研 究

コンパウンド・シーケンスの設定*

—生産システムの研究(第3報)—

岡村 健二郎** 山 品 元**

Establishment of Compound Sequences

—Study on Production System (3rd Report)—

Kenjiro OKAMURA, Hajime YAMASHINA

This paper deals with the problem of establishing a method for compound sequences. Compound sequences were considered by first constructing work element groups, and this approach was found to be advantageous from the point of view of being able to handle various technological restrictions and conditions otherwise difficult to formulate, to say nothing of precedence relations. It is also meritorious in that appreciable reduction in calculation times necessary for determining an optimum sequence was possible.

From the viewpoint of the above method, elemental union was introduced for constructing a feasible work element group, and to prevent the overlapping of a set of these groups the combinatorial matrix approach was tried. A table of these sets was drawn up so as to simplify the preparation of compound sequences. Moreover consideration of the problem of the best arrangement for overlap type compound sequences resulted in the formulation of several effective results. Then compound product of work element groups was introduced for the establishment of compound sequences. This leads to easy, systematic and mechanical establishment of compound sequences.

1. 緒 言

近年、企業間競争が激しくなるにしたがい、経営の合理化と生産性の向上を目的とする各種ソフトウェアの開発が強く要求されている。本研究もこうした認識に立ち、機械工業において遭遇する各種の生産システムについて、主にシーケンス設定の立場から、その最適化ならびに管理のための方策を見出そうとするものである。機械工業においては、生産性の良いことを一つの大きな目的としており、実施シーケンスのいかんによって生産性が大幅に影響を受けることから、この問題のもつ意味は重大である。

さて、シーケンスを作業の行ない方に着目して分類すれば、基本的にリニア・シーケンスとコンパウンド・シーケンスの二つに大別できる。第1報¹⁾ではリニア・シーケンスの組織的な設定法について論じ、第2報²⁾ではその応用として遷移値最小問題を取り扱った。本報の目的は引き続いてコンパウンド・シーケンスの設定法を考察することにある。

コンパウンド・シーケンスとは同時にいくつかの要素作業を遂行していく場合のシーケンスのことをいい、仕事の処理手順の高度の能率化に伴って問題となってくるものである。組立てラインやトランスファ・マシンなどにおけるシーケンスはその好例である。このようなコンパウンド・シーケンスの設定には先のリニア・シーケンスとはまた違った考慮がはらわれなければならない。

ここでは以下の理由によりつぎのような方法をとる。すなわち、一つのステップで行なういくつかの要素作業の集まり——要素作業グループと名づける——をあらかじめ作成し、これを用いてコンパウンド・シーケンスの組織的な設定法を考察する。その理由は、個々の要素作業を考えてコンパウンド・シーケンスの設定を行なうよりも、一つのまとまった形をとる要素作業グループを用いて設定を行なう方が能率的であり、その要素作業グループに実際上の諸々の条件——たとえば、組合せ上の制約条件、位置上の制約条件など——をおり込ませることによって、より良好なシーケンスの設定が行なえると考えたからである。

このことより、要素作業グループを用いてコンパウンド・シーケンスを設定する場合に必要ないくつかの言葉の定義と記号を導入して問題の具体的な定式化を行なっ

*原稿受付 昭和44年12月27日。昭和44年度精機学会春季大会学術講演会(昭和44年4月4日)にて発表。

**正 会 員 京都大学工学部(京都市左京区吉田本町)

たあと、要素作業グループの組織的な作成法を考察し、ついで、一般的なコンパウンド・シーケンスの設定法を論じる。

2. 定義と記号

コンパウンド・シーケンスを要素作業グループの行ない方に着目して分類すればつぎの2種類に大別できる。

(1) リニア・タイプ・コンパウンド・シーケンス

要素作業グループ中の個々の要素作業が他と重複することなく、順次、処理される場合のシーケンス

(2) オーバラップ・タイプ・コンパウンド・シーケンス

要素作業グループ中の個々の要素作業が他と重複しながら行なわれる場合のシーケンス

この2種類のコンパウンド・シーケンスに対してステーションでの作業進行と時間の関係の一例を示すと図1のようになる。従来のコンパウンド・シーケンスは、主に(1)のタイプのものについてのみ語られており、(2)のタイプのシーケンスについて論じたものはほとんど見あたらない。しかも、不幸なことには、(1)のタイプのシーケンス設定法では、(2)のタイプのシーケンスを満足に作成することができない。本論文では、(2)が(1)の特別な場合であることを示し、(1)、(2)両タイプのシーケンス設定に対して統一した方法を展開する。

さて、第1報で用いた記号以外につぎのような言葉の定義と記号を導入する。

- $x_i \leq x_j$: X を構成するある要素作業 x_i と他の要素作業 x_j との間に先行関係があるということ
- $x_i \not\leq x_j$: x_i と x_j との間に先行関係がないと

(a) Linear type compound sequence
(b) Overlap type compound sequence

図1 作業進行と時間の関係

いうこと

- I型の遷移 : アローを通して遷移する場合、その遷移をI型の遷移と名づける。要素 x_i から要素 x_j にI型の遷移する場合、記号では $c_i(x_i, x_j) = I$ と記す。
- II型の遷移 : アローを通らないで遷移する場合、II型の遷移と名づける。記号では $c_i(x_i, x_j) = II$ と記す。
- $P\{x_j\}$: $= \{x_j \mid x_i \gg x_j, x_i, x_j \in X\}$
 x_i に向かうアローを持つ要素作業の集合
- $F\{x_i\}$: $= \{x_j \mid x_i \ll x_j, x_i, x_j \in X\}$
 x_i から出るアローが到達する要素作業の集合
- $P_A\{x_i\}$: $= \{x_j \mid x_i \geq x_j, x_i, x_j \in X\}$
 x_i に先行するすべての要素作業の集合
- $F_A\{x_i\}$: $= \{x_j \mid x_i \leq x_j, x_i, x_j \in X\}$
 x_i に後続するすべての要素作業の集合
- λ : 要素作業グループ。リニア・タイプの場合には一次元配列、オーバラップ・タイプの場合には多次元配列となる。特に λ が具体的な配列で表示されていることを示す場合には記号 Ar を用いる。多次元配列の場合、横の並びを行配列、縦の並びを列配列と名づける。行配列には記号 α_i 、列配列には β_j を用いる。一般に、各行配列の濃度は異なるけれども、便宜上、多次元配列の最初に位置する列配列を初期列配列、最後に位置する列配列を最終列配列と名づける。なお、多次元配列は一般に行列のような形式に図示されるが、誤解のない場合、印刷の都合上 / で区切って示す。
- $\mu(x_i)$ under Ar : 配列 Ar における最終列配列の一要素である x_i からつぎに遷移する要素の集まり。
- $m\lambda_i$: m は λ_i のグループ濃度、すなわち、 λ_i を構成する要素作業の個数を表わす。 i は個々の要素作業グループ

ブを区別するために添えるものであるが、特に必要のない場合は示さない。時には、コンパウンド・シーケンスにおける割付けの位置を示す。

- ξ : 部分コンパウンド・シーケンス
- Ξ : 完全コンパウンド・シーケンス
- $m\xi_i$: m は ξ_i のシーケンス濃度、すなわち、 ξ_i を構成する要素作業グループの個数を表す。 i は個々のコンパウンド・シーケンスを区別するために添えるものであるが、特に必要のない場合には示さない。
- $m\xi_i(\lambda_1, \lambda_m)$: 要素作業グループ λ_1 から始めて要素作業グループ λ_m に至る m シーケンス濃度の部分コンパウンド・シーケンス
- $S\{\theta\}$: θ が含む要素作業の集合、 θ は要素作業または要素作業グループまたはコンパウンドシーケンス
- $\forall x(x \in S\{\theta\})$: すべての x に対して命題 $x \in S\{\theta\}$ が成立する。
- $t(\theta)$: θ を行なうに必要な時間
- $t(x_1 \cdots x_j)$ in Ar : 配列 Ar における行配列 $(x_1 \cdots x_j)$ を処理するに必要な時間で Ar の関数となる。
- $\tau(\theta)$: トータル・タイム $\tau(\theta) = \sum_{x_i \in \theta} t(x_i)$
- $\oplus$: エレメント和(後述)

3. 要素作業グループの作成

まず最初に、最も基本的なリニア・タイプの要素作業グループの作成について論じ、ついでそれによってえられた知識を基にしてオーバラップ・タイプの要素作業グループの作成へと発展させる。

3.1 リニア・タイプ

3.1.1 要素作業グループとその先行関係

要素作業グループは一般に複数個の要素作業が集まったものであるが、便宜上、単一の要素作業も一つの要素作業によって構成された要素作業グループと考える。

生産システムを構成する要素作業の数が n 個であるとき、理論的に考えられる要素作業グループの総数は $2^n - 1$ 個であるが、先行関係による制約、他の技術的な諸々の制約などによって実行可能な要素作業グループの数は急激に減少する。以下においては、要素作業グループはその

中の構成要素のすべてをおえてからつぎの要素作業グループに移ることができるものと約束する。この約束のもとではつぎのような場合は実行可能とならない。すなわち、ある要素作業グループを構成するいくつかの要素作業の集まりがあって、それらの間で先行関係の定義されているものがあり、しかもそれらがその要素作業グループに含まれていない別の要素作業を介して間接に先行関係が定義されているような場合で、このようなものは実行可能な要素作業グループの対象から除かれる。同様に、他の技術的あるいは経済的な諸々の制約によって課せられる条件を満足しない要素作業グループも実行可能でないとして要素作業グループの対象から除かれる。

以下では要素作業グループが実行可能であると仮定してその先行関係を論じ、そのあと他の技術的な諸々の制約条件の取扱いについて考察する。

さて、先行関係を乱すことなく要素作業グループを実行するためには、その要素作業グループを構成している個々の要素作業の持っている先行要素作業のすべてを終えておく必要がある。いま、要素作業グループ λ につき $P\{x_i\}, F\{x_i\}, P_A\{x_i\}, F_A\{x_i\}$ を用いてつぎのような集合:

$$P\{\lambda\} = \{x_j \mid x_j \geq x_k, x_j, x_k \in \bigcup_{x_i \in \lambda} P\{x_i\}, x_j \in \lambda\} \quad (1)$$

$$F\{\lambda\} = \{x_j \mid x_j \leq x_k, x_j, x_k \in \bigcup_{x_i \in \lambda} F\{x_i\}, x_j \in \lambda\} \quad (2)$$

$$P_A\{\lambda\} = \{x_j \mid x_j \in \bigcup_{x_i \in \lambda} P_A\{x_i\}, x_j \in \lambda\} \quad (3)$$

$$F_A\{\lambda\} = \{x_j \mid x_j \in \bigcup_{x_i \in \lambda} F_A\{x_i\}, x_j \in \lambda\} \quad (4)$$

を導入すれば、これらはそれぞれ、 λ に属する要素作業へ向うアローを持つ要素作業で λ に属さないものの集合、 λ に属する要素作業から出るアローが到達する要素作業で λ に属さないものの集合、 λ に属する要素作業に先行するすべての要素作業で λ に属さないものの集合、 λ に属する要素作業に後続するすべての要素作業で λ に属さないものの集合を表わしている。目的とする要素作業グループ λ の先行関係はこれら $P\{\lambda\}, F\{\lambda\}, P_A\{\lambda\}, F_A\{\lambda\}$ を用いることによって表示される。第1報で導入した直接遷移の条件 $c(i, j)$ に対応するものは $P\{\lambda\}$ であり、一般的には要素作業グループ λ の先行関係を表示するのにこの $P\{\lambda\}$ でもってすれば十分である。

これらの議論においては要素作業グループ λ の実行可能性を仮定している。すなわち、実行可能な要素作業グループの先行関係は、上記の式のようにして、それが持っている個々の要素作業の先行関係から簡単に導ける。したがって、つぎにいかにすれば実行可能な要素作業グループを作成することができるかを明らかにしなければならない。

3.1.2 エレメント和と組合せ行列

要素作業グループを作成する場合の要点は、前述のように、つぎの2点にある。

- (1) 要素作業グループの実行可能であること
- (2) 要素作業グループを組織的に重複なく作成すること。それぞれの目的のためにエメント和と組合せ行列を導入する。

まず最初に、エメント和導入の糸口を作るため、いかなる場合に m 濃度の要素作業グループ $m\lambda_i$ にこれに含まれていない別の要素作業 x_j とをつけ加えて濃度の一つ高い新しい要素作業グループを作成することができるかを考察する。 $m\lambda_i$ を構成している任意の要素作業 x と x_j との関係に着目してつぎの二つの場合に分けて考える。

- (1) x と x_j の間に先行関係がない場合

先行関係を乱すことなく、 $m\lambda_i$ に x_j をつけ加えて新しい要素作業グループを作ることができる。これを $\oplus$ なる記号を用いて

$$m\lambda_i \oplus x_j = m^{+1}\lambda_{i'}$$

と表わすことにする。ここで i' は i との対応関係を示す。

- (2) x_j と x との間に先行関係がある場合

x_j が $P\{m\lambda_i\}$ または $F\{m\lambda_i\}$ のいずれかに属していれば、すなわち、 x_j が $m\lambda_i$ に含まれていない別の要素作業を介して x と間接に先行または後続していなければ、 $m\lambda_i$ に x_j をつけ加えて新しい要素作業グループを作ることができる。

以上のいずれにも該当しなければ $m\lambda_i$ に x_j をつけ加えることはできない。

これらの操作を行なう手順を整理すれば、図2のようなフローチャートを与える。この手順をエメント和と名づける。このエメント和は、 $m\lambda_i$ を構成する要素作業と x_j との間が直接遷移しうる関係にある場合に、 $m\lambda_i$ に x_j がつけ加えられるということを主張している。各要素作業間の直接遷移の条件は第1報で述べたように基本

行列によって表示される。そこでつぎに、要素作業グループを構成する要素作業とそれに含まれない別の要素作業との間の直接遷移の条件を基本行列を用いて表わすことを考える。そのためには、まず最初に、要素作業グループの構成要素である要素作業の配列順序を約束によって一つに統一しておくことが望ましい。このことは、後にのべるように、実行可能な要素作業グループを組織的に重複なく作成するためにも必要なことである。以下において、要素作業グループにおける要素作業の配列順序は、ランクつけた要素作業の番号の順に従うものと約束する。このような約束のもとで、要素作業グループの配列における最後の配列要素から直接遷移しうる要素を基本行列によってとらえる。要素作業グループ $m\lambda_i$ の最後の配列要素を x_i とすれば、基本行列の x_i 行は $F\{x_i\}$ と x_i とまったく先行関係のないものを示したものであるということが出来る。しかしながら、このことから直ちに、 $m\lambda_i$ に基本行列 x_i 行の直接遷移可能な要素作業のすべてをつけ加えてそれぞれ新しい要素作業グループを作成しようと結論することはできない。そこで以下では、これらの直接遷移可能な要素作業のうち、いずれのものが新しい要素作業グループを作る有資格者となりえるかをエメント和を用いて考察することにする。

- (1) まず第1に、エメント和のブロック1から、 x_i 行の直接遷移可能な要素作業で $S\{m\lambda_i\}$ に含まれているものは有資格者となりえない。

- (2) $c(x_i, x_j) = \phi$ なるとき、 x_i から x_j に無条件に遷移可能ということから、 x_j は新しい要素作業グループを作る有資格者である。そして、この場合、新しく作成された要素作業グループ $m^{+1}\lambda_{i'}$ の先行要素作業集合 $P\{m^{+1}\lambda_{i'}\}$ については次式が成り立つ。

$$P\{m^{+1}\lambda_{i'}\} = P\{m\lambda_i\} \quad (5)$$

- (3) いま、

$$c(x_i, x_j) - S\{m\lambda_i\} = c'(x_i, x_j) \quad (6)$$

とおいたとき、 $c'(x_i, x_j) = \phi$ なる関係があれば、直接遷移の条件である先行要素作業が $S\{m\lambda_i\}$ に含まれていることになり、この場合も x_j は新しい要素作業グループを作る有資格者となる。また同様に(5)式が成り立つ。

- (4) $c'(x_i, x_j) \neq \phi$ 中なるとき、つぎの関係：

$$\forall x(x \in c'(x_i, x_j)) \leq S\{m\lambda_i\} \quad (7)$$

が成り立てば、すなわち、 x が $S\{m\lambda_i\}$ のいずれにも先行していれば、 x_j は有資格者である。(7)式の関係は $c'(x_i, x_j)$ に含まれる x が先行関係において $m\lambda_i$ と x_j との中間に位置するようなことがないということ。つまり、 x_j が $F\{m\lambda_i\}$ に含まれているということを主張するものである。この場合(1)式から次式が成り立つ。

図2 エメント和

図3 基本行列によるエレメント和
 $c'(x_i, x_j) = c(x_i, x_j) - S[\lambda_i]$

$$P\{\lambda_i'\} = \{x_h | x_h \geq x_i, x_h, x_i \in P\{\lambda_i\} \cup c'(x_i, x_j)\} \quad (8)$$

(5) (2),(3),(4)の条件のいずれかが成立しない場合には x_j は新しい要素作業グループを作る有資格者となりえない。

以上のことを整理すれば図3のようなフローチャートを与える。注意すべきことは、図3の基本行列を用いたエレメント和では、条件として図2におけるブロック3、ブロックの4の $P\{\lambda_i\}$ 、 $F\{\lambda_i\}$ のうちの $F\{\lambda_i\}$ のみを利用していることである。すなわち、基本行列を用いたエレメント和では、濃度の高い要素作業グループを作成するのに元の要素作業グループに後続する方の要素作業をつけ加えて行なう。このことは、先の図2のエレメント和では x_i が λ_i につけ加えられるかどうかを判定するために $P\{\lambda_i\}$ と $F\{\lambda_i\}$ の両方を用意してやる必要があり、きわめて繁雑であったのに対して、図3のエレメント和では、要素作業グループ内の作業の配列を一つに統一することにより、 λ_i における最後の配列要素 x_i に後続する $F\{x_i\}$ ——すなわち基本行列——を用意してやることだけでよい。このことは、組織的な要素作業グループの作成のためにとってきわめて望ましいことであり、基本行列を導入した目的も実はここにあった。

さて、以上のことで先に示した問題点の(1)を解決したのでつぎに(2)に移る。そのために組合せ行列を導入する。

組合せ行列は、横列に各要素作業をランクづけた番号順に配列し、縦行に λ_i を配列して、 λ_i と x_j とのエレメント和をとるものである。まず基本行列の上三角の部分を利用して濃度2の要素作業グループを作成し、こ

図4 先行関係図

	1	2	3	4	5	6
1	1	2	3	4	5	6
1	1	3	3	2	5	5
2	2	4		3	5	6
2,3	2,3	5				
4,5	4,5	6				
1	1,2		1,2,3	1,2,4		
1	1,3			1,2,3,4	1,2,3,5	
1	2,3			2,3,4	2,3,4,5	
3	2,4					
2	2,5				2,3,4,5	
2	3,4					3,4,5,6
2	3,5					4,3,5,6
2,3	4,5					2,3
5	4,6					4,5,6
3,4	5,6					
1	1,2,3			1,2,3,4	1,2,3,5	
1	1,2,4				1	
1	2,3,4				2,3,4,5	
1	2,3,5					3
3	2,4,5					2,4,5,6
2	3,4,5					2
4	3,5,6					3,4,5,6
2,3	4,5,6					

図5 要素作業グループの作成段階

れらを行列の縦行を延長してつけ加える。それらをふたたび横列の要素作業とが組合わされてできる λ_i を求める。以下、同様にして、順次、高濃度の要素作業グループを作成していく。この際の注意としては、重複した要素作業グループが作成されてくるトラブルを避けることが肝要である。横列に要素作業をランク順に配列し、基本行列の上三角の部分のみを利用するのはこのためであり、また λ_i における配列で最後に位置する要素作業が横列で占める位置より右にある要素作業についてのみ上記の操作を行なう。

以下、例題として図4の先行関係図を用いて説明を行なう。図4の先行関係図に対する組合せ行列の一部、濃度4の要素作業グループの作成段階までを図5に示す。組合せ行列の第1行、ならびに、第2列の第2行から第7行までは、それぞれ各要素作業をランク順に記入したもので、それぞれの行・列のラベルを示したものであり、第1列の第2行から第7行までは先行要素作業を示したものである。先に述べた注意から、第2行から第7行までの上三角の部分に対してのみ基本行列を用意する。濃度2の要素作業グループは、第1報で述べたリニア積と同じ考え方から、基本行列の0でない各要素に、

その行のラベルを並記したものである。たとえば、第3行の2に対しては、3と4と5があるから、これより濃度2の要素作業グループは、2.3, 2.4, 2.5である。また、これらの要素作業グループの先行要素は先の(5)式あるいは(8)式によって1, 1, 3と求められる。このようにして作られた濃度2の要素作業グループは第8行以下に書き加えられて濃度3の要素作業グループの作成に供される。第8行目に目を転じる。8行目に書きこまれた要素作業グループは1.2である。この要素作業グループの最終要素2からは、3行目を見ればわかるように、3, 4, 5に遷移できる。3, 4に対しては無条件に遷移可能で、これより濃度3の要素作業グループ1.2.3, 1.2.4が作成される。5に対しては、3をあらかじめ通過していることを要し(7)の関係式が成り立たないことから新しい要素作業グループを作成することができない。以下、同じようにして、まったく機械的に次第に高濃度の要素作業グループが作成される。

3.1.3 要素作業グループと他の制約条件

実際には、先行関係以外の他の現実的な諸々の制約条件が加わって、要素作業グループがむやみに高濃度になることを制限する。たとえば、組立てラインやトランスファ・マシンでは、一つのステーションで行なういくつかの要素作業の集まり——要素作業グループ——の総作業時間値がある一定の時間——サイクル・タイム、タクト・タイム——を越えてはならないように設定される。また、ワークの形状・大きさなどワークの特性とこれに対する作業方式——ワーク固定かワーク可動か等——の技術的・経済的な理由から、互いに組合わせることのできる要素作業に制限を受けたり、一つのステーションでの作業位置や作業を行なう方向が限定されることがある。このようなものでは、同一ステーションに割りつけられる要素作業は、それらの制約において同一の範ちゅうに入らなければならない。これらの制約条件は、順序の制約に比して取り扱いやすく、組合せ行列においてエレメント和を考えると同時に考慮できる。

さて、以上の過程を経てできた要素作業グループは、このままではシーケンス設定の際に不便であるため、これらを適当な形に整理しておく必要がある。組合せ行列の横列に示した要素作業のランクづけした番号順にしたがって表に整理しておくのが便利である。このようにしてできた要素作業グループ表は、先行関係図の持っている順序の制約やその他の制約条件を要素作業グループによって整理したものであるということが出来る。

3.2 オーバラップ・タイプ

3.2.1 配列問題

オーバラップ・タイプの要素作業グループにおいては要素作業グループ内における各要素作業の配列が問題となる。すなわち、リニア・タイプ（一次元配列）の要素作業グループにおいては、各要素作業間の遷移時間を各要素作業の処理時間に比して無視すれば、前述のように、それらの実行可能な配列は一つに統一することが可能であった。したがって、この場合、配列問題はわれわれの注意を引くものではなかった。これに対して、オーバラップ・タイプ（多次元配列）の要素作業グループにおいては、配列の仕方によりその処理時間は大きく変わってくる。この場合、われわれの関心はいかなる方法で多次元配列を求め、また最小の処理時間で行なえる配列——最小配列と名づける——を見出すかということにある。

オーバラップ・タイプの要素作業グループを作成するには以下にのべるように大別して二つの方法が変えられる。ただし、以下においては、要素作業グループの配列は先行関係を乱さずにそれを構成する全要素作業をもっともコンパクトにつめて作った実行可能な配列のみを問題とする。

① 順序の制約が支配的と考えれば、最初に初期条件として n 次元初期配列要素を与えて、これに要素作業を順次つけ加えて高濃度の配列を求めてゆき、しかるのち、同じ要素作業を含む配列の中で最小配列を決定する。あるいは、この変形として、あらかじめ全体としての要素作業グループの構成要素を決定した後、その中での最小配列を求める。

② オーバラップ・タイプにおいては構成要素の数も大きくなることから、実行可能な要素作業グループの数は膨大なものとなることが容易に想像される。したがって、このことを避けるためには、オーバラップ・タイプの要素作業グループの構成要素としてその行配列をあらかじめ作成し、しかるのちに、組合わせ可能な行配列を求めて多次元配列の要素作業グループを構成する。最小配列は行配列を組合わせる場合に考察される。

①は順序の制約に、②は組合わせの数をおさえることに重点を置いた考え方である。この二つの方法を慎重に考察した結果、①の方法による方が優れていると結論するに至った。④の方法の欠点は、別個に求まった行配列を組合わせる場合の論理操作が順序の制約により複雑を極めたものとなり、それに必要な情報の記憶は組合わせの数をおさえようとする努力を徒労に終らしてしまう。しかも要素作業グループ全体としては同じであっても、行配列の形によって異なった配列がばらばらに求まり、最小配列を求める上においても利点は見出せない。この

ことから、本論文においては①の方法によって多次元配列を求めることにする。なお、ここでは、説明の便宜上①の方法の変形、すなわち、全体としての要素作業グループの構成要素を決めた上でその中の最小配列を見出す。①の方法とその変形との違いは、最小配列を見出す際に後者の方が必要な比較が簡単に行なわれるということであり、多次元配列を求める操作は変わらない。

さて、実行可能な要素作業グループについてその配列を前述の基本行列によるエレメント和を応用することによって求める場合、多次元配列であることから、つぎのような配慮を必要とする。すなわち、リニア・タイプの場合のようにランクづけした番号順に要素作業を配列できないために基本行列の上三角だけでなく全部分を必要とし、さらに間接遷移の条件をも明示しなければならない。たとえば、図6の先行関係図に対して間接遷移の条件までを明示した基本行列は図7のようになる。図においては間接遷移の部分に点線を引いてある。一般に間接遷移の条件までつけ加えれば、最終配列のある要素につけ加えられる要素は、必ず最終配列の他の要素にもつけ加えられる。

さて、間接遷移の条件までを含めた基本行列に対して初期条件、すなわち、 n 次元配列においては n 行の初期配列要素を与えれば、以下は後に示すような形で図3の基本行列によるエレメント和を応用することによって実行可能な配列のすべてを組織的に求めることができる。

まず、 n 行の初期配列要素の第 i 行目より遷移可能な要素は、 n 行の初期配列要素を除いたもののうち、その第 i 行の要素より遷移可能なもので、遷移の条件の先行

図6 先行関係図

	1	2	3	4	5	6
1					3,4) 5	2,4) 6
2					3,4) 5	4) 6
3					4) 5	2,4) 6
4					3) 5	2) 6
5						2) 6
6					3) 5	

図7 間接遷移の条件までを明示した基本行列

要素として n 行の初期配列要素以外のものを求めない要素である。このことから、 n 濃度の初期配列より $(n+1)$ 濃度の配列が作成される。そのそれぞれについて同じようなステップを繰り返し使用することによって実行可能な配列のすべてを求めることができる。しかしながら、このままの方法では同じ配列が重複して作成されてくること起きる。このことを避けることが実行可能な配列のすべてを求める場合のもっとも重要な配慮となる。

<重複を避けるための工夫>

基本行列の縦の行を延長していった m 濃度の配列まで書き加えられているとする。 m 濃度の配列:

$$Ar1 = \{\dots/p\dots x/\dots\}$$

において x に z を接続することが可能であって $p \cdots x \cdot z$ なる配列が $(m-1)$ 濃度以下のところで、あるいは m 濃度のところでは $Ar1$ の配列より上方ですでに作成されている場合には重複がおこる。したがって、このような場合 $Ar1$ に z の接続した $(m+1)$ 濃度の配列:

$$Ar2 = \{\dots/p\dots x \cdot z/\dots\}$$

はこの作成過程ではとらない。

この方法によって同じ配列を重複して作成することが避けられ、実行可能な配列のすべてを組織的に求めることができる。

3.2.2 最小配列

最小配列を求める場合の①の方法とその変形との違いは、前述したように後者の方が簡便に必要な比較が行なえるということにあり、そのため後者は要素作業グループの構成要素をあらかじめ決めておくという、いわば二度手間な操作を必要とする犠牲をはらっている。最小配列を見出すのに別個の定理・方法を用いるものでないから本質的な差はなく、ここでは説明の都合上、例題等で①の方法の変形で説明しているが、①の方法に対してもまったく同様に行なえる。

さてそこで、上記のようにして求めた実行可能な配列の中から最小配列を求める問題に移る。それはどの遷移が他の遷移に比べて、あるいはどの配列が他の配列に比べて優位であるかを考察することになる。まず最初に基本行列について考察する。

<優位定理1> 基本行列において x から y に遷移可能であり、また逆に y から x にも遷移可能である場合、つぎの条件: $P_A(x) \in P_A(y)$ かつ $F_A(x) \ni F_A(y)$ を満足するならば、 $x \cdot y$ を含む配列の処理時間は $y \cdot x$ を含むそれよりも大きくなることはない。したがって遷移 $y \cdot x$ は考慮の対象外における。

(証明) 処理時間に関して、どちらの方が少なくてすむかという可能性の問題である。まず $P_A(x) \in P_A(y)$ の方

図 8 優位定理 1 のための説明図

	1	2	3	4	5	6
1		2	3	4	3,4) 5	2,4) 6
2	X		1) 3	1) 4	3,4) 5	4) 6
3		2		X	4) 5	2,4) 6
4		2	3		3) 5	2) 6
5		X				2) 6
6			X		3) 5	

図 9 図 7 に優位定理 1 を適用した結果

について考察する。図 8 上 $(x \cdot y)$ においては、 $P_A(y) - P_A(x)$ を x を処理しおえるまでに行なえばよく、図 8 下 $(x \cdot y)$ においては y を処理しおえるまでに (図 8 上における x を着手する前までに) $P_A(y)$ を行なっておかねばならない。すなわち、全体として $x \cdot y$ を含む配列の方がよりコンパクトに行なえる可能性をもつ。後者の $F_A(x) \ni F_A(y)$ についても同様の考察が行なえる。これらのことから上述のことがいえる。

この定理によって基本行列を非常に簡単な形に直すことが可能となる。たとえば、図 7 に対して優位定理 1 を適用した結果を図 9 に示す。図で $\times$ 印をつけたものが定理 1 により除去されたものである。これによって間接遷移の条件を付け加えれば最終列配列のある要素につけ加えられる要素は必ず他方にもつけ加えられるということがなくなる。

さて、つぎは配列における優位性を論じる番である。

<優位定理 2> つぎのような配列:

$$\text{Ar } 3 = \{\dots/p \dots x / \dots/g \dots y/\}$$

に対して、 x および y に z を接続することが可能で、しかも $t(p \dots x) \geq t(q \dots y)$ と仮定する。このとき、もし $x \in c(y, z)$ ならばつぎの配列:

$$\text{Ar } 4 = \{\dots/p \dots x \cdot z / \dots/q \dots y / \dots\}$$

が優位である。

(証明) 図 10 (b) においては $t(p \dots x) - t(q \dots y)$ が idle になってしまうが、(a) においてはこの時間も使用しうる可能性がある。配列 (a), (b) が後に接続される要素に影響を与えるのはこの idle 部分のみである。したがって

図 10 優位定理 2 のための説明図

図 11 優位定理 2 系のための説明図

このことから (a) の方が優位であると結論される。

<優位定理 2 系> つぎのような配列:

$$\text{Ar } 5 = \{\dots/p \dots x / \dots/q \dots y / \dots\}$$

において、 x, y におのおの u, v を接続することが可能で、しかも $t(p \dots x) \geq t(q \dots y)$ と仮定する。このとき

$$\bullet \quad c(x, u) = \phi \quad \text{or} \quad c(x, u) \ni y$$

でかつ

$$\bullet \quad c(y, v) \ni x$$

ならばつぎの配列:

$$\text{Ar } 6 = \{\dots/p \dots x \cdot v / \dots/q \dots y \cdot u / \dots\}$$

が優位である。

(証明) 図 11 に対して定理 2 と名じ考察を行なうことによって示される。

<優位定理 3> 実理 2 と同様の仮定のもとで、もし

$$\mu(x) \text{ under Ar } 1 = \mu(y) \text{ under Ar } 2 = z$$

のときはつぎの配列:

$$\text{Ar } 7 = \{\dots/p \dots x / \dots/q \dots y \cdot z / \dots\}$$

が優位である。

(証明) 図 12 (a) において、仮定より y に接続可能なものは z のみであるから、配列元 $p \dots x \cdot z$ の後に y に接続可能なものがあったとしても図の太い点線部分は idle になってしまう。これに対して (b) または (c) の場合には idle になる部分が (a) に比べて大きくなることはない、よっ

図 12 優位定理 3 のための説明図

て上述のことがいえる。

<優位定理 4> 定理 2 と同様の仮定のもとで、もし

$$\mu(y) \text{ under } Ar \ 1=z \text{ であつ } c_i(y, z) = I$$

ならばつぎの配列:

$$Ar \ 8 = \{ \dots / p \dots x / \dots / q \dots y \cdot z / \dots \}$$

が優位である。

(証明) 定理 3 と同じ図 12 を使う。(a) のような配列をとると、 y から z への遷移が I 型であるため図の太い点線部分が idle になってしまう。(b), (c) では idle になる部分が (a) に比べて大きくなることはない。よって (b), (c) の方が優位である。

この定理を有効に用いるためには遷移が何型であるかの情報を基本行列に与えてやる必要がある。図 7 においては I 型の遷移の部分に下線を引いてある。

<優位定理 5> つぎのような配列:

$$Ar \ 9 = \{ \dots / p \dots x \cdot z / \dots / q \dots y / \dots \}$$

に対して z および y に w を接続することが可能で、しかも $t(p \dots x) \geq t(q \dots y)$ と仮定する。いま

$$\mu(y) \text{ under } Ar = \{a, b, \dots, c\}$$

であつ

$$c(y, a) \ni x, c(y, b) \ni x, \dots, c(y, c) \ni x$$

ならばつぎの配列:

$$Ar \ 10 = \{ \dots / p \dots x \cdot z \cdot w / \dots / q \dots y / \dots \}$$

は劣等である。

(証明) 図 13 において (a) の場合図の太い点線部分が idle になる。これに対して $t(p \dots x) \geq t(q \dots y) + t(w)$ となる (b) の場合には明らかに (a) よりも idle は少なくてすむ。(c) は (b) に対して $q \dots y \cdot w$ の後に直ちに $\dots z$ をつけ加えることができるときにさらに idle が少なくてすむことを示している。(d) は $t(p \dots x) < t(q \dots y) + t(w)$ となる場合で (c) と同じように行配列 $q \dots y \cdot w$ の後に $\dots z$

図 13 優位定理 5 のための説明図

をつけ加えることにすれば明らかに (a) より idle は少なくなる。結局、以上のことから配列 (a) に対してつねにそれより優位な配列が存在しえるということがいえる。

<優位定理 6> 二つの n 次元配列 $Ar11, Ar12$ があり、それらの最終列配列をそれぞれ β_1, β_2 としたとき、

$$S\{Ar \ 11\} = S\{Ar \ 12\}$$

$$S\{\beta_1\} = S\{\beta_2\}$$

を満足しているものとする。いま、 $Ar \ 11$ における最終の要素 $x_i (i=1, 2, \dots, n)$ を含む行配列 $(\dots x_i)$ の所要時間 $t(\dots x_i)$ と $Ar \ 12$ における x_i を含む行配列 $(\dots x_i)$ の所要時間 $t(\dots x_i)$ との間に、

$$t(\dots x_i) \text{ in } Ar \ 11 \leq t(\dots x_i) \text{ in } Ar \ 12 \\ (i=1, 2, \dots, n)$$

なる関係があるとき、配列 $Ar \ 11$ が優位である。

(証明) 自明である。

これらの定理によって最小配列の存在する範囲をせまることができる。最終的にはこうして残ったいくつかの配列の中から最小配列を見出すことになる。

最後に初期条件、すなわち初期列配列の与え方について若干の考察を行なう。一次元配列の場合には、各要素作業がそれぞれ初期要素作業となる資格者であって、それぞれにつけ加えうる要素作業を見出して次第に高濃度の要素作業グループを作成した。 n 次元配列の場合、 n 個の並列した要素作業の組を作って初期列配列とするが、要素作業によっては順序の制約によって並列した n 個の要素作業を選び出すことができない場合がある。こ

のときは n 次元配列を作成するのに 足りない分だけのダミイ ϕ をつけ加えて n 次元配列とする. n 個の並列とした要素作業の組みの組織的な作り方はもはや問題にするのに足りないのではちいらない.

あらかじめ要素作業グループが決定されていて, その構成要素の先行関係を整理したときランク 1 の節点の数 m が配列の次元数 n をこえている場合初期条件としては mC_n 個の与え方がある. この場合, つぎの優位定理によってあらかじめこの数をおさえておくことができる.

<優位定理 7> n 次元初期配列 β_3 と β_4 において, $(n-1)$ 個の要素は共通で他の一つがそれぞれ異なっているとす. β_3 のそれを x , β_4 のそれを y としたとき,

$$t(x) \geq t(y) \text{ かつ } F_A(x) \supset F_A(y)$$

ならば β_3 が優位である.

(証明) この定理の証明は, 第 4 報のライン・バランシングのところでもさらに一般的な形に拡張した上で行なうのでここでは省略する.

つぎに一つの具体例を示す. すなわち先の図 7 に対して最小配列を求める. 図 7 の場合の初期条件は配列 $\{1/2\}$ で与えられる. この場合, ランク 1 の数が配列次元数と一致することから新たにダミイをつけ加える必要はない. さて, 図 9 より先のエレメント和を用いて濃度 3 の配列を求めると 図 14 のように 4 個求められる. すなわ

	1	2	3	4	5	6
$\{1/2\}$			$\{1.3/2.3\}$	$\{1.4/2.4\}$		
$\{1.3/2.3\}$				$\{1.3/2.4\}$		
$\{1.4/2.4\}$			$\{1.4.3/2.3\}$	$\{1.4/2.3\}$		$\{1.4.6/2.6\}$
$\{1.2.4\}$			$\{1.3/2.4\}$	$\{1.2.4.3\}$		$\{1.2.4/2.6\}$
$\{1.3/2.4\}$				$\{1.3.5/2.4\}$	$\{1.3.6/2.4\}$	
$\{1.4/2.3\}$				$\{1.3/2.4.5\}$	$\{1.3/2.4.6\}$	
				$\{1.4.5/2.3\}$	$\{1.4.6/2.3\}$	
				$\{1.4/2.3.5\}$	$\{1.4/2.3.6\}$	
$\{1.3/2.4.5\}$					$\{1.3.6/2.4.5\}$	$\{1.3/2.4.5.6\}$
$\{1.4/2.3.5\}$					$\{1.4.6/2.3.5\}$	$\{1.4/2.3.5.6\}$
$\{1.3/2.4.6\}$				$\{1.3.5/2.4.6\}$	$\{1.3/2.4.6.5\}$	
$\{1.4/2.3.6\}$				$\{1.4.5/2.3.6\}$	$\{1.4/2.3.6.5\}$	

図 14 エレメント和による多次元配列とその最小配列

ち, 配列 $\{1/2\}$ の 1 には $\mu(1)$ under $\{1/2\} = \{3, 4\}$ であるから $\{1.3/2\}$, $\{1.4/2\}$ と求められ, また $\mu(2)$ under $\{1/2\} = \{3, 4\}$ であるから $\{1/2.3\}$, $\{1/2.4\}$ とえられる. つぎにこれらを新たに行を作って書き加えてゆく. 配列 $\{1.3/2\}$ のあとには $\mu(3)$ under $\{1.3/2\} = \phi$, $\mu(2)$ under $\{1.3/2\} = \{4\}$ であるから $\{1.3/2.4\}$ が作成される. 同様にして $\{1/2.3\}$ から $\{1.4/2.3\}$ が作成される. つぎに $\{1.4/2\}$ と 3 列目とでは $\{1.4.3/2\}$ と $\{1.4/2.3\}$ が作成される. 前者においては $\mu(2)$ under $\{1.4.3/2\} = \{5, 6\}$ であるが $c(2.5) = \{3, 4\}$, $c(2.6) = \{4\}$ であるために優位定理 5 から $\{1.4/2.3\}$ に比して劣等であるために除去される. また後者は重複を避けるための方法により除去される. つぎに配列 $\{1.4/2\}$ と 6 列目では

$$\left\{ \begin{array}{c} 1. 4. 6 \\ 2 \end{array} \right\} \text{ と } \left\{ \begin{array}{c} 1. 4 \\ 2 \end{array} \right\} \Big|_6$$

が作成される. ここで記号 $|$ は分断記号である. 上記の場合には 4 を終えてから 6 が行なえることを意味している. さて上の二つのうち後者は優位定理 2 により除去される. また前者において $\mu(6)$ under $\{1.4.6/2\} = \phi$, $\mu(2)$ under $\{1.4.6/2\} = \{3\}$ で行配列 2.3 はすでに作成されていることから, 重複を避けるための方法により除去される. つぎに $\{1/2.4\}$ の行に移る. $\{1.3/2.4\}$ は $\{1/2.4.3\}$ に比して優位である. なぜなら $\mu(1)$ under $\{1/2.4.3\} = \{3, 6\}$. $3 \in \mu(1)$ に対しては定理 4 により, また $6 \in \mu(1)$ に対しては $c(1.6) \supset 2, 4$ であるため定理 2 により優位である. また $\{1.3/2.4\}$ は重複を避けるための方法により除去.

$$\left\{ \begin{array}{c} 1 \\ 2 \cdot 4 \end{array} \right\} \Big|_6 \text{ と } \left\{ \begin{array}{c} 1 \\ 2. 4. 6 \end{array} \right\}$$

は後者が定理 2 により優位, また後者においては $\mu(1)$ under $\{1/2 \dots 6\} = 3$, $\mu(6)$ under $\{1/2.4.6\} = \phi$ これはすでに 1.3 なる行配列があることから重複を避けるための工夫により除去. よって結局, $\{1.3/2.4\}$ と $\{1.4/2.3\}$ が濃度 4 の配列として残る. これら濃度 4 の配列からは図のようなステップをへて濃度 5 の配列をえる. このうち $\{1.3/2.4.5.6\}$, $\{1.4/2.3.5.6\}$, $\{1.3/2.4.5.6\}$, $\{1.4/2.3.6.5\}$ はそれぞれ $\{1.3.6/2.4.5\}$, $\{1.4.6/2.3.5\}$, $\{1.3.5/2.4.6\}$, $\{1.4.5/2.3.6\}$ に比して優位定理 3 により劣等であるゆえ消去される. つぎに $\{1.3.6/2.4.5\}$, $\{1.3.5/2.4.6\}$ はそれぞれ $\{1.4.6/2.3.5\}$, $\{1.4.5/2.3.6\}$ に比して優位定理 6 により除去される. 結局, $\{1.4.6/2.3.5\}$, $\{1.4.5/2.3.6\}$ の中に最小配列がある. 簡単な計算の結果これらはともに最小配列になっていることを知る.

4. コンパウンド・シーケンスの設定

図15 コンパウンド積

コンパウンド・シーケンスの設定は

- (1) $\bigcup_i \lambda_i = X$
- (2) $\lambda_i \cap \lambda_j = \phi \quad (i \neq j)$
- (3) $x_i \in S\{\lambda_i\}, x_j \in S\{\lambda_j\}$ で $x_i \leq x_j$ ならば $\lambda_i \leq \lambda_j$

を満足する要素作業グループの集合を見出すことによってえられる。リニア・シーケンスの設定には、要素作業間の直接遷移が先行関係を乱さないための重要な概念であったと同様に、コンパウンド・シーケンスにおいては、要素作業グループ間の直接遷移が問題となる。前者においては $c(i, j)$ が直接遷移の条件を代表していたが、後者においては $P\{\lambda\}$ がそれに対応する。

このことから、1個の部分コンパウンド・シーケンス ${}^{m-1}\xi_i(\lambda_1, \lambda_{m-1})$ と1個の λ_m によって先行関係を乱さずに新たなコンパウンド・シーケンス ${}^m\xi_i(\lambda_1, \lambda_m)$ を作成する場合の手順を整理すれば、結局図15のようなフローチャートをえる。①は λ_m を構成する要素作業が ${}^{m-1}\xi_i(\lambda_1, \lambda_{m-1})$ に含まれないことをチェックするためのものである。②は λ_m を行なう前に終えておかねばならない要素作業が ${}^{m-1}\xi_i(\lambda_1, \lambda_{m-1})$ に含まれているかどうかをチェックするためのものである。結局①と②の条件を満足するときに濃度が一つ高い新しいコンパウンド・シーケンス ${}^m\xi_i(\lambda_1, \lambda_m)$ がえられる。このようにして逐次コンパウンド・シーケンスをえる手順をコンパウンド積と名づける。

コンパウンド・シーケンスの設定は、このコンパウンド積を用いることによってつぎのようになされる。すなわちまず最初に割り付け可能な要素作業グループをすべて列挙して濃度1のコンパウンド・シーケンスを作成する。それらのそれぞれに対して、コンパウンド積を用いて割り付け可能な要素作業グループを列挙し、濃度2のコンパウンド・シーケンスを作成する。以下、順次、このステップを繰り返すことによって、 $|X|=n$ なる場合、多くとも n 回のステップによってすべてのコンパウンド・シーケンスがえられる。

5. 結 論

本論文においてはシーケンス設定問題の基本問題であるコンパウンド・シーケンスの設定法を考察した。すなわち、コンパウンド・シーケンスの設定にあたっては、その構成要素である要素作業グループをあらかじめ作成しておくという方法をとった。これによる利点は、定式化の困難な種々の条件までを考慮することができることと、問題によっては最適解を見出すために必要な試行錯誤の回数を大幅に減らすことができることにある。欠点は現実的な規模の問題では要素作業グループの総数が莫大なものとなることである。しかしながらこの欠点は必ずしもあてはまらない。本論文で展開した方法は、あらかじめ作成した要素作業グループを用いてシーケンスを設定するのであるから、望ましくない要素作業グループを考慮の外において、最初から要素作業グループの数を絞っておくことができる。さらに、要素作業グループの作成自体、きわめて機械的で誤りなく簡単に行なえるものであり、要素作業グループ作成の労と、最適解を見出すための必要な計算の労を比べた場合、前者の方が軽微となる場合が多い。

これらのことから、まず実行可能な要素作業グループ作成のためにエレメント和を定義した。これによって順序の制約を満足する要素作業グループをきわめて機械的に算出することが可能となった。ついで、組合わせ行列を導入することによって重複なく必要な要素作業グループを作成し、これらを最終的に要素作業グループ表という形で整理することによってコンパウンド・シーケンスの設定に便ならしめた。この要素作業グループ表は先行関係図を要素作業グループによって整理したものといえる。さらにオーバーラップ・タイプ・コンパウンド・シーケンスの場合に遭遇する最小配列問題についても考察し、有効ないくつかの優位定理をえた。最後に、要素作業グループ表を用いてコンパウンド・シーケンスの組織的な設定法を考察し、その目的のためにコンパウンド積を導入した。結局、コンパウンド・シーケンスの設定はこのコンパウンド積を用いることによってきわめて機械的かつ組織的に行なえる。

文 献

- 1) 岡村, 山品: リニア・シーケンスの設定, 生産システムの研究 (第1報), 精密機械, **35**, 8 (1969) 509.
- 2) 岡村, 山品: 最適リニア・シーケンスの決定, 生産システムの研究 (第2報), 精密機械, **35**, 9 (1969) 567.