

PERT/MANPOWER問題の最適解法に関する開発研究

A Study on Development of Optimal Scheduling Technique for PERT/MANPOWER Problem

立命館大学 春名 攻*
立命館大学大学院 ○山田 幸一郎**
立命館大学大学院 滑川 達***

BY Mamoru HARUNA, Koichiro YAMADA, Susumu NAMERIKAWA

本論文は、従来から組み合わせの数が膨大となり実用レベルでの適用が不可能とされていた工程計画におけるPERT/MANPOWER問題に対して、ネットワークトポロジー理論を導入することによって、効率的かつ合理的な最適解法を開発することができた。ネットワークトポロジー理論に基づく手法やアルゴリズムは本研究において開発したものであり、工程ネットワークにおける技術的順序関係と管理的順序関係の2つの順序関係を、グラフ理論で扱われているカットセットの概念を利用することにより、トポロジカルな写像関係として整理することができるものである。なお、本論文では、このPERT/MANPOWER問題を、複種類の資源に制約が与えられたときのような、一般的な場合に対して工期を最小にするスケジューリングの問題として定式化している。

【キーワード】 工程計画、PERT/MANPOWER、ネットワークトポロジー理論

1. はじめに

工程計画において、工事用資源の増減を無計画に行うことは非常に困難であるとともに、非常に工事が高価になる場合が多いのが現状である。また、近年における労働力不足、さらには工期が与件として存在するということを考えると、各種の工事用資源の投入量に対する制限を考慮した最適工程計画のためのシステムモデルの開発を行うことが必要であると考えられる。そこで、工事用資源の量的制約のもとでのスケジューリング問題に対して、我々がこれまでに基礎的研究として開発してきた方法、つまり、ネットワークにおける作業間の順序関係をより操作性の高いものとして取り扱っていくという、ネットワークの持つトポロジー的な性質を利用した数理計画モデルを適用することによりこの問題にアプローチすることとした。そして、工程計画業務のより一層の合理化・効率化のために必要な計画・管理技法の

中で、重要なPERT/MANPOWER（資源配分問題）の最適解を求めるということに対しての新たな手法の開発を行なった。

従来、PERT/MANPOWERの問題においては、近似解を求める方法として資源山崩し法等が開発されてきた。また、別の視点から資源の運用問題を取りあげて最適解を求めていくアプローチとしては、ブランチバウンド法等の算量削減の解法を用いる方法が開発されているが、爆発的な算量に対処できないものと認識されている。そこで、本研究では従来から開発してきた資源配分手法を発展させ、PERT/MANPOWERと同様の実用レベルでの条件のもとで、最適解を求める数理計画手法の開発を行なうことが可能となった。

特に本研究では、同時時間帯に施工されている作業は、必ず同時施工が可能な作業で構成されていなければならないということに着目した。この同時に施工可能な作業の構成の中には、資源に制約が課せられている建設プロジェクトの場合、当然工程ネットワーク上の順序関係のみにより抽出した、ある時刻において同時施工が可能な作業全てを、その時刻で施工させることができないものが含まれていることが多い。このためこの種の問題は、必然的に同時に

* 理工学部 環境システム工学科 教授 工学博士
** 理工学研究科 土木工学専攻
*** 理工学研究科 環境社会工学専攻
〒525 滋賀県草津市野路町1916
TEL(0775)66-1111(EXT8770)
FAX(0775)61-2667

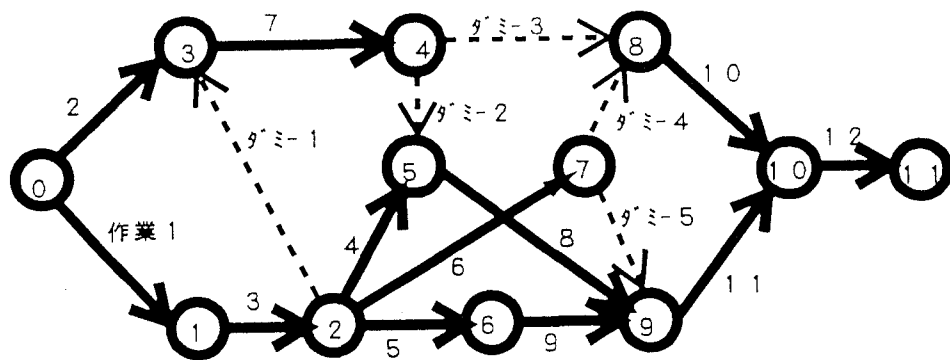


図-1 対象工程ネットワーク

表-1 作業日数および必要資源数

作業	結合点		所要 日数	必要資源数		
	i	j		I	II	III
1	0	1	10	5	2	3
2	0	3	14	4	3	4
3	1	2	3	6	2	4
ダミ-1	2	3	0	0	0	0
4	2	5	5	2	0	2
5	2	6	2	4	2	2
6	2	7	8	3	3	3
7	3	4	8	4	4	3
ダミ-2	4	5	0	0	0	0
ダミ-3	4	8	0	0	0	0
8	5	9	4	7	3	5
9	6	9	6	4	2	0
ダミ-4	7	8	0	0	0	0
ダミ-5	7	9	0	0	0	0
10	8	10	2	4	4	2
11	9	10	3	4	2	2
12	10	11	2	3	4	5

表-2 資源制約数

職 種	I	II	III
制限量	10	6	8

施工する作業の構成を求めていく組み合わせ問題となる。よって最小の計算量で最適解を求めるためには、まず同時に行うことのできる作業を効率的に抽出し、同時刻に施工する作業の組み合わせ数を極力抑えなければならない。本論文では、本研究で開発したネットワークポロジ理論に基づく方法を用いることにより、この組み合わせを導出される解の最適性を保証する最低限の数に抑えることができたと考えられる。

また、従来の本研究における資源配分手法は、「資源に作業を配分する」と考えることとして問題を解決していた。しかし、複種類の資源制約条件下での問題解決に関しては、組み合わせ数も膨大となり、また極端に複雑な問題となる場合が多くあった。

そこで、資源配分手法を、「作業に資源を配分する」と考えることによって、複種類の資源を同時に配分することができ、問題の難易度も1種類資源制約の場合と変わりのない手法を開発することができた。

なおここで、PERT/MANPOWERの問題を、以下のよう

に定義しておくこととする。すなわち、「工程上の全ての作業を終了するまでに要する日数が最小となる工程を求めることを目的とする。但し、1つの作業は複種類、複数の資源を必要とし、各種種類の資源には同時に使用できる数には制限がある。また、任意の作業が開始されるとその作業を途中で停止させることはできない。」

以後本論文では、図-1で示す例題モデル（表-1：作業日数と必要資源数、表-2：制約資源数）を用いて説明を行うこととする。また、図-1の例題モデルに対してPERT計算を行いその資源の山積み図を作成したものを参考のために図-2（いずれの種類の資源も資源制約数を超過している）に示しておく。

2. 資源配分問題に関する考察

建設工事においては、技術上の問題から工事施工に関わる基本的な順序関係として一意的に与えられる技術的順序と、工期の短縮や資源の効率的運用を考えて計画者が任意に設定を行う管理的順序の2種類の順序関係が存在している。例題のモデルで示すと、作業1と作業3は技術的順序関係であり、作業3は作業1が終了した後に行わなければならない。当然、作業1と作業3は同時に施工することはできない。また、作業1と作業2は管理的順序関係であり、作業1と作業2は同時に行うこともでき、作業1→作業2、作業2→作業1という施工順序を設定してもかまわない。この管理的順序関係は、新たな順序設定をしない限り多くの同時施工を許可することになり、工期短縮を実行することができたとしても使用資源が集中するといったリスクを負うことになる。そして、資源配分問題の最適化を考えていくうえでは、作業の順序関係を如何に合理的・効率的

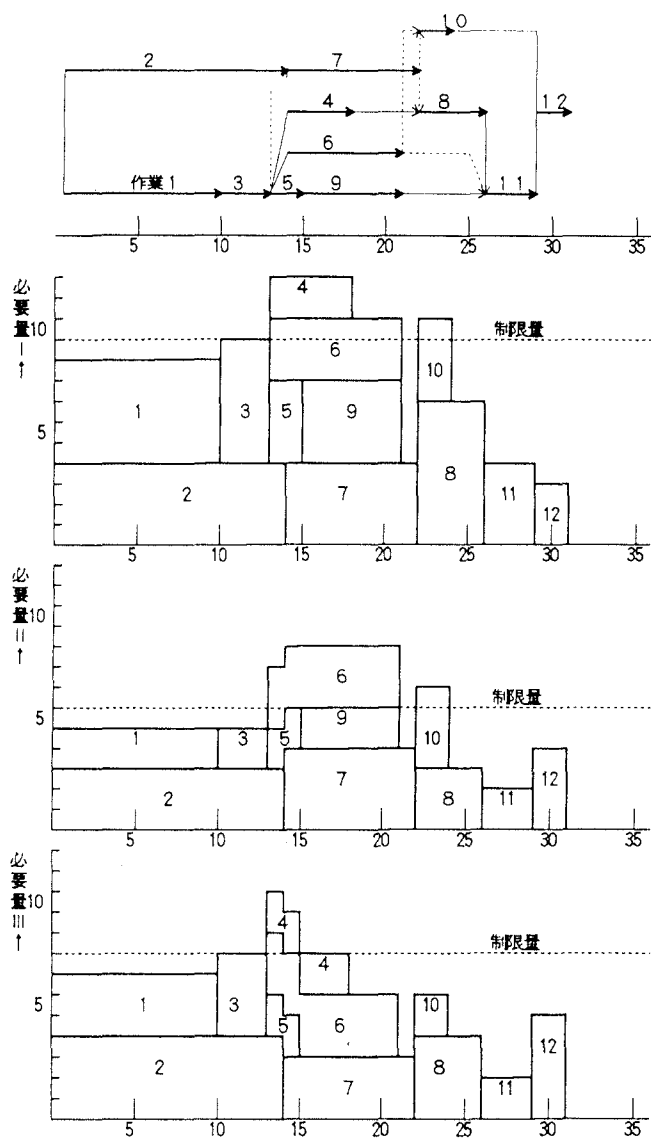


図-2 PERT計算により算出した
工程計画の資源山積み図

に決定していくかが重要であると考えた。理論的には、作業の順序関係の決定問題は、順序組み合わせの探索問題であり、対象となる工程ネットワークが大きくなるにしたがい、新たに付加される順序組み合わせも膨大な数になるとともに、複雑な組み合わせ問題となり、コンピュータ利用において、最適解が算出されたとしても、なかなか効率的な方法であるとは言いがたい問題である。つまり、より効率的に最適解を決定するためには、組み合わせを極力押さえる必要がある。

そして資源配分問題を考える場合に、資源に制約がある場合、ない場合を問わず、工程の任意の時間に行われている作業は常に同時施工が可能な作業でなければならない。同時に工程の技術的順序関係を踏まえていなければならない。この管理的順序関係

と技術的順序関係を整理しない限り、膨大な作業の組み合わせを生じさせることとなる。本研究では、管理的順序関係と技術的順序関係を合理的に区別するための理論的な展開を実現させている。

また、工程に資源の制約がない場合には、同時間に行うことのできる全ての作業を施工させることができるが、しかし反対の場合には、同時間帯に行う作業の数を制限しなければ資源の制約条件を満たすことができないために、同時間帯に行う作業の組み合わせを設定しなければならないのである。本研究では、この同時間帯に行う作業の組み合わせを極力削減させるためにネットワークのトポロジカルな性質を利用することで解決することができた。これが本論文で言うネットワークトポロジー理論である。

3. ネットワークトポロジー理論

本論文でのネットワークトポロジー理論とは、対象となる工程ネットワークには管理的順序関係と技術的順序関係の2種類の順序関係が複雑に絡み合っている存在しており、これらを合理的に判別する方法である。まず、管理的順序関係である作業をグルーピングするために、グラフ理論で扱われているカットセット（以後カットという）を、「カットに含まれる作業は全て管理的順序関係である」という条件を満たすように本研究で開発したカット（図-3）を用いることによりこのグルーピングに対応している。そして、工程ネットワークにおける技術的順序関係にある作業とカットによりグルーピングされた管理的順序関係にある作業を整理するために、カット間の順序関係を構造化させることにより対応できるのである（図-4、図-5参照）。本論文では、このカット間の順序関係の構造を示すものをカットネットワーク（図-5）と名して扱うこととする。なお、内面的な工程ネットワークの要素を保持したまま、表面的に異なったネットワークに変換することからもネットワークトポロジー理論と命名している。

(1) カットセット（カット）

本研究では、従来のPERT/CPMで定義されているカット概念に着目することとした。このカット概念は「ネットワークにおける全ての経路（パス）を同時に短縮する事が可能な作業の集合」と捉えられてき

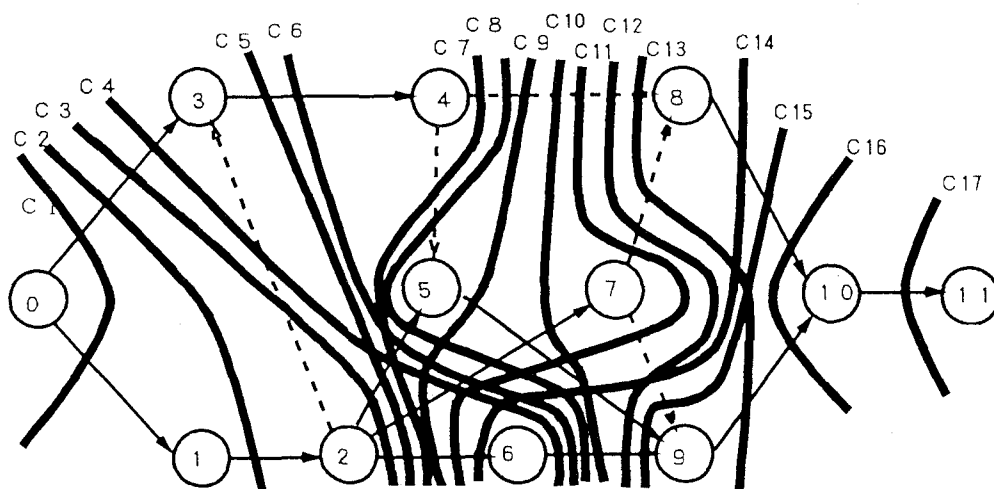


図-3 工程ネットワークにおけるカット

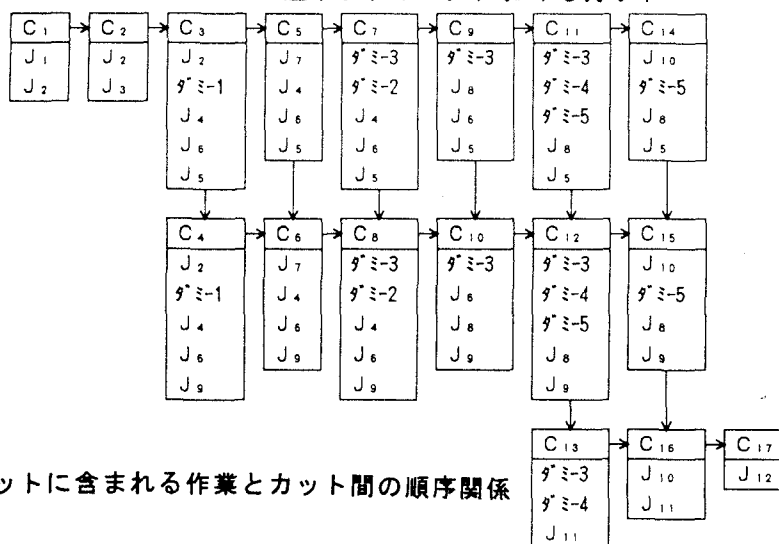


図-4 カットに含まれる作業とカット間の順序関係

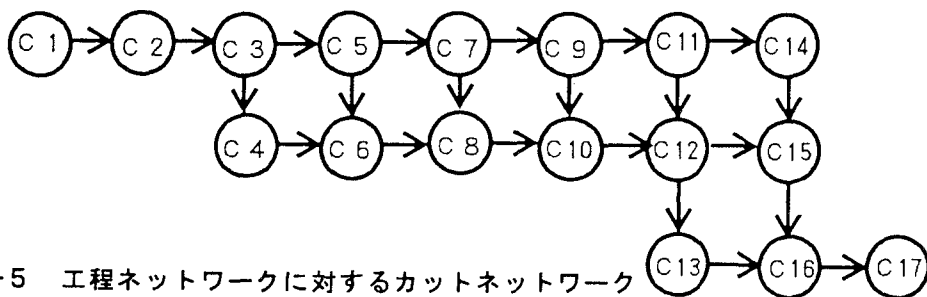


図-5 工程ネットワークに対するカットネットワーク

たが、「同時施工が可能な作業の集合」として捉えることとした。この概念より、カットの持つべき必要条件を

[1] 任意のカットは常に工程を2分する。

[2] 任意のカットに含まれる作業間には技術的順序関係を存在させない。

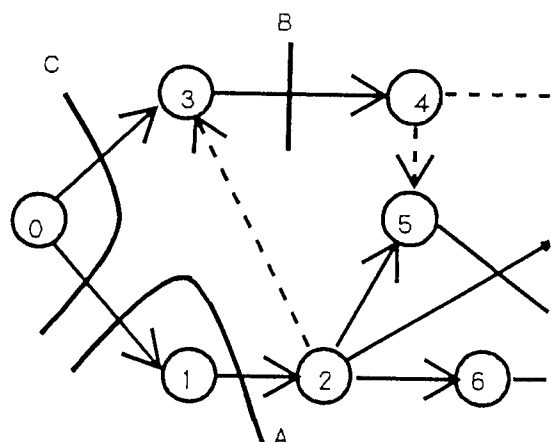
とする。図-6を用いて示すと、カットAは技術的順序関係である作業1と作業3を含んでいるために、つまり[2]の必要条件に反しているために、本研究で扱うカットではない。また同様にカットB

は工程を2分していないカットであり、[1]の必要条件を満たしていないために、本研究のカットとはしない。本研究で扱うカットは、カットAのような工程を2分しており、かつ、カットに含まれる作業は全て管理的順序関係であるものとする。よって、[1]、[2]の必要条件によりどの作業とどの作業が同時刻に作業を行うことが可能なのかを把握することができるのである。言い換えれば、任意の作業がどの作業と管理的順序関係にあるのかを容易に把握することができるのである。

(2) カットネットワーク

当然工程ネットワークは、時間の経過と共に進んでいくものであり、時間と共に施工される作業も変化してくるものである。そこで本研究では、カットを単独で扱わずにカットの相互関係を考慮していくこととした。カ

ットがあくまでも作業の集合に対応して求められることに着目すれば、「作業の持つ順序関係を写像したカット間の順序関係」も存在するはずである。つまり、カット上の作業の順序関係を集約すれば、カットも工程ネットワークの作業と同様に順序関係を持つこととなり、もとの工程ネットワークの順序関係が保持されることとなる。すなわち、「カット間には、作業および工程ネットワークの構造関係がトポロジカルに写像されている」こととなる。つまり各々のカットは、「同時施工が可能な作業の集合」



カットA：技術的順序関係の作業を含む。
 カットB：ST間において工程を2分していない。
 カットC：ST間を2分し、
 技術的順序関係の作業を含まない。

図-6 カットの種類と本研究におけるカット

を示しているために、カットを時間系列につなげあわせることにより、工程ネットワークの全作業を技術的順序関係と管理的順序関係に整理することができるのである。つまり、カットより作業の管理的順序関係を、カットを時間系列に並べることより作業の技術的順序関係を考慮することができるのである。

上述の関係をい用いれば、図-5に示すように、カット同士の順序関係により1つのネットワーク（カットネットワーク）として表現することができる。

また、定義されたカットが工程上で交錯する場合は、それらカット間の順序関係は、存在しない状態であることがわかる。つまり、カットネットワークのルートはカット同士が交錯しないように、工程ネットワークの始点から順次取り出した一連のカットからなる経路を複数の経路を重ね合わせたものから成り立っているとも言える。このことから、カットネットワークの特性をまとめると、次のようである。

[1] 始点と終点を結ぶ任意の1経路は、もとの全作業を含んでいる。

[2] 経路の順序は、工程ネットワークの順序関係を保持している。

以上のようなカットの特性を考慮したカットネットワークを利用して、本研究では最適資源配分モデルに対して検討を行うこととした。

(3) ネットワークトポロジー理論の検討

a) カットセットの抽出方法

本研究では、従来のグラフ理論で扱われているカ

ットの抽出方法にかわり、より合理的に抽出する必要があると考えた。そのためには以下のような2つの注意事項を満たす必要がある。

(1) 工程ネットワークを対象にカットを抽出することから、工程ネットワークの初期条件である作業リストのデータのみから直接カットを求める必要がある。

(2) カットの抽出において、カットの持つべき必要条件を満たすカットをそのつど検索することなく、直接抽出する必要がある。

また、[1]、[2]の2つのカットの持つべき必要条件を数学的に表し直すと、

[1]' 任意のカットに含まれる全作業の先行集合と可達集合の和集合に工程ネットワークを構成する全作業が含まれていなければならない。

[2]' 任意のカットに含まれる作業の先行集合、可達集合の作業がそのカットに含まれていてはならない。

ここで、先行作業をある作業以前に発生する作業、可達（後続）作業をある作業以後に発生する作業とする。それに、先行集合を先行作業の集合、可達集合を可達作業の集合とする。

以下にカット抽出のための(1)、(2)の注意事項を踏まえ、[1]'、[2]'の必要条件を満たす本研究で開発したカットの抽出方法を述べることにする。

カットの抽出方法は、まず、ネットワークプランニングの際、初期条件として与えられる「作業リスト」（表-3）のみを入力情報とする。「作業リスト」は、任意の作業の次に起こる作業、もしくはその作業のひとつ前に起きた作業との関係を表したものであり、マトリックス（関係行列）に表示すると（表-4）の様に表すことができる。そして、構造化理論問題で用いられているISM法(Interpretive Structural Modeling)を用いて、この関係行列を構造化させ、ネットワークの構造を表す行列である可達行列（表-5）を算出する。この可達行列は、ある作業の先行集合、もしくは可達集合を求める場合に、可達行列のその作業の要素の列ベクトル、もしくは行ベクトルを調べることで求められる。しかし、ここではカットを抽出することを目的としているために、列ベクトル、または行ベクトルを調べるのみで、あ

表-3 作業リスト

作業	後続作業
1	3
2	7
3	4、5、6、ダミ-1
ダミ-1	7
4	8
5	9
6	ダミ-4、ダミ-5
7	ダミ-2、ダミ-3
ダミ-2	8
ダミ-3	10
8	11
9	11
ダミ-4	10
ダミ-5	11
10	12
11	12
12	-

表-4 作業の関係行列（' はダミ-作業を示す）

	1	2	3	1'	4	5	6	7	2'	3'	8	9	4'	5'	10	11	12
1			1														
2								1									
3				1	1	1	1										
1'								1									
4											1						
5												1					
6													1	1			
7									1	1							
2'											1						
3'															1		
8																1	
9																	1
4'															1		
5'																1	
10																	1
11																	1
12																	

表-5 可達行列

	1	2	3	1'	4	5	6	7	2'	3'	8	9	4'	5'	10	11	12
1	1		1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2		1						1	1	1	1				1	1	1
3			1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1'				1				1	1	1	1				1	1	1
4					1						1					1	1
5						1						1				1	1
6							1						1	1	1	1	1
7								1	1	1	1				1	1	1
2'									1		1					1	1
3'										1					1		1
8											1					1	1
9												1				1	1
4'													1			1	1
5'														1		1	1
10															1		1
11																1	1
12																	1

表-6 先行・可達行列

	1	2	3	1'	4	5	6	7	2'	3'	8	9	4'	5'	10	11	12
1			1	1	1	1	1										
2								1	1	1	1				1	1	1
3	1			1	1	1	1	1	1	1	1	1	1	1	1	1	1
1'	1			1				1	1	1	1				1	1	1
4	1										1					1	1
5	1											1				1	1
6	1												1	1	1	1	1
7	1	1	1	1	1				1	1	1				1	1	1
2'	1	1	1	1	1						1					1	1
3'	1	1	1	1	1										1		1
8	1	1	1	1	1	1					1					1	1
9	1					1										1	1
4'	1														1		1
5'	1															1	1
10	1	1	1	1	1	1	1				1						1
11	1	1	1	1	1	1	1	1	1	1	1	1	1	1			1
12	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	

る作業と技術・管理的順序関係にある作業を判断できることが望ましい。つまり、抽出された可達行列をもとに以下の計算をすることで解決することができる。

先行・可達行列 = 可達行列 + 可達行列' - 単位行列
（' は行列の転位変換）

そして、可達行列から作成した先行・可達行列（表-6）において、ある作業 j_i の行ベクトル $M_L(j_i)$ （ j_i ）に注目し、そのベクトル要素 $a_i = 0$ に対応する作業を全て抽出する。 $a_i = 0$ となっている作業 i と管理的順序関係にある作業であり、作業 i を含んでいるカットに含まれる作業である。そして、ここで抽出した作業を j_k とすると、 j_k の行ベクトル $M_L(j_k)$ と、先の $M_L(j_i)$ を、ブール則（ $0+0=0, 1+0=1, 1+1=1$ ）を適用して足し合わせて新たにできた行ベクトル $\{M_L(j_i) + M_L(j_k)\}$ において

$$\{M_L(j_i) + M_L(j_k)\} = [a_{ij} + a_{kj}]$$

$$= \begin{cases} 1 : \text{それ以外の作業の要素} \\ 0 : j_i, j_k \text{の要素} \end{cases}$$

を満足するものをカットとみなすこととする。ここでは、作業間の技術的順序関係が「ある」、「ない」のみを要求するために 0, 1 の表現とした。

つまり、 $\{M_L(j_i) + M_L(j_k)\}$ のベクトル要素において、 j_i と j_k のみが 0 であり、それ以外が全て 1 であれば、 j_i と j_k は順序関係を持たず、また j_i と j_k の可達集合、先行集合の和はネットワークを構成する全ての作業を含んでいる。すなわち、本研究におけるカットの必要条件をを満たしており、カットとみなすことができる。

以下に例題モデルに対して、カット c_3 抽出までの一例を示す。先行・可達行列の j_2 の行ベクトルに着目すると、作業 j_2 は、作業 $j_1, j_3, j_{1'}$ （ $j_{1'}$ はダミ-作業 1 を示す）、 $j_4, j_5, j_8, j_9, j_{4'}, j_{5'}$ とは技術的順序関係を有していない。そこで、作業 j_2 に作業 $j_{1'}$ の行ベクトルの要素を加えると、

$$M_L(j_2) = [0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 1, 1, 1]$$

$$M_L(j_{1'}) = [1, 0, 1, 0, 0, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 1, 1, 1]$$

$$M_L(j_2) + M_L(j_{1'}) = M_L(j_{2_{1'}})$$

$$= [1, 0, 1, 0, 0, 0, 0, 1, 1, 1, 1, 1, 0, 0, 0, 1, 1, 1]$$

となる。この段階では、作業 $j_2, j_{1'}$ 以外の作業 j

4, j₅, j₆, j₉, j₄, j₅ に対して技術的順序関係を持っていないこととなる。言い替えれば、作業 j₄, j₅, j₆, j₉, j₄, j₅ と作業 j₂, j₁ とは、同一のカットに含まれる可能性があることとなる。それに、作業 j₄, j₅, j₆, j₉, j₄, j₅ の行ベクトルについては検討を加えていないので、この段階では、まだカットは工程を 2 分していないこととなる。つづいて、先行・可達行列の作業 j₄ の行ベクトルの要素を $M_L(j_{21})$ に加えると、

$$M_L(j_{21}) + M_L(j_4) = M_L(j_{214}) \\ = [1, 0, 1, 0, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 1, 1, 1]$$

となり、作業 j₂, j₁, j₄ と管理的順序関係にある作業は、作業 j₅, j₆, j₉, j₄, j₅ となり、続けて作業 j₅ の行ベクトルの要素を $M_L(j_{214})$ に加える。
 $M_L(j_{214}) + M_L(j_5) = M_L(j_{2145})$

$$= [1, 0, 1, 0, 0, 0, 0, 1, 1, 1, 1, 1, 0, 0, 1, 1, 1]$$

この段階で作業 j₂, j₁, j₄, j₅ と管理的順序関係にある作業は、作業 j₆, j₄, j₅ であり、続けて作業 j₆ の行ベクトルの要素を $M_L(j_{2145})$ に加えると、
 $M_L(j_{2145}) + M_L(j_6) = M_L(j_{21456})$

$$= [1, 0, 1, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]$$

となる。この段階で作業 j₂, j₁, j₄, j₅, j₆ と管理的順序関係である作業はなくなり、それら以外の他の作業とは技術的順序関係が存在しており、カットの必要条件である [1], [2] を満たしたこととなり、カットが探索されたこととなる。つづけて同様の処理を行った結果、図-3 のように 17 個のカットが得られた。

b) カットネットワークの抽出方法

カットネットワークは、カット同士の一対比較、すなわち、2 つのカット間の順序関係を調べれば容易に求められる。つまり、2 つのカットベクトルを構成する作業間に同じ先行後続関係が認められる場合のみ、これらのカット間に順序関係が存在すると判断すればよい。

この関係にもとづいて、全てのカット間の順序関係を機械的に算出することも可能である。それには、カット C_i に含まれる全ての作業が、カット C_j の所有する作業の可達行列に全て含まれる場合には、カット C_i はカット C_j の後続関係にあるという条件を用いればよい。

このようにして、カット同志の比較が可能になれば、グラフの構造化手法を用いて全てのカット間の構造を求めることができる。

(4) ネットワークトポロジーのアルゴリズム

ここで、作業リストからカットネットワーク抽出までの一連のアルゴリズムの流れに関して順を追って述べることとする。

I. 与えられた作業リストから、初期関係行列、2 値関係行列、可達行列、先行・可達行列の作成。

1) 初期関係行列 <O> の作成

作業リストから「作業 j_i は作業 j_j の後続作業か」という 1 対比較において、そうであるならば 1, そうでなければ 0 を記入し、初期関係行列 <O> を作成する。

2) 2 値関係行列 の作成

単位行列 <I> を用いて、 $B = O + I$ として 2 値関係行列 を求める。

3) 可達行列 <R> の作成

 のべき乗をブール演算により、作業間の順序関係が全て構造化されるまで、つまり $B^{k+1} = B^k (= R)$ まで計算し可達行列 <R> を求める。ここで、 $R = [a_{ij}]$ において

$$a_{ij} = \begin{cases} 1 : \text{作業 } j_i \text{ が } j_j \text{ の後続作業} \\ 0 : \text{それ以外の時} \end{cases}$$

4) 先行・可達行列 <M> の作成

先に求めた可達行列 <R> の作業 j_i の行ベクトル $R_L(j_i)$, 列ベクトル $R_C(j_i)$ に対して $M_L(j_i) = R_L(j_i) + R_C(j_i)^t$ としてブール則をもちいて <M'> を求める。つまり $M = M' - I$ により先行・可達行列 <M> を求める。ここで、 $M = [a_{ij}]$ において

$$a_{ij} = \begin{cases} 1 : \text{作業 } j_i \text{ が } j_j \text{ の先行} \\ \quad \text{または後続作業の時} \\ 0 : \text{それ以外の時} \end{cases}$$

II. 先行・可達行列を用いた、カット抽出法

- 1) I. で作成した先行・可達行列 <M> において、ある作業 j_i の行ベクトル $M_L(j_i)$ に注目し、 $a_{ij} = 0$ の作業を全て抽出する。
- 2) いま抽出した作業群の行ベクトル $M_L(j)$ を、

いま注目している作業 j_i の行ベクトル $M_L(j_i)$ にブール則 ($0+0=0, 1+0=1, 1+1=1$) を用いて加え、その行ベクトルの和、 $M_L(j_i) + M_L(j)$ においてその行ベクトルの和を構成している作業の要素のみが0となれば、カットであると決定する。

なぜなら、その足し合わせた行ベクトルで、'0'は、管理的順序関係を、'1'は、技術的順序関係を意味しており、よって、足し合わせた作業同士は、管理的順序関係になり、カットの条件[2]を満たしたことになる。また、足し合わせた作業以外の作業の要素は'1'であり、それらの作業以外に管理的順序関係が存在しないためにカットの条件[1]を満たしたことになる。

$$M_L(j_i) + M_L(j) = \begin{cases} 1 : \text{それ以外の作業の要素} \\ 0 : j_i, j \text{ の要素} \end{cases}$$

このとき、 j_i を中心として全ての組合せについて考える。

3) 全作業の行ベクトルについて、同様に行なう。

Ⅲ. カット間関係の構造化によるカットネットワークの作成

1) カット間の関係行列 $\langle D \rangle$ に対して単位行列 $\langle I \rangle$ を加えて2値関係行列 $\langle N \rangle$ を求める。つまり、次式のような N を求める。

$$N = D + I$$

2) 行列 N の $K-1$ 乗が K 乗と等しくなるまでべき乗を繰り返し、可達行列 $\langle N' \rangle$ を求める。

$$N' = N^K = N^{K-1}$$

3) カット間の構造関係を可達行列 $\langle N' \rangle$ より求める。可達行列 $\langle N' \rangle$ においてカットが可達関係でも先行関係でもない要因の集合を逐次求め、その集合ごとの要因と隣接する集合間の要因の関係を求めることにより、要因間の順序関係を求める。

このようにカット間の関係行列を上述のアルゴリズムに適用することにより構造グラフが得られると同時にカットネットワークの作成が可能になる。

4. 最適資源配分問題に関する検討

カットは管理的順序関係である作業の集合であり同時時間帯に行うことのできる作業の集合である。一方、カットネットワークの任意のルートは、同時時間帯に行うことのできる作業を工程ネットワークの技術的順序関係を踏まえた形で時間軸上につなぎ合わせたものである。つまり、資源に制約が課せられた場合に同時時間帯に行われる作業群は、このカットネットワークのルートに沿った作業群と等しくなる。

本研究では、上述のネットワークポロジ理論を用いることにより作業の技術、管理的順序関係が時間的に配列されたカットネットワークに沿って資源制約条件下で工程を進めていくことで効率的に資源を作業に配分させていくことができた。たとえ資源制約条件により作業の進捗が複雑になろうともカットネットワークのルートに沿いながら作業を進めさせるために、与えられた工程ネットワークの順序関係を踏まえた工程計画が作成されるのである。

(1) 作業配分順序の設定

カットネットワークの各々のカットは、同時に施工することができる作業の集合ではあるが、資源制約を考慮している作業の集合体ではないために、どのような作業をその時間帯に行うべきであるのかを設定する必要が生じてくる。本研究では最適解抽出を目的としているために、その設定はカットに含まれている作業の全ての組み合わせを求めなければならない。しかし、この組み合わせはあくまでもそのカット内での工程を表すものであり、ネットワーク全体を考慮しているものではない。そこで、本研究では、作業の行われかたの組み合わせを考えるのではなく、作業の開始される配分順序として考えることにより問題を解決することができた。しかし、複数のカット中に同じ作業が重複して含まれているために、この状態で工程ネットワークの作業全体の配分順序を設定することは不可能である。

そこで、カットの移動に着目することとした。カットが移動する際には、作業の種類を前のカットに含まれる作業と後のカットに含まれる作業とが共通である作業、前のカットのみで含まれる作業、後のカットのみで含まれる作業というように分けることができる。つまり、前のカットのみで含まれる作業

表-7 カットネットワークの1つの経路に含まれる作業

カット (ルート順)	c 1	c 2	c 3	c 4	c 6	c 8	c 10	c 12	c 13	c 16	c 17
カットに含まれる作業	j 1 j 2	j 2 j 3	j 2 j 4 j 5 j 6	j 2 j 4 j 6 j 9	j 7 j 4 j 6 j 9	j 4 j 6 j 9	j 8 j 6 j 9	j 8 j 9	j 11	j 10 j 11	j 12

カットの移動	c 0 → c 1	c 1 → c 2	c 2 → c 3	.	.	c 16 → c 17
新しい作業	j 1, j 2	j 3	j 4, j 5, j 6	.	.	j 12
作業配分順序 (番号は作業 を示す。)	1 → 2	1 → 2 → 3	1 → 2 → 3 → 4 → 5 → 6	.	.	.
			1 → 2 → 3 → 4 → 6 → 5	.	.	
			1 → 2 → 3 → 5 → 4 → 6	.	.	
			1 → 2 → 3 → 5 → 6 → 4	.	.	
			1 → 2 → 3 → 6 → 4 → 5	.	.	
			1 → 2 → 3 → 6 → 5 → 4	.	.	
	1 → 3 → 2	1 → 3 → 2	1 → 3 → 2 → 4 → 5 → 6	.	.	
			1 → 3 → 2 → 4 → 6 → 5	.	.	
			1 → 3 → 2 → 5 → 4 → 6	.	.	
			1 → 3 → 2 → 5 → 6 → 4	.	.	
			1 → 3 → 2 → 6 → 4 → 5	.	.	
			1 → 3 → 2 → 6 → 5 → 4	.	.	
			1 → 3 → 4 → 2 → 5 → 6	.	.	
			1 → 3 → 4 → 2 → 6 → 5	.	.	
			1 → 3 → 4 → 5 → 2 → 6	.	.	
			.	.	.	
			.	.	.	
			1 → 3 → 6 → 5 → 4 → 2	.	.	
	2 → 1	2 → 1 → 3	.	.	.	
			.	.	.	

：カットが移動し際に新しく付加した作業の配分順序

注：最終的な作業の配分順序は、カットが c 17 に移動したときに作成されるものである。

図-7 作業の配分順序 (一部掲載)

に関しては、次のカット以降には含まれないために、前のカットの段階で作業の配分順序を設定しなければならない。つまり、そのカットで配分順序を設定しなければならない作業をカットネットワークの始点終点間で整理することで全ての作業に対して配分させる地点を設定することができるのである。

その作業配分順序の設定方法を、図-5のカットネットワークのルートが $C_1 \rightarrow C_2 \rightarrow C_3 \rightarrow C_4 \rightarrow C_6 \rightarrow C_8 \rightarrow C_{10} \rightarrow C_{12} \rightarrow C_{13} \rightarrow C_{16} \rightarrow C_{17}$ と移動する時の例を交えながら以下の順序に従って説明することとする。

① カット c_s (c_s : カットネットワークの始点) に含まれている作業の全ての順序組み合わせを設定する。

例: c_1 には j_1 、 j_2 の作業が含まれており、その組み合わせは $j_1 \rightarrow j_2$ 、 $j_2 \rightarrow j_1$ である。

② 続いてカット c_s から $c_{s'}$ に移動したときに、どの作業 j_x が終了したためにそのカット $c_{s'}$ に移ったのかを調べる。さらに $c_{s'}$ に新しく含まれる作業 j_y を求める (j_x 、 j_y は一つの作業とは限らない)。

例: c_1 から c_2 に移動したときに j_1 が終了したために c_2 に移ることができ、新しく j_3 が含まれてくる。

③ ①で作成した順序組み合わせにおける、②で求めた作業 j_x より後の作業の組み合わせを削除する。

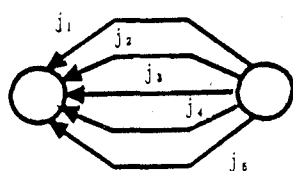
例: $j_1 \rightarrow j_2$ の場合を例にとると、 j_1 より後の j_2 を削除する。

④ ③において組み合わせを削除した作業と②で求めた作業 j_y とにより、全順序組み合わせを作成する。

例: j_1 の後に続く順序は、 j_2 と j_3 により $j_2 \rightarrow j_3$ 、 $j_3 \rightarrow j_2$ の組み合わせが作成され、この段階で $j_1 \rightarrow j_2 \rightarrow j_3$ 、 $j_1 \rightarrow j_3 \rightarrow j_2$ の作業配分順序が作成された。

⑤ 以後、カットネットワークの終点にいたるまでカットの移動毎に②、③、④を繰り返し続ける。

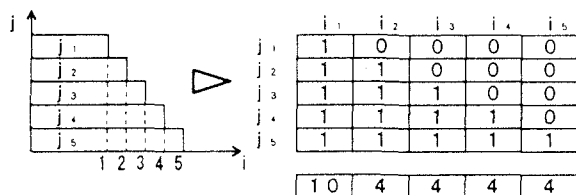
この作業の配分順序方法に従って設定したところ、図-7のように設定することができた。また、本研究では最適解の抽出を目的としているために、カット内でグルーピングされた作業に対しては、それらの作業の配分順序を意図的に与えることはできない



例題ネットワーク

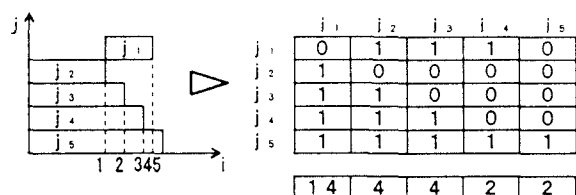
	作業日数	必要 資源数
j ₁	10	1
j ₂	14	1
j ₃	18	1
j ₄	22	1
j ₅	26	1

実施状況マトリックス



例題ネットワークの行列表現

配分時間ベクトル



資源制約条件下での行列表現の一例

(資源制約数: 4)

図-8 実施状況マトリックスに関する説明図

めに、全ての考慮できる作業の配分順序の組み合わせを設定しなければならない。

(2) 作業の実施状況の検討

作業が何時の時間帯で実施しているのかを明確にさせるために、本研究では、作業が開始される時刻と終了される時刻を有していることに着目した。つまり、任意の作業は開始時刻と終了時刻を2つの時間を有しており、当然、工程にN個の作業がある場合に、工程全体ではN個の作業の開始時刻、もしくは終了時刻が存在すればよいこととなる。従って、作業の実施状態を $N \times N$ の2次元行列 G （以下、実施状況マトリックスと呼ぶ）に設定することが可能となる（図-8にて実施状況のマトリックス化に関して図示）。ここで、実施状況マトリックスの*i*方向は時間区間を、*j*方向は作業番号を示している。そして、実施状況マトリックス $G \{a_{ij} \mid i=1, 2, \dots, N, j=1, 2, \dots, N\}$ の要素 a_{ij} を、「区間*i*において作業*j*を 0：行なっていない、1：行なっている」と表わすことができる。そして、 G の要素 a_{ij} の*i*区間の時間長を表わすベクトルを配分時間ベクトル

$X \{x_i \mid i=1, 2, \dots, N\}$ と設定する。但し、その x_i の時間的区間の長さは、作業の配分順序により異なってくるものであり、事前に設定することはできない。ここに、作業の実施状況を以下のように定義することができる。

$$\sum_{i=1}^N a_{ij} \cdot x_i = d_j$$

$$(i=1, 2, \dots, N, j=1, 2, \dots, N)$$

$$a_{ij} \in \{0, 1\} \quad x_i \geq 0$$

d_j ：作業日数

そして、本研究の目的である工期最小の目的関数を以下のように定義することができる。

$$\lambda = \min \sum_{i=1}^N x_i$$

λ ：最小工期

(3) 作業への資源配分

以上で作業への資源配分のための準備を整えることができた。ここでは、上述した実施状況マトリックスに、資源を前説した作業の配分順序に従って作業を配分させていくことを述べることにする。

資源を配分させていく際に考慮する点は、2点ある。1つは以下の式で示すように、常に工程が資源制約数以内で行なわれていることである。

$$\sum_{j=1}^N a_{ij} \cdot r^k_j \leq R^k$$

r^k_j ：作業*j*が必要としている種類*k*の資源数

R^k ：種類*k*の資源制約数

2点目は、カットネットワークから設定された作業の配分順序は、どの作業が行なわれるとカットが移動するのかわを示していないために、別の言い方をすれば、どの作業とは同時に施工を行なうことができないのかわを示していないために、資源を配分させていく過程で、再び同時間帯に行なわれる作業と新しく配分する作業が管理的順序関係または技術的順序関係なのかを調べなければならないことである。そこで、前説した作業の先行・可達行列 $\langle M \rangle$ を用いて新しく配分する作業*j*の実施される時間帯（ G における区間*i*）において行われている作業群*J*とが技術的順序関係であるかどうか調べなければならない。つまり、以下の条件を満たす必要がある。

作業配分順序

j₁→j₂→j₃→j₄→j₅→j₇→j₉→j₆→j₈→j₁₀→j₁₁→j₁₂

①

	1	2	3	4	5	6	7	8	9	10	11	12
1	1											
2												
3												
4												
5												
6												
7												
8												
9												
10												
11												
12												
	10											

②

	1	2	3	4	5	6	7	8	9	10	11	12
1	1											
2	1	1										
3												
4												
5												
6												
7												
8												
9												
10												
11												
12												
	10	4										

③

	1	2	3	4	5	6	7	8	9	10	11	12
1	1											
2	1	1	1									
3			1									
4												
5												
6												
7												
8												
9												
10												
11												
12												
	10	3	1									

④

	1	2	3	4	5	6	7	8	9	10	11	12
1	1											
2	1	1	1									
3			1									
4				1	1							
5												
6												
7												
8												
9												
10												
11												
12												
	10	3	1	4								

⑤

	1	2	3	4	5	6	7	8	9	10	11	12
1	1											
2	1	1	1									
3			1									
4				1	1	1						
5				1	1							
6												
7												
8												
9												
10												
11												
12												
	10	3	1	1	3							

⑥

	1	2	3	4	5	6	7	8	9	10	11	12
1	1											
2	1	1	1									
3			1									
4				1	1	1						
5				1	1							
6												
7						1	1	1				
8												
9												
10												
11												
12												
	10	3	1	1	3	4						

⑦

	1	2	3	4	5	6	7	8	9	10	11	12
1	1											
2	1	1	1									
3			1									
4				1	1	1						
5				1	1							
6												
7						1	1	1	1			
8												
9						1	1					
10												
11												
12												
	10	3	1	1	3	3	1					

⑧

	1	2	3	4	5	6	7	8	9	10	11	12
1	1											
2	1	1	1									
3			1									
4				1	1	1						
5				1	1							
6												
7						1	1	1	1			
8												
9						1	1					
10												
11												
12												
	10	3	1	1	3	3	1	8				

⑨

	1	2	3	4	5	6	7	8	9	10	11	12
1	1											
2	1	1	1									
3			1									
4				1	1	1						
5				1	1							
6												
7						1	1	1	1			
8										1		
9						1	1					
10												
11												
12												
	10	3	1	1	3	3	1	4	4			

⑩

	1	2	3	4	5	6	7	8	9	10	11	12
1	1											
2	1	1	1									
3			1									
4				1	1	1						
5				1	1							
6												
7						1	1	1	1			
8										1		
9						1	1					
10											1	
11												
12												
	10	3	1	1	3	3	1	4	4	2		

⑪

	1	2	3	4	5	6	7	8	9	10	11	12
1	1											
2	1	1	1									
3			1									
4				1	1	1						
5				1	1							
6												
7						1	1	1	1			
8										1		
9						1	1					
10											1	
11											1	1
12												
	10	3	1	1	3	3	1	4	4	2	1	

⑫

	1	2	3	4	5	6	7	8	9	10	11	12
1	1											
2	1	1	1									
3			1									
4				1	1	1						
5				1	1							
6									1	1		
7					1	1	1	1				
8								1				
9					1	1						
10										1		
11										1	1	
12												1
	10	3	1	1	3	3	1	4	4	2	1	2

最適工程計画案

図-9 計算過程図（作業配分順序に従い、作業を配分状況マトリックス上に配分していく。）

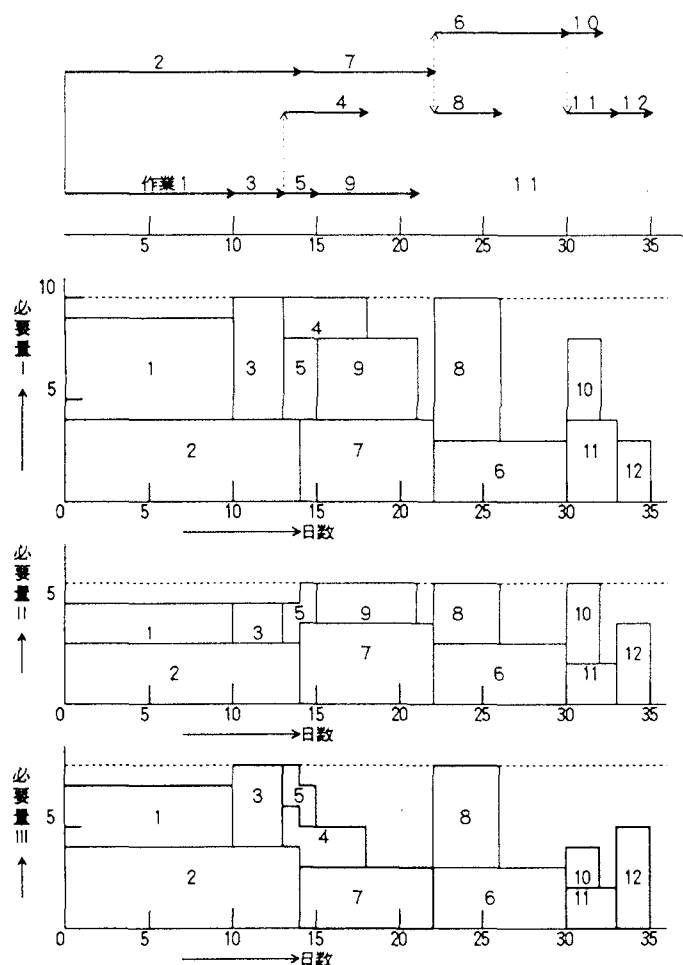


図-10 図-9の計画案のアローダイアグラム
および資源山積み図

$$m_{ji} = 0$$

m_{ji} : 先行・可達行列の要素であり、作業 i と作業 j が管理的順序関係であれば“0”、技術的順序関係であれば“1”である。

J^i : G における時間帯 i で行われている作業群

j : 配分しようとしている作業

以上より資源を資源に配分させていく過程で考慮する点は、常に資源制約数内で工程を行なうこと、同時時間帯に技術的順序関係にある作業が存在してはならないことである。

そして、作業の実施状況マトリックスへの割り付け箇所は、工期最小を目的としているために、その作業が最も早く開始することができる場所とする。しかし、その作業に資源を配分することで資源制約数を超えてしまう場合、およびその作業と同時間帯に行う作業が技術的順序関係である場合には、その作業の実施させる時間帯をずらすことで回避することとする。実施状況マトリックス上では、その作業の実施させる時間区間 (i 方向) を資源制約数以内

で工程が行なえるまでずらすこととする。

そして、以上の資源配分に関する検討を踏まえ例題モデルに対して最適解を作成する作業配分順序 ($j_1 \rightarrow j_2 \rightarrow j_3 \rightarrow j_4 \rightarrow j_5 \rightarrow j_7 \rightarrow j_9 \rightarrow j_{10} \rightarrow j_{11} \rightarrow j_{12}$ (最適解の作業配分順序の一例)) により計算を行った計算過程を図-9で示す。また、その計算結果をもとにアローダイアグラムおよび資源山積み図を作成したものを図-10で示す。

5. おわりに

一概にどのくらい効率的な手法であるのかを定量的に説明することは、たとえ作業数と同じであったとしてもネットワークの形が異なっていれば計算量も異なってくるため難しい。ただ、この手法の場合、管理的順序関係が多く含まれる工程に従い計算量も増えるものとなる。また、資源配分問題に対してブランチバウンド法を用いた手法では、あくまでも探索型の手法であるために、余分な計算をしなければならなかったが、本研究の手法では、ネットワークトポロジーを用いることで、この余分な計算を極力削減させることができた。そして、PERT/MANPOWERにたいする本研究の過去の研究の問題点であった複種類資源の配分を解決することができた。最後に、ここに開発した資源配分手法は、実用レベルでの使用可能性があるものへ展開することが可能となったと考える。今後はPERT/CPMを考慮した手法や、資源制約が非一定の場合の条件下での手法の改善へと挑んでいくことにする。

【参考文献】

- 1) 春名 攻: 建設工事における施工管理に関するシステム論的研究, 京都大学学位論文, 1971, 9.
- 2) 春名 攻, 荒川 和久, 山田 幸一郎: ネットワークトポロジー理論を用いた最適工程計画の理論開発に関する研究, 土木学会建設マネジメント研究論文集, 1993, 12.
- 3) 春名 攻, 原田 満, 荒川 和久: ネットワークトポロジー理論を用いた工程計画のための日程短縮モデルに関する研究, 土木学会土木計画学・研究論文集, 1992, 11.
- 4) 荒川 和久: ネットワークトポロジー理論を用いた最適工程計画のスケジューリングモデルに関する研究, 立命館大学修士論文, 1993, 3.
- 5) E. L. Lawler and D. E. Wood : Branch and Bound Method: A Survey, JORSA Vol. 15, 1967.
- 6) James J. O'Brien : CPM in Construction management, McGraw-Hill, 1965.
- 7) Claude Berge: Theory of Graph and its application, Methuen & Co Ltd, 1962.
- 8) 関根 智明: ORライブラリー-11 PERT・CPM入門, 日科技連, 1989, 8.