DMFW)이 이루어진다.

이 DMFW는 약간의 시간경과後 実行(DMCFW)된다. 以後 어느정도 시간이 지나면 生産完了(PFR)에 따라 INVFW와 工場倉庫의 총 공급수량(ASCFW)은 증가하게 된다. 이때 工場倉庫의 出荷價格(CPFW)에 의해 工場倉庫의 出荷計劃(TRP)이 성립되고 DMCT와 TRP를 견주어 보아 工場倉庫의 出荷率(TRTFW)이 결정된다. 또한 TRTFW에 의해 새로운 CPFW도 결정된다.

#### 4.1.4 生産工場 分野

DMCFW와 出荷價格에 따른 工場生産率(PRC-P)을 비교함으로써 平均生産率(APRF)이 결정되고, 工場目的值(OVF)와 INVFW에 의해 生産開始率(PSR)과 原材料購買率(MPR)이 결정된다.

#### 4.1.5 原材料工場 分野

MPR에 의해 原材料工場의 注文殘高(OBLMF)가 발생하고, 原材料工場의 目的值(OVMP)와 原材料工場의 生産意思決定(DMMF)이 일어난다.

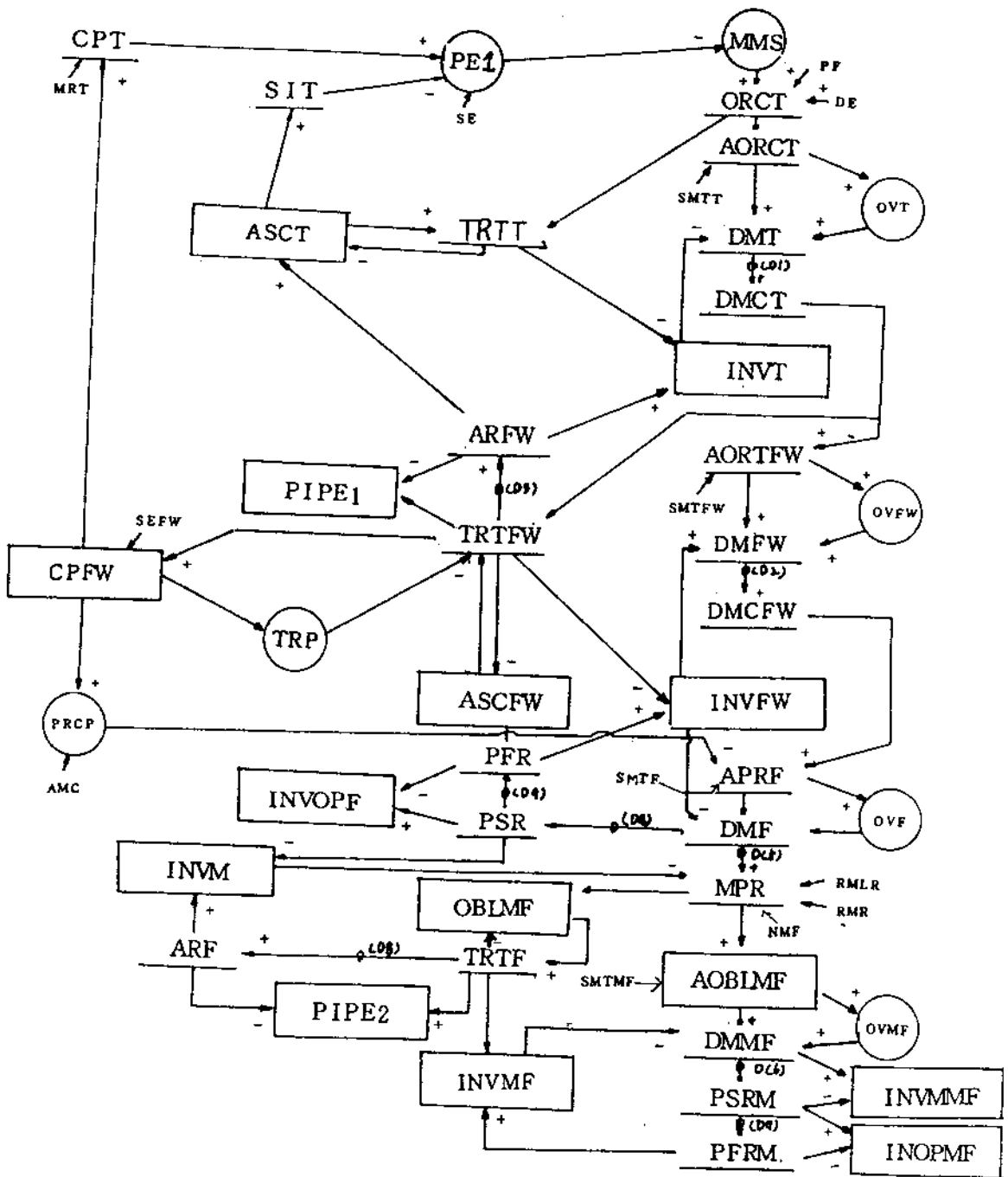
#### 4.1.6 製品價格 分野

CPFW에 의해 中間商의 出庫價(CPT)가 결정되며, 製品價格(PE 1)은 中間商의 공급 충분(SIT)에 의한 價格變動과 CPT에 의해 결정된다.

### 4.2 事例시스템의 階層分類圖 作成

事例시스템에 대해 4.1에서 主要分野만 선택하여 記述한 바 있으나, 本節에서는 總66個의 構成要素로 구성된 全体 시스템을 <그림16>과 같은 影響圖로 표시하였다. <그림16>을 볼 것 같으면 ○, —, □ 에 要素를 나타내는 度數의 略호가 記入되어 있는데, 여기서 ○에 記入된 要素는 補助率(auxiliary), — 上에 記入된 要素는 率(rate)을, □ 內에 記入된 要素는 水準(level)을 나타내어 DYNAMO[10, 11]에서 사용하는 原則과 一致시키게 하기 위해서임을 밝혀 둔다.

<그림 16>같이 從屬關係를 이루고 있는 시스템을 本稿에서 제안하고 있는 「시스템記述 알고리즘」을 活用하여 階層分類圖를 作成해 보면 <그림 17>과 같다. 但, <그림 16>의 構成要素 略記와 <그림 17>의 番號사이의 相關關係는 <표 3>과 같다.



〈그림 16〉 事例시스템의 影響圖

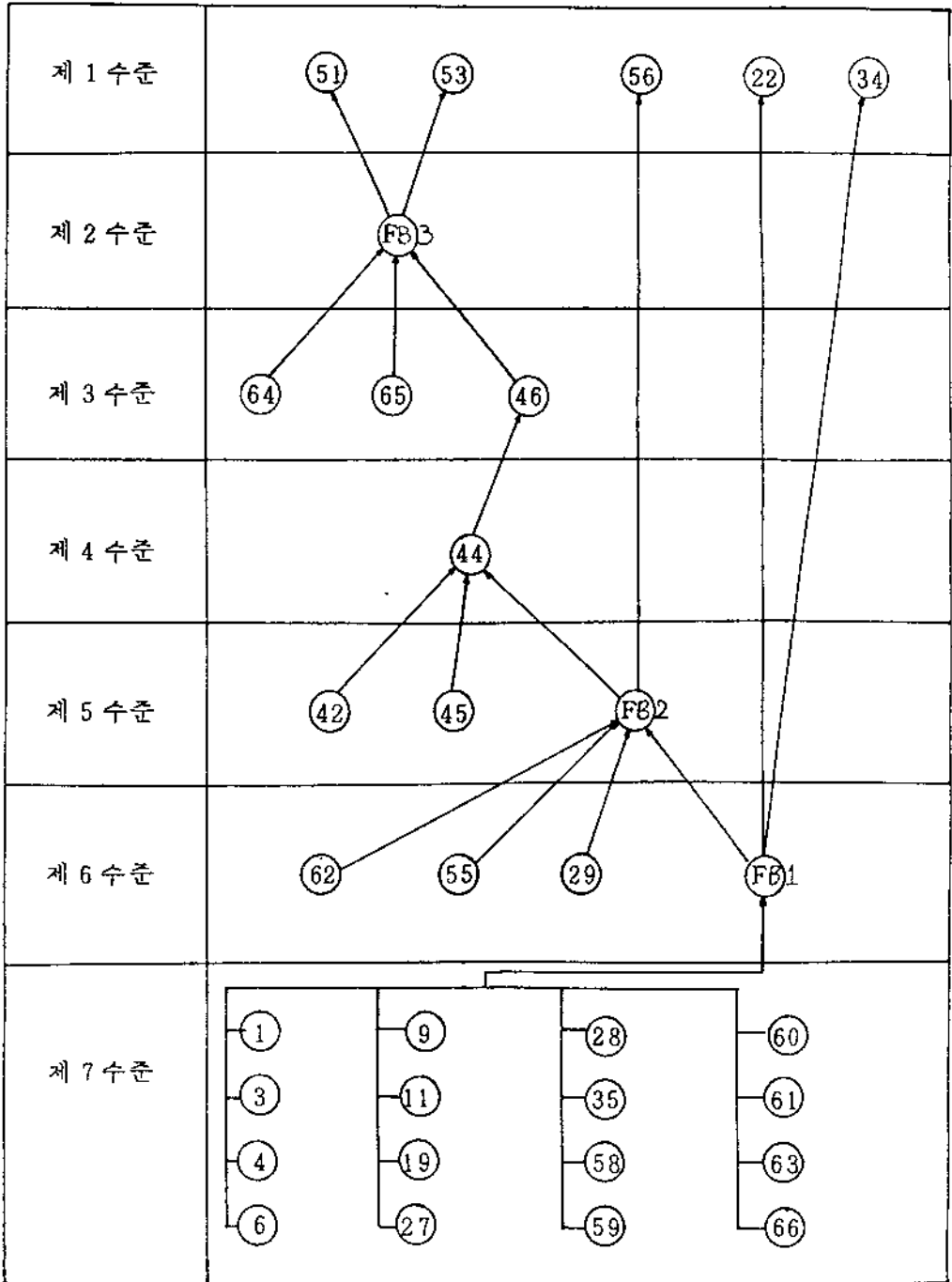


그림 17. 事例시스템의 階層分類圖

〈丑3〉構成要素番號 및 略記

수 준	번 호	
1	22	PIPE1(Pipeline Inventory between Factory Warehouse and Trader)
	34	INVOF(Inventory On-Process at Factory)
	51	INVMF(Inventory of Raw Materials at Raw Material Factory)
	53	INOPMF(Inventory On-Process at Raw Material Factory)
	56	PIPE2(Pipeline Inventory between Raw Material Factory and Factory)
2	FB3	이하 FB3의 구성변수임
	48	DMMF(Decision Making at MF)
	49	INVMF(INVEntory of MF)
	52	PSRM(Production Start Rate of MF)
	54	PFRM(Production Finish Rate of MF)
3	46	OVMF(Object Value of MF)
	64	D6 (Delay Time at MF)
	65	D7 (Time for Production)
4	44	AOBLMF(Average Order Backlog of MF)
5	42	NMF(Number of MF)
	45	SMTMF(Smoothing Time at MF)
	FB2	이하 FB2의 구성 변수임
	41	MPR(Material Purchasing Rate of Factory)
	43	OBLMF(Order Backlog of MF)
	47	INVM(Inventory of Raw Material at F)
	50	ARF(Arriving Rate from F)
	57	TRTF (Transshipment Rate to F from MF)
6	62	D8 (Time Requirement for Transportation from MF to F)
	55	RMLR (Raw Material Loss Rate)
	29	RMRR(Raw Material Requirement Rate Per 1 SYD)
	FB1	이하 FB1의 구성 변수임
	2	ORCT (Order Rate from Customer to Trader)
	5	PE1(Price Effect)
	7	SIT (Supply Increasment by Trader)
	8	CPT (Clearance Price of Trader)
	10	AORCT (Average Order Rate from Customer to Trader)
	12	OVT (Object Value of Traders)
	13	DMF (Decision Making of Trader)
	14	TRTT (Transshipment Rate to Trader from FM)
	15	ASCT (Aggregate Supply Capacity of Trader)

16	DMCT (Decision Making Committed by Trader)
17	INVT (INVENTORY of Trader)
18	ARFW (Arriving Rate from Factory Warehouse)
20	AORTFW (Average Order Rate from Trader to FW)
21	OVFW (Object Value of FW)
23	DMFW (Decision Making at FW)
24	CPFWE (Clearance Price at FW)
25	TRTFW (Transshipment Rate to FW from F)
26	DMCFW (Decision Making Committed at FW)
30	TRF (Transshipment Rate by Price)
31	INVEFW (INVENTORY of FW)
32	ASCFW (Aggregate Supply Capacity of FW)
33	PFR (Production Finish Rate)
36	APRF (Average Production Rate of F)
37	OVF (Object Value of F)
38	PSR (Production Start Rate)
39	DMF (Decision Making at F)
40	PRCP (Production Rate by Clearance Price)
7	1 MMS (Motivated Marketing Size)
	3 PF (Proportion of F)
	4 DE (Demand Elasticity)
	6 SE (Supply Elasticity)
	9 MRT (Margin Rate of Trader)
	11 SMTT (Smoothing Time at Trader)
	19 SMTFW (Smoothing Time at Fw)
	27 AMC (Average Manufacturing Cost)
	28 SEFW (Supply Elasticity at FW)
	35 SMTF (Smoothing Time at F)
	58 D1 (Delay Time at Trader)
	59 D2 (Delay Time at FW)
	60 D3 (Time Requirement for Transportation from FW to Trader)
	61 D4 (Time for Raw Material Purchasing)
	63 D5 (Delay at F)
	66 D9 (Time for production)

## 5. 結 論

本 研究에서 提案한 「시스템記述알고리즘」을 시스템記述(System description) 및 시스템分析에 活用해 본 결과 다음과 같은 結論을 얻을 수 있었다.

1) 시스템記述이 용이해 진다. 왜냐하면 어떠한 兩變數間의 從屬關係만 알더라도 시스템記述을 시작할 수 있기 때문이다.

2) 시스템設術이 体系的으로 된다. 왜냐하면 意思決定過程을 확실히 알려주는 M行列을 얻을 수 있기 때문이다.

3) 시스템記述用 資料入力の 生産성을 높일 수 있고, 이에 따르는 時間과 努力을 줄일 수 있

다(알고리즘의 6 段階 參照).

4) 意思決定過程을 손쉽게 확인할 수 있다. 왜냐하면 意思決定過程을 한 눈에 볼 수 있는 階層分類圖를 얻을 수 있기 때문이다.

5) 시스템分析을 용이하게 할 수 있다. 왜냐하면 시스템의 動的 特性을 거의 전적으로 좌우 하다시피 하는 피드백 루프를 별도로 分類하기 때문이다(알고리즘의 3 段階).

그러나 S行列의 作成時 아직까지 手作業을 해야 할 것은 本 알고리즘에서는 要求하고 있다(6 段階 參照). 手作業을 自動으로 전환시킬 수 있는 알고리즘의 開發이 追後の 研究方向으로서 關 要하다고 생각한다.

## References

1. ANSOFF, H.I. and SLEVIN, D.P., "An Appreciation of Industrial Dynamics", *Management Science*, Vol. 14, No. 7, pp. 383-397, 1968.
2. CALSON, B.R., "An Industrialist Views Industrial Dynamics", *Industrial Management Review*, pp. 15-20, 1964.
3. COYLE, R.G., *Management System Dynamics*, Wiley, Chichester, 1977.
4. FEY, W.R., "An Industrial Dynamics Case Study", *Industrial Management Review*, pp. 79-99, 1962.
5. FORRESTER, J.W., *World Dynamics*, Wright-Allen Press, Cambridge, 1971.
6. \_\_\_\_\_, *Urban Dynamics*, MIT Press, Cambridge, 1969.
7. \_\_\_\_\_, "Industrial Dynamics-A Response to Ansoff and Slevin", *Management Science*, Vol. 14, No. 9, pp. 601-618, 1968.
8. \_\_\_\_\_, "Industrial Dynamics-After the First Decade", *Management Science*, Vol. 14, No. 7, pp. 398-415, 1968.
9. \_\_\_\_\_, "Market Growth as Influenced by Capital Investment", *Industrial Management Review*, pp. 83-105, 1968.
10. \_\_\_\_\_, *Industrial Dynamics*, The M.I.T. Press: Cambridge, 1961.
11. PUCH, A.L., *DYNAMO User's Manual*, 5th ed., MIT Press, Cambridge, 1977.
12. ROBERTS, E.B., "New Directions in Industrial Dynamics", *Industrial Management Review*, pp. 5-14, 1964.
13. \_\_\_\_\_, "Industrial Dynamics and the Design of Management Control System", *Management Technology*, Vol. 3, No. 2, pp. 100-118, 1963.
14. SCHLAGER, K.J., "How Managers Use Industrial Dynamics", *Industrial Management Review*, pp. 21-29, 1964.



15. WARFIELD, J.N., "Toward Interpretation of Complex Structural Models", IEEE Transactions on Systems, Man, and Cybernetics, Vol. SMC-4, No. 5, pp. 405-417, 1974.
16. \_\_\_\_\_, "Developing Subsystem Matrices in Structural Modeling", IEEE Transactions on Systems, Man, and Cybernetics, Vol. SMC-4, No. 1, pp. 74-80, 1974.
17. \_\_\_\_\_, "Developing Interconnection Matrices in Structural Modeling", IEEE Transactions on Systems, Man, and Cybernetics, Vol. SMC-4, No. 1, pp. 81-87, 1974.
18. \_\_\_\_\_, " Binary Matrices in System Modeling", IEEE Transactions on Systems, Man, and Cybernetics, Vol. SMC-3, No. 5, pp. 441-449, 1973.
19. WOLSTENHOLME, E.F., "The Development of System Dynamics as a Methodology for System Description and Qualitative Analysis", Journal of the Operational Research Society, Vol. 34, No. 7, pp. 569-581, 1983.
20. \_\_\_\_\_, "System Description in Perspective", Journal of the Operational Research Society, Vol. 33, No. 6, pp. 547-556, 1982.
21. 손 민 수, 購買独占下에서의 市場動態에 관한 시스템 시뮬레이션 研究, 崇田大学校 大学院 碩士學位論文(1985. 8.)
22. 최 인 수, 시스템記述技法의 開發 및 시스템 分析과 統制에 대한 活用研究, 서울大学校 大学院 博士學位論文(1985. 2.)