IV-59　　ネットワークプランニングにおける資源配分計画

京都大学工学部　正員　吉川和広
京都大学工学部　学生員　○春名　攻
中央復建コンサルタンツ　正員　滝田憲二

1. はじめに　ネットワーク手法による工程計画へのシステムアプローチでは、作業の流れを時間的な特性をもつグラフの構造として把握することが最も本質的な、かつ出発となるプロセスを構成している。また、この流れの構造の基礎となるのは、作業間の順序関係である。さて、現在のネットワーク手法が開発された大規模装置産業においては、上述の作業の流れは生産における技術的な側面ならびに機械設備の位置的な関係から、ほぼ一意的に定められる。当然のこととして、工程ネットワークも一意的に決定され、従って、このプロセスから導びかれるネットワークに対し、インプットデータ（Activity Duration Time）を用いてスケジュール計算を行ない、この結果に基づいて作業の管理を行なっていったとしても殆んど問題はない。建設工事においても、技術的な側面から決定される施工順序は、殆んどの場合、一意的に定まると仮定してよい。つぎに、実際の工事を施工していく場合に、一般的には施工対象である構造物あるいは施工場所をいくつかのブロックに分割し、しかる後に、これらの各ブロックに対し、上述の技術的順序関係を用いて順次建設し、最終的に完成品を生み出すという手段がとられる。ここでは、ブロック間の作業順序は主として、機械、作業要員、その他の資源的な制約に基づく順序関係によって規定される。従って、上述の分割単位や、ブロック間の作業の施工順序によって、全体的な作業順序関係は大きく変わり得るものであるといえる。以下においては、資源としてm種の機械を1台ずつ使用する場合を例にとり、この順序関係を決定するために、管理システムからみた1つの目的を設定し、合目的な順序関係をだめるためのモデルの策定について述べることにする。

2、作業順序決定モデル　一般の建設工事においては、その中心的な管理対象を時間とすることが多いが、ここでも、目的関数をプロジェクト完了時間λと設定し、これを最小化することを目的とする。すなわち、λを最小とするような、ネットワークならびにスケジュールを最適解とする。さらにここでは、Planning と Scheduling を、1つのプロセスで同時にとりあつかい、合目的な形で、Planning と Scheduling を同時決定しようとするものである。

a、定式化　先述したように、技術的な側面からみた作業の順序関係は、種々の検討を加えることにより殆んど一意的に決まるが、これを技術的順序関係マトリックスと呼び、(p_{ij}^T)であらわすこととする。ここでp_{ij}^Tは、

$$p_{ij}^T=\begin{cases}1, & i\ll j \text{（}i\text{が}j\text{の直接的先行作業であるとき）}\\ 0, & i\not\ll j \text{（その他の場合）}\end{cases} \tag{1}$$

とあらわす。つぎに、技術的な順序関係以外の要因で決まる順序関係があるが、ここでは、m種(各1台)の施工機械によって決められる順序関係を例としてとりあげる。いま、これをマトリックス(p_{ij}^g) $(g=1,2,\cdots,m)$とあらわすと、p_{ij}^gもp_{ij}^Tと同様に、次のように示される。

$$p_{ij}^g=\begin{cases}1, & i\ll j \text{ かつ } i,j\in J_g\\ 0, & i\not\ll j \text{ あるいは } i,j\notin J_g \ (g=1,2,\cdots,m)\end{cases} \tag{2}$$

ここで、J_gは機械gを必要とする作業集合とする。（以下においては、J_gを競合作業集合と呼ぶ。）
従って、工程における順序関係（p_{ij}）は、

$$(p_{ij}) = (p_{ij}^T) + \sum_{g=1}^{m} (p_{ij}^g) \qquad (3)$$

とあらわすことができるが、p_{ij}は、上式における加法演算の規則を

$$\left.\begin{array}{ll} 1+1=1, & 1+0=1 \\ 0+0=0, & 0+1=1 \end{array}\right\} \qquad (4)$$

とすると、p_{ij}は、

$$p_{ij} = \begin{cases} 1, & i \ll j \\ 0, & i \not\ll j \end{cases} \qquad (5)$$

となり、p_{ij}^T, p_{ij}^gと全く同様の表現を行なうことができる。

b. 制約条件　以上のような定義を行なえば、制約条件は次式のように示される。

$$\left.\begin{array}{llll} 1.\ p_{ij}^g = 1 \text{ or } 0, & i,j \in J_g & 4.\ p_{ii}^g = 0, & i \in J_g \\ 2.\ p_{ij}^g \cdot p_{ji}^g = 0, & i,j \in J_g & 5.\ \sum_{i \in J_g} p_{ij}^g = \sum_{j \in J_g} p_{ij}^g = 1 & \\ 3.\ p_{ij}^g = 0, & i \text{ or } j \bar{\in} J_g & 6.\ U_i^g - U_j^g - n_g p_{ij}^g \leq n_g - 1 & \end{array}\right\} \qquad (6)$$

ただし、ここでは1つの仮想の作業$j_g^* \in J_g$を想定することが必要であり、作業j_g^*の所要時間を0として与えることが必要である。さて、変数を$\{(p_{ij}^g)\}$, $\{U_i^g\}$とし、制約条件(6)式の下でλを最小にするような$\{(p_{ij}^g)\}_{opt.}$を求めることが、ここでの問題となる。この問題にDynamic Programmingを用いて定式化を行ない、解法について論ずる。

3. Dynamic Programmingによる解法　まず始めにn_gの競合作業よりなる競合作業集合J_gをつぎのように定義する。

$$J_g = \{j_1^g, j_2^g, \cdots, j_l^g, \cdots, j_{n_g}^g\}, \quad J_g \cap J_{g'} = \Phi \ (\text{empty}), (g \neq g') \qquad (7)$$

つぎに、D.P.計算を行ないやすくするために、配分計算のステップをつぎのように定める。すなわち、次式のように、$\{J_g\}$を要素とする順列J_Mを定める。

$$J_M = (J_1, J_2, \cdots\cdots, J_m) \qquad (8)$$

このようにして順列J_Mを求めれば、自然数（$k=1,2,\cdots\cdots,N$）を上述のJ_Mの要素j_l^gに対応させることができる。すなわち　$J_M = (j_1, j_2, \cdots\cdots, j_k, \cdots\cdots j_N)$,　$N = \sum_{g=1}^{m} n_g$　(9)

従って、D.P.の配分計算のステップは$k=1,2,\cdots\cdots N$となる。このように配分stepを決定すると、$p_{ij_k}^g = 1$ ($i, j_k^g \in J_g$)とおいたときのプロジェクトの完了時刻の増分$\Delta\lambda_i^{(k)}$は、

$$\left.\begin{array}{l} \Delta\lambda_i^{(k)} = \max\left[ES_{j_k}^{(k)} - ES_{j_k}^{(k-1)} - TF_{j_k}^{(k-1)},\ 0\right] \\ \text{ここで、} ES_{j_k}^{(k)} = \max\left[ES_{j_k}^{(k-1)},\ EF_i^{(k-1)}\right] \\ \quad i, j_k \in J_M, \quad k = 1, 2, \cdots, N \end{array}\right\} \qquad (10)$$

で与えられる。ここで、$ES_{j_k}^{(k)}$, $EF_i^{(k-1)}$, $TF_{j_k}^{(k-1)}$はそれぞれ、step k, $k-1$で求められる最早開始時刻、最早完了時刻およびトータルフロートをあらわしており、各作業の所要時間と（p_{ij}）を与えれば容易に計算できる。従って、$\lambda^{(k)}$をstep kに対応するプロジェクト完了時刻とすると、

$$\left.\begin{array}{ll} \lambda^{(0)} = \lambda^T & ((p_{ij}) = (p_{ij}^T)\text{としたときのプロジェクト完了時刻}) \\ \lambda^{(k)} = \min\limits_{i \in J_M} \{\lambda^{(k-1)} + \Delta\lambda_i^{(k)}\} & \end{array}\right\} \qquad (11)$$

と表わされ、式(3)(4)(6)(10)の下で逐次計算によって$\lambda_{opt.}$ $\{(p_{ij}^g)\}_{opt.}$が求められる。