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

～工事用資源の割付問題を中心として～

Study on Development of Optimization Model for Construction Scheduling Based on Theory of Network Topology

立命館大学　　　　　春名　攻 *
兵庫県　　　　　　　荒川　和久**
立命館大学大学院　○山田幸一郎***

By Mamoru HARUNA, Kazuhisa ARAKAWA, Koichiro YAMADA

厳しさを増す施工環境や施工条件のもとで、「人、機械、材料といった資源の調達・運用を最大限に努力して、より早く、より安く、かつ安全に施工を完了すること」は、建設業にとって重要な問題である。そして、建設工事マネジメントのシステム化をはかるとともに、建設工事施工に関わる業務の合理化・効率化と生産性の向上を実現していくことが必要である。そこで、本研究では、現場マネジメント業務の中でも中核的業務として位置づけられる工程計画業務に着目し、これらの業務をより一層合理化・効率化するために必要なシステマティックなマネジメント業務を支える計画・管理技法の新たな開発について検討を行った。特に、本研究においては、工程ネットワークの持つ性質を変えることなく、より操作性の高いものとして取り扱っていくといったネットワークトポロジー理論を用いた新しいタイプの資源割り付けモデルについての開発研究を行った。

【キーワード】工程計画、ネットワークトポロジー、カット理論

1．はじめに

本研究では、現場マネジメント業務の中でも、中核的業務として位置づけられる工程計画業務に着目し、数理計画モデルを導入した工程計画のシステム化のための理論検討を行うこととした。

特に、本論文では、まず、計画者が過去の経験や資源の運用・転用といった問題を考慮して試行錯誤的に決定していた管理的な施工順序の決定問題に着目するとともに、工事資源が管理的順序関係の決定に大きな影響を与えているものと考えれば、効率的な資源運用を考慮した、最適化という問題に対して検討を加えていく必要があると考えた。

そこで、本研究では、我々がこれまでに基礎的研究として開発してきた、ネットワークトポロジーの理論をベースとする、新しいタイプの資源割り付けモデルに関する検討を行った。

2．効率的な施工順序決定方法に関する検討

ここでは、我々がこれまでに基礎的研究として開発してきた、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、ネットワークトポロジー的な数理計画モデルを施工の順序関係決定モデルにも適用することによって、より効率的な施工順序決定問題の解法の方法についい

*　理工学部土木工学科
TEL(075)465-7856 FAX(075)465-8201
**　姫路土木事務所
TEL(0792)81-3001(EX325)
***　理工学研究科土木工学専攻　TEL(075)465-7856

て述べるとともに、工事への投入可能な資源量の制約のもとで最小工期を与える工程計画の作成方法について述べることとする。

（１）技術的順序と管理的順序に関する考察

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。この管理的順序関係は、新たな順序関係を設定しない限り、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。工程計画・管理では、契約工期と資源の量的制約が課された状態で、主要資源の投入効率を最大にすることが要求されるが、順序関係の問題はこの管理的順序関係の問題に他ならず、その設定及び決定の方法は今日の課題として残されている。

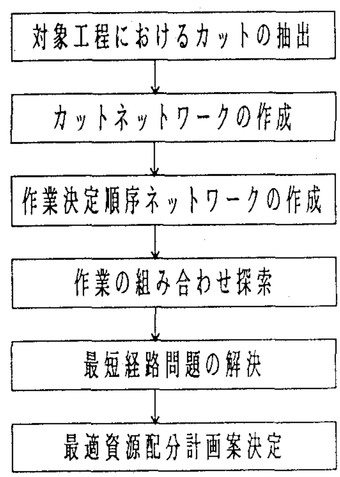
（２）最適工程計画の作成に関する検討

ａ）本モデルの解法方法に関する検討

工程計画の最適化を考えていくうえでは、やはり管理的な順序関係に着目し、この順序関係を如何に合理的・効率的に決定していくかが重要であると考える。

理論的には、管理的順序関係の決定問題は、順序組み合わせの探索問題である。この組み合わせ問題の有効な手段として、ブランチ・バウンド法がある。しかし、対象となる工程ネットワークが大きくなるにしたがって、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いいがたい。つまり、より合理的、効率的に最適解を決定するためには、組み合わせを極力減らし、コンピュータの特性を有効に活用できる数値演算形態にしていく必要があると考えた。そこで、本研究では、これらの考えを念頭にいれ、これまでにも述べてきたネットワークトポロジーのカット理論に注目することとした。

そこで、本研究の資源問題解法方法を図－１に示すような６段階の処理プロセスに沿って求めていくこととし、本モデルの内容をこの処理プロセスに沿って説明していくこととする。まず、①対象工程ネットワークをもとにカットを作成し、②この抽出されたカット間の関係をトポロジカルに等値変換したカットネットワークを作成し、③このカットネットワークにカットが移動した時に新しく加わってくる作業の決定順序を重ねた作業決定順序ネットワークを作成し、④この作業決定順序ネットワークに沿いながら、対象工程の技術的順序関係および、資源の制約を満たしながら資源に作業を配分させていくものである。そして、本研究の資源配分問題の最適解となる組み合わせを、工期を最小となる組み合わせとし、研究をおこなった。また、ここでは、問題の内容や理論的検討をわかりやすく説明するために、図－２に示すような工程の要件が与えられ、資源制約数が２という場合をとりあげて理論的検討の内容とその有効性についての検討を行うこととする。



図－１　最小工期探索モデルの検討フロー

３．カット探索方法

本研究では、従来のＰＥＲＴ／ＣＰＭにおいて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）を同時に短縮することが可能な作業の集合」として捉えられてきたが、本研究では、このカット概念を理論上、「同時施工が可能な作業の集合」として再認識することとした。このことによって、容易に設定すべき順序関係の組合せを作り出すことが可能であり、また、これらのカットに含まれる作業

間に順序関係を設定することが可能であることが理解できると考えた。

そこで、従来までのグラフ理論を用いてのカット探索にかわり、より合理的、効率的にカットを探索するためには、以下のような、２つの注意事項を考える必要があった。

1）工程ネットワークを対象に、カットを探索することから、この初期条件である作業リストのデータだけから直接的にカットを求めることが必要である。

2）カットの探索において、下で述べるような必要条件[1],[2]の条件を満足する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。

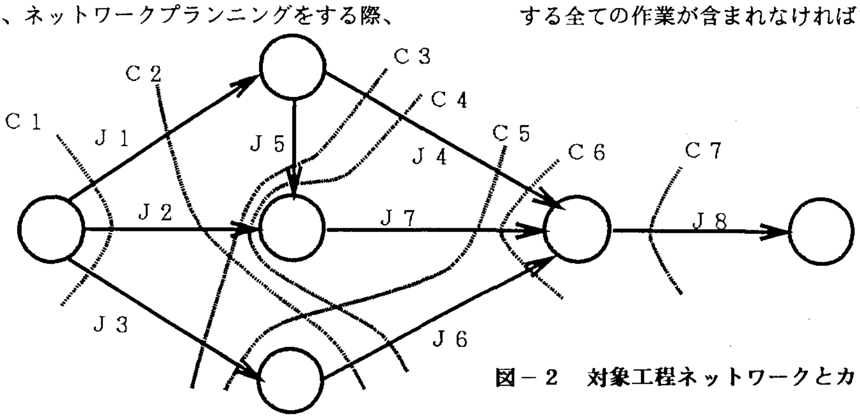
そこで、ここでは、1）の注意事項を満足するために、ネットワークプランニングをする際、初期条件として与えられる＜作業リスト＞（表－1）だけを用いて初期関係行列から＜先行・可達行列＞（表－3）を作成し、この＜先行・可達行列＞を利用し、カットを探索する手法について述べる。また、2）の注意事項については、カットを探索していく過程で説明していくこととする。ここで、本研究におけるカットの持つべき必要条件は、下記に示す[1],[2]の２つの条件である。

[1] 任意のカットは、つねに工程を２分する。

[2] カットに含まれる作業間には、順序関係を存在させない。

上の[1],[2]条件は、すなわち次のようになる。

[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ての作業が含まれなければならない。



図－2　対象工程ネットワークとカット

＜作業日数リスト＞

作業	所要日数（日）
J 1	5
J 2	10
J 3	4
J 4	9
J 5	4
J 6	8
J 7	7
J 8	8

表－1　作業リスト

作業	先行作業	後続作業
J 1	－	J 4, J 5
J 2	－	J 7
J 3	－	J 6
J 4	J 1	J 8
J 5	J 1	J 7
J 6	J 3	J 8
J 7	J 2, J 5	J 8
J 8	J 4, J 6, J 7	－

表－2　可達行列

	J1	J2	J3	J4	J5	J6	J7	J8
J1	1	0	0	1	1	0	1	1
J2	0	1	0	0	0	0	1	1
J3	0	0	1	0	0	1	0	1
J4	0	0	0	1	0	0	0	1
J5	0	0	0	0	1	0	1	1
J6	0	0	0	0	0	1	0	1
J7	0	0	0	0	0	0	1	1
J8	0	0	0	0	0	0	0	1

表－3　先行・可達行列

	J1	J2	J3	J4	J5	J6	J7	J8
J1	0	0	0	1	1	0	1	1
J2	0	0	0	0	0	0	1	1
J3	0	0	0	0	0	1	0	1
J4	1	0	0	0	0	0	0	1
J5	1	0	0	0	0	0	1	1
J6	0	0	1	0	0	0	0	1
J7	1	1	0	0	1	0	0	1
J8	1	1	1	1	1	1	1	0

[2]' 任意のカットに含まれる作業の先行集合、可達集合には、その作業以外カットに含まれる作業は存在してはならない。

ここで、先行作業を、ある作業以前に発生する作業、後続（可達）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。

つまり[1]'[2]'の2つの条件を考え、このような必要条件を満足するものを本研究におけるカットとする。そして、従来のものとちがい[1]', [2]'の条件について以下のように考えた。

今回、あらたに、可達行列（表－2）から作成した先行・可達行列において、ある作業J_iの行ベクトル$M_L(J_i)$に注目し、そのベクトル要素$a_{ij}=0$に対応する作業を全て抽出する。そして、いま抽出した作業をJ_kとすると、J_kの行ベクトル$M_L(J_K)$と、さきの$M_L(J_i)$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、順序関係が「ある」、「ない」だけを要求するために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だけが0で、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である[1],[2]を満たしており、カットであるとみなすことができる。

① 先行・可達行列の性質

可達行列では、ある作業の先行集合、可達集合を求める場合、それぞれ列ベクトル、行ベクトルをみなければならない。そこで、行ベクトルまたは列ベクトルのいずれかをみるだけで、先行集合と可達集合が同時に求められるような行列を、先行・可達行列<M>とよぶこととする。

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（$0+0=0, 1+0=1, 1+1=1$）を適用して、<M'>を求め、$M=M'-I$（I：単位行列）より先行可達行列<M>を求める。

ここで、$M=[a_{ij}]$において

$$a_{ij}=\begin{cases} 1 : \text{作業}J_j\text{が}J_i\text{の先行又は後続作業の時} \\ 0 : \text{それ以外の時} \end{cases}$$

② カット探索の手順

今回のカット抽出までの流れについて順を追って述べる。

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_jは作業J_i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<B>の作成

単位行列<I>を用いて、$B=O+I$として2値関係行列<B>を求める。

3）可達行列<R>の作成

<B>のベキ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$B^{K+1}=B^K(=R)$まで計算し可達行列<R>を求める。ここで、$R=[a_{ij}]$において

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

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、$M=M'-I$より先行・可達行列<M>を求める。ここで、$M=[a_{ij}]$において

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

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だけが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。

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

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）　全ての作業の行ベクトルについて、同様に行なう。

③カット抽出の一例

以下にカット抽出の対象として一例を示す。先行・可達行列のJ_1の行ベクトルに着目すると、作業J_1は、作業J_2，J_3，J_6とは技術的順序関係を有していない。そこで、作業J_1に作業J_2の行ベクトルの要素を加えると、

$M_L(J_1)=\{0,0,0,1,1,0,1,1\}$

$M_L(J_2)=\{0,0,0,0,0,0,1,1\}$

$M_L(J_1)+M_L(J_2)=M_L(J_{12})$

$=\{0,0,0,1,1,0,1,1\}$

となる。この段階では、作業J_1，J_2以外の作業J_3と作業J_6に対しても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J_3，J_6と作業J_1，J_2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J_3，J_6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。言い替えれば、作業J_3，J_6と作業J_1，J_2とは、同一のカットに含まれる可能性があることとなる。つづいて、先行・可達行列の作業J_3の行ベクトルの要素を$M_L(J_{1+2})$に加えると、

$M_L(J_{1+2})+M_L(J_3)$

$=\{0,0,0,1,1,1,1,1\}$

となり、作業J_1，J_2，J_3以外の作業については技術的順序関係が存在しており、また、検討した作業のみ管理的順序関係となり、カットの必要条件である[1]，[2]を満たしたこととなり、これをカットとする。

つづいて、$M_L(J_{1+2})$に作業J_6の要素を加えると、

$M_L(J_{1+2})+M_L(J_6)$

$=\{0,0,1,1,1,0,1,1\}$

となり、同様にこれもカットとみなすことができる。

このような方法により、例題についてカットを求めたところ、図－2で示すように7つのカットが探索された。

4．カットネットワークへの等価変換

本研究では、カットをそれぞれ単独で扱わずに、カットの相互関係を考慮して行くこととした。

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、「作業のも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存されることになる。

すなわち、「カットには、作業および工程ネットワークの関係構造がトポロジカルに写像されている」こととなる。つまり、工程ネットワークにおける作業の順序関係は、そのままカット間の関係において

も成立しているのである。

さて、上述の関係を用いれば、図－３に示すように、カットも、工程ネットワークと同様に、１つのネットワーク構造（以後、カットネットワークと呼ぶ）として描くことが可能である。

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。また、言い方を変えれば、カットネットワーク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、今回、本研究では、資源割り付けモデルについて検討することとした。

なお、このような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、すべ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。これには、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各カッドのレベルを求めることとする。つまり、まず、上述した方法で一対比較して求められるカット間の関係行列 D に対して単位行列Ｉを加えて行列N_cを求める。
つまり、次式のようなN_cを求める。

$$N_c = D + I$$

つぎに、行列N_cの（Ｋ－１）乗がＫ乗と等しくなるまでベキ乗を繰り返し、可達行列N_c'を求める。

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

次にこの行列を用いて、各要素S_iに対して

$$\text{可達集合}: R(S_i) = \{S_j \mid c'_{ij} = 1\}$$

$$\text{先行集合}: A(S_i) = \{S_j \mid c'_{ji} = 1\}$$

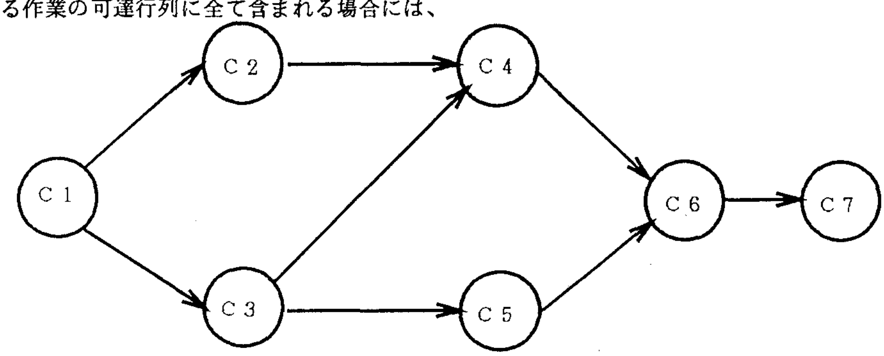
を求め、要素のレベル決定は、この可達集合と先行集合を用いて、

$$R(S_i) \cap A(S_i) = R(S_i)$$

となる可達関係でも先行関係でもない要因の集合を逐次求め、レベルごとの要因と隣接するレベル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。このようにカットの関係行列を上述のようなアルゴリズムに適用することによって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。

５．作業組み合わせ

従来の本研究における資源配分問題の組み合わせ方法は、まず、あらかじめステージ毎に資源制約を考慮したうえで、組み合わせを作成し、その組み合わせをカットネットワークの経路に沿って、組み合



図－３　対象工程ネットワークにおけるカットネットワーク

わせていくという順序にしたがい処理をおこなう経路探索問題であった。

今回、本研究では、あらかじめステージ毎に組み合わせを作成する事をせずに、カットネットワークに沿いながら制約条件を同時に満足させながら、資源を配分していく新たな手法を開発した。そのことにより、組み合わせをより合理的かつ効率的におこなうことができ、そして、組み合わせ作成・経路探索を通して計算量を減少させることができ、迅速に最適解を求めることができた。

5．1従来の組み合わせ方法

そこで、従来の手法との違いを比較するために、従来の方法から説明することとする。図－3に示すようなカットネットワークにおけるカットC_iに含まれる作業について、ブランチバウンド法を適用することで、各々のステージの組み合わせを決定した。つまり、各カットに含まれる作業を資源制約をブランチバウンド法の条件として組み合わせを作成することとした。

ここで、各ステージにおけるカットに含まれる作業のブランチの方法であるが、分かりやすく説明するために、ステージ1からステージ2に関しての一例をとって説明していくこととする。ここでの、ブランチをおこなう際の条件として、資源制約は2とする。

ステージが一つ進んだときのカットの移動の一例として、C_1からC_3の移動について検討することとする。ここでは、カットがC_1からC_3に移動するときにカットに含まれる作業の変動は、$C_1\{J_1, J_2, J_3\}\rightarrow C_3\{J_2, J_3, J_4, J_5\}$となり、新たに加わる作業は、$J_4$, J_5となる。つまり、この新しく加わってくる作業を前ステージのカットC_1までで決定された順序関係の後に加えることとする。ブランチの対象となる作業は、C_3に含まれるJ_4, J_5となる。

しかし、このように求められた組み合わせを前のステージ（ここではステージ1）までの順序関係につけ加えるだけでは、不十分であると考えられる。なぜなら、新たなステージで決められた順序組み合わせが、前ステージで決められた作業のどの作業の後に、どのようにつけ加えるべきであるかという検討ができないからである。

そこで、ここでは、同じ資源Sであっても資源に番号をつけ種類を区別（S_1, S_2）することにより、その資源の種類も含めた形で新しいステージでの組み合わせを決定していくこととした。その例として、資源制約が2つの時の場合を下に示す。

1)資源S1：J4
　資源S2：J5

2)資源S1：J5
　資源S2：J4

3)資源S1：J4→J5
　資源S2：

4)資源S1：J5→J4
　資源S2：

5)資源S1：
　資源S2：J4→J5

6)資源S1：
　資源S2：J5→J4

このような方法で、各ステージにおいてブランチを行ない、これを経路問題として、動的計画法を用いて解くこととした。このように、あらかじめ各ステージにおける組み合わせを作成しておき、その組み合わせ経路に沿いながら資源に作業を配分していくこととした。

5．2本研究における組み合わせ方法

今回、本研究では従来の手法にかわり、この作業の順序組み合わせ問題を、カットが移動する際に新しく加わってくる作業のみについて図－4に示すような経路に沿って作業を組み合わせていくこととする。ここで新しく加わってくる作業のみについて検討するのは、前ステージまでで含まれる作業については、すでに決定済みであり、カットが移動した際に、前ステージに含まれている作業については検討する必要がないからである。

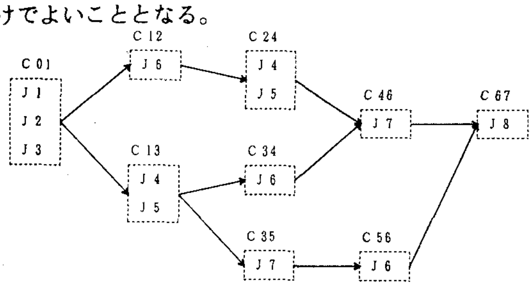
また、その新しく加わってくる作業を図－5のように作業J4→作業J5、作業J5→作業J4のように区別したのは、どの作業を先に順序づけさせるのかを設定させる必要があるからである（以後、このネットワークを作業決定順序ネットワークと呼ぶ）。つまり、先に処理される作業については決定済みであり、この決定済みの作業の組み合わせをもとに、次の作業の順序づけをおこなうためである。

ここで、説明を分かりやすくするために、従来の組み合わせ方法で説明したと同様に、図－3のC_1からC_3へとカットが移動したときをとって図－6を用いて説明することとする。カットがC_1からC_3へ移

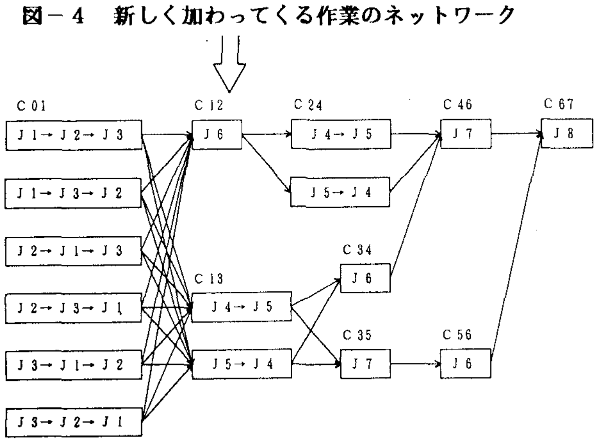
動すると新しく加わってくる作業はＪ4，Ｊ5となる。

作業Ｊ4を先に決定する場合と、作業Ｊ5を先に決定する場合とでは、組み合わされた結果が異なってくることは明確である。また、その組み合わせに引き続き、つぎのステージで仮にカットがC_3からC_5に移り作業Ｊ7が新しい作業として組み合わされることとなると、図－6で示すように全く異なった2つの組み合わせが作成されることとなる。ゆえに、同ステージ内であっても作業の決定させる順序を設定する必要が生じてくるのである。

次に、図－7の組み合わせで示すように作業Ｊ6は作業Ｊ3が終了しないと作業を開始できないという場合について検討することとする。ここで問題となってくるのは、図－7．②の場合である。図－7．①，②の両方の場合も作業Ｊ6は、同時刻に開始されることとなるが、図－7．②の場合は、作業Ｊ3の後に資源に待ち時間を与えてしまうこととなる。これは、次の図－7．③と同等のことを意味していることと考えられる。つまり、図－7．②の組み合わせは、ただ単に非効率的な時間の使い方をしているにすぎず、図－7．①の組み合わせのみについて検討するだけでよいこととなる。



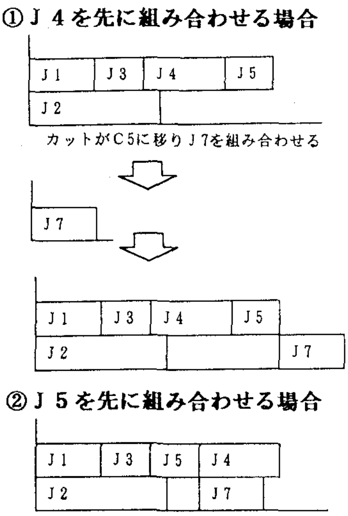
図－4　新しく加わってくる作業のネットワーク



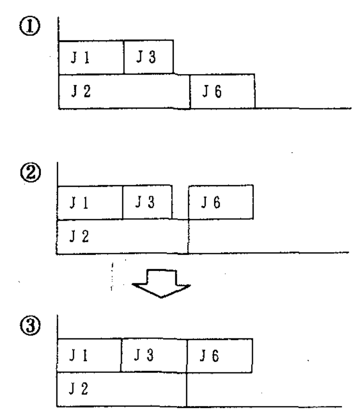
図－5　作業決定順序ネットワーク

そこで、このような図－7のように、同時刻で開始される組み合わせが2つ以上存在する場合では、資源に待ち時間を与える組み合わせに、評価関数としてペナルティーを与え、それ以降の経路の順序の組み合わせについては検討する必要をなくすこととした。

このことにより、ペナルティーを与えられた経路については、それ以降の組み合わせを行なう必要がなくなり、従来の組み合わせ問題よりも各ステージにおける組み合わせ作成をおこなう分の算量が減り、一層合理的かつ効率的に最適解を導くことが可能となった。



図－6　作業決定順序に関する組み合わせ



図－7　ペナルティーに関する組み合わせ

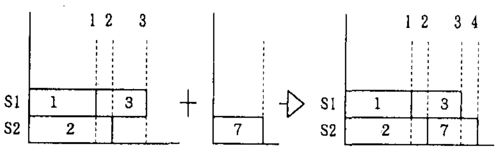
5．3作業組み合わせの定式化

本研究の資源配分問題を行なうにあたり、以上で説明した組み合わせ方法を実行するための定式化について説明することとする。

本研究では、資源配分問題を図－1のフローの流れに示すように、①まず、対象工程ネットワークをもとにカットを作成し、②この抽出されたカット間の関係をトポロジカルに等値変換したカットネットワークを作成し、③このカットネットワークにカットが移動した時に新しく加わってくる作業の決定順序を重ねた作業決定順序ネットワークを作成し、④この作業決定順序ネットワークに沿いながら、対象工程の技術的順序関係および、資源の制約を満たしながら資源に作業を配分させていくものである。そして、本研究の資源配分問題の最適解となる組み合わせを、工期を最小となる組み合わせとし、研究をおこなった。

1）定式化

まず、作業の組み合わせの状況を、図－8のように表わすことにより、組み合わせを行列で表すことができる。



図－8　作業組み合わせ

以下この図－8を例にとって作業の組み合わせの定式化について説明することとする。

まず、図－8の関係を日数で表示した行列＜F＞（以後、割付日数行列と呼ぶ）を作成することができる。

ここで、F(Dk, j)＝［a_{ij}］において

F(Dk, j)＝［a_{ij}］＝D：作業割付日数

	1	2	3	4	5	・・・j・・	n
1	D1	D2	D3	D4	-	・・・・・・・・	-

この行列の列ｊは作業が終了するごとに、全資源に渡り時間断面に区切りを与えた区切りの番号である。F(Dk, j)は、ｊによって区切られた時間である。ここで、Ｄｋは、探索経路番号であり、作業決定順序ネットワークのルートに資源制約を考慮させ資源に割り付ける順序のルートに順序をつけたものである。この割付日数行列は、作業の組み合わせが変化するにしたがい、当然、変化するものである。言い替えれば、全ての組み合わせ（全ての経路）についての割付日数行列を作成すれば、この行列＜Ｆ＞の中には必ず最適解が存在していることとなる。従来型のように、事前に各ステージにおける作業について組み合わせをおこなうことは、非常に非効率的となるため、本研究においては、図－5．2で示すような作業決定順序ネットワークに沿い、制約条件を満たしながら作業を資源に割り付けることとした。以下のⅠ．Ⅱ．Ⅲでは、制約条件および、ペナルティーに関しての定式化を説明することとする。

2）技術的順序関係の制約

F'k＝［a_{ij}］＝Ｘ：作業番号　　（k:経路番号）

	1	2	3	4	5	・・・j・・・	n
1	1	3	3	-	-	・・・・・・・・・	-
2	2	2	7	7	-	・・・・・・・・・	-

図－8を作業番号で表示した行列を＜F'k＞とする。

この＜F'k＞は、技術的順序関係を踏まえた順序により資源を配分させていくための行列である。言い替えれば、この＜F'k＞の行列の列ｊに含まれる作業間には、先行もしくは可達関係を持たせてはいけないということになる。

よって、F'k(i, j)の任意のｊの列ｊに含まれる作業Ｉkjについては、先行可達行列＜Ｍ＞においてＩkjに含まれる作業は、互いに管理的順序関係（M_L（ＩKj）＝０　Ｍ：先行可達行列）でなければならない。ここで、M_L（Ｉkj）＝［a_{ij}］において

$$M_L（Ikj）=［a_{ij}］= 0$$

となる要素には作業Ｉkjを含んでいなければならない。このように、M_L（Ｉkj）において、作業Ｉkjの要素については、管理的な順序関係でなければならない。

3）資源の制約

図－8を資源が活動しているかどうかを行列によって表わすと＜F''k＞となる。ここで、１は作業中

を、0は資源の待ちを示している。ここで、F''k＝［a_{ij}］において

$$F''k = [a_{ij}] = \begin{cases} 0：作業待ち \\ 1：作業中 \end{cases}$$

	1	2	3	4	5	・・・j・・・・	n
1	1	1	1	0	0	・・・・・・・・・	0
2	1	1	1	1	0	・・・・・・・・・	0

この行列は、新しく加わってくる作業を組み合わせる際に、どこの地点に新しい作業を加えればよいのかという問題を解決するために有効となる。

新しく加わってくる作業を

$\sum_{i=1}^{s} F''k(i,\ j) < S$　　S：資源制約数

を満たす列のF''k(i,j)＝0となっていて最もjの値の小さい地点MINjに加えていけばよいこととなる。

しかし、新しく加わってくる作業を、F'k(i,MINj)の地点で活動させることができない場合（作業間に技術的順序関係を持つ場合）には、資源が活動していなくともF''k(i,MINj)の値を1にすることとする。また、F'k(i,MINj)においては、実際には作業を行なっていないのでF'k(i,NINj)＝0にすることとする。このようにすることにより、次の新しい作業を、組み合わせる際に、どの資源のどの地点に加えるべきであるのかを明確にすることができる。

そして、待ちを与えた場合にはペナルティーを与えておく。ここで、ペナルティーとは、資源に待ちを与えたという意味を持たせるものである。

4）ペナルティー

ここで、同時刻で開始される組み合わせが2つ以上存在する場合、資源に待ち時間を与える組み合わせには、評価関数としてペナルティーを与え、それ以降の経路の順序の組み合わせについては検討する必要がないということについて、説明することとする。

まず、新しく加わってくる作業を前のステージまでで決定した組み合わせに組み合わせると、資源制約の数だけの異なった組み合わせが算出されるが、この組み合わせの中で、

F'a≠F'b

F（a,　j）＝F（b,　j）　a,b：経路番号

を満たす組み合わせが存在する場合には、資源に待ちを与える組み合わせにペナルティーを与え、それ以後のこの組み合わせの経路については、検討しないこととなる。

このⅠ，Ⅱの制約条件の処理を踏まえながら、カットネットワークの最終レベルまで新しく加わってくる作業の組み合わせを組み合わせる。

そして、全ての経路について組み合わせをおこなうと

F＝［a_{ij}］

	1	2	3	4	i	n
1	D_{11}	D_{12}	D_{13}	D_{14}	・・・・・・	D_{1n}
2	D_{21}	D_{22}	D_{23}	・	・・・・・・	D_{2n}
3	D_{31}	D_{32}	・	・	・・・・・・	・
i	・	・	・	・	・・・・・・	・
m	D_{m1}	D_{m2}	・	・	・・・・・・	D_{mn}

のような行列となる。

そして、全ての経路Dkに対して

【最短経路問題の定式化】

＜目的関数＞

$$\mathrm{MIN} \sum_{j=1}^{N} F(i,\ j) = \lambda$$

λ：最短工期日数

＜制約条件＞

$$M_L(I_{kj}) = 0 : I_{kj}$$

（作業Ｉｋｊの要素が0であることの意）

$$\sum_{i=1}^{S} F''k(i,\ j) \leq S$$

M_L（Ｉkj）：先行可達行列においてF'ｋ（ｓ，ｊ）の任意のｊの列ｊに含まれる作業群Ｉkjの行ベクトルの要素

Ｓ　：資源制約数

$$\mathrm{MIN} \sum_{j=1}^{n} F(i, j) = \lambda$$

となる日数が最小工期であり、このときの経路の組み合わせが最適な工程となる。

5）動的計画法よる最小工期の探索

本研究では、各ステージに含まれる新しい作業の決定順序を結ぶ経路、作業決定順序ネットワークを、最短経路問題として動的計画法を用いて解くこととした。この経路問題を動的計画法を用いて解くと、最小工期３２日が得られ、図－９に示すような３つの最適工程計画案が作成された。この計画案については、従来の手法と同様となり、今回の資源配分問題の理論的開発を証明したことと考える。

６．他のモデルへの対応

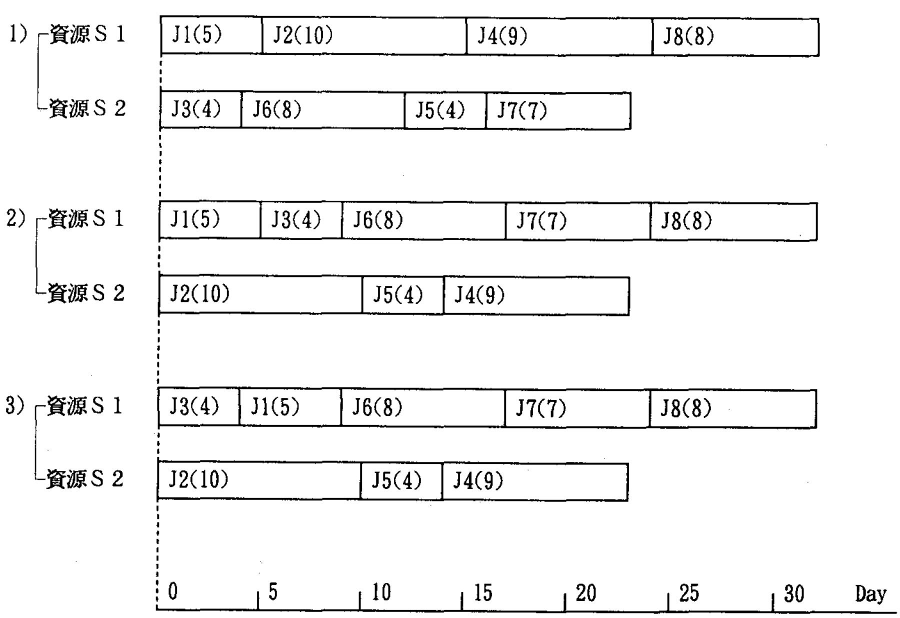
今回開発した手法を用いて、別のモデル（図－１０）に対して検討することとする。資源制約数は、上述のモデルと同様に、２とする。このモデルのカットネットワークは、図－１１に示すようなネットワークとなり、このカットネットワークに沿いながら、最小工期を目的として、経路を探索した。その結果、図－１２のような工期が３５日の最適解が得られた。

７．おわりに

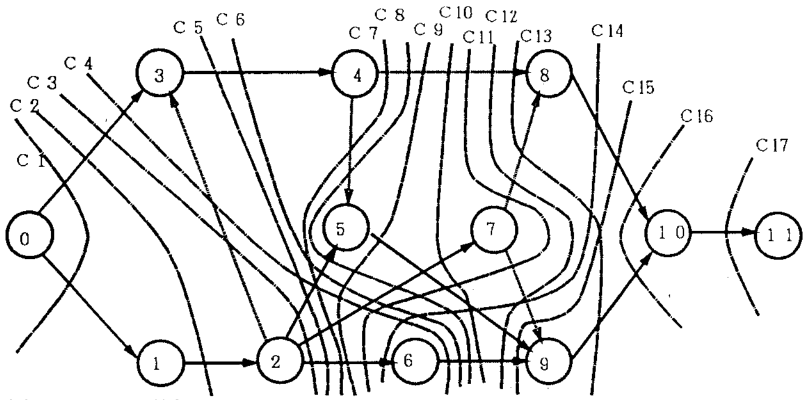
従来から考えられていたグラフ理論を用いた、カット探索方法に変わる、より合理的・効率的なカットの探索方法の開発を行った。特に、本研究では、工程計画における初期条件である作業リストだけから直接的にカットを求める方法を開発した。これによって、実務レベルの工程においてもより効率的にカットの探索が可能になるものと考える。それに、作業の組み合わせ問題の解法において、ネットワークトポロジーにもとづくカット概念を組み合わせることによって、作業の組み合わせ操作を極力減らし、迅速に最適解を導くことが可能となった。また、このことによって、資源量の制約のもとで最小工期を与えるような工程計画案の作成を可能となった。

また、今回は、科学的手法を取り入れた最適工程計画に関する研究ということで、実際の工事に適用していくためには、数多くの検討が必要である。

とくに、本研究における資源割付問題は、各アクティビティに共通な１種類の資源についてのみ検討を行なったが、今後は複数種の資源にも対応できるようなモデルに拡張していく必要があると考える。



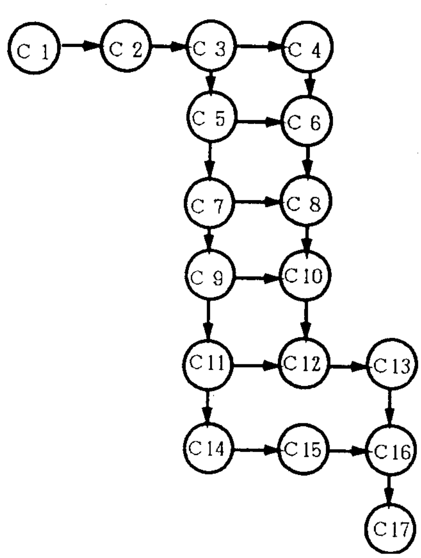
図－９　最適工程計画案　　（）内は作業所要日数



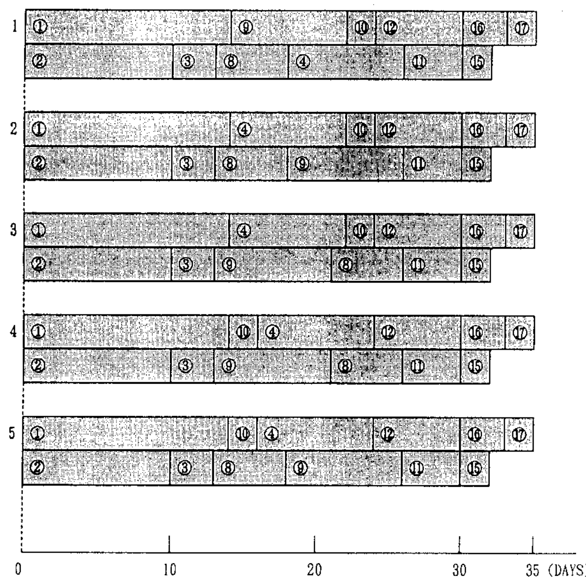
図－１０　工程ネットワークとそのカット

＜作業日数リスト＞

作業ノード（ｉ，ｊ）	作業番号	作業日数（日）
0，3	1	1 4
0，1	2	1 0
1，2	3	3
3，4	4	8
2，3	5	0
4，8	6	0
4，5	7	0
2，5	8	5
2，7	9	8
2，6	1 0	2
5，9	1 1	4
6，9	1 2	6
7，8	1 3	0
7，9	1 4	0
8，1 0	1 5	2
9，1 0	1 6	3
1 0，1 1	1 7	2



図－１１　工程ネットワークのカットネットワーク



図－１２　最適工程計画案

【参考文献】

1)春名　攻，原田　満，荒川　和久：最適工程計画を目指したスケジューリングモデルの開発研究，第９回建設マネジメント問題に関する研究発表・討論会講演集，1991，12

2)石原　藤次郎 校閲，吉川　和広 著：土木計画とＯＲ，丸善

3)吉川　和宏　編著：土木計画学演習，森北出版

4)James J. O'Brien：CPM in Construction management，McGraw-Hill Book Company

5)R. Bellman, S. Dreyfus:Applied Dynamic Programming ，Princeton University Press

6)E. L. Lawler, D. E. Wood:Branch and Bound Method :A. Surver, JO