〔農工研技報　212
1 ～ 12，2012〕

パイプライン非定常流の剛性モデル・閉路解析における全域木と補木の辺の決定支援方法

田中良和・中田　達・樽屋啓之

目　次

Ⅰ　緒　言

受益地の水利用計画に合致した合理的な水管理を行うために，全国各地の幹線用水路のパイプライン化が進められて久しい。さらに時代は遷移し，近年の農地利用集積，水田畑作の進展，高温障害への対応などの営農変化に伴って用水需要が変化しつつある。そのために，多くの送配水パイプラインシステムは当初の水利用計画との齟齬を来し，難しい水管理を強いられている。用水需要に弾力的に対応する方法の一つとして，中間調整池の新設やため池，地下水などの既存水源を再活用した送配水パイプラインシステムへの更新を検討することが考えられる。送配水パイプラインシステムの設計，運用管理には，管内圧力水頭や管内平均流速の制御が重要であり，これらの検討には数値解析の果たす役割が大きい。送配水パイプラインシステム内におけるバルブやポンプの操作に伴う急激な水理過渡現象を把握するためには，弾性体モデルによる数値解析が行われる。しかし，この手法はコンピュータの CPU の使用時間が長いものであるため，中間調整池や既存水源と接続した送配水パイプラインシステムの流量輸送の緩やかな過渡現象を長時間にわたって解析するには不向きである（内藤ら，1983）。鬼塚（1971）は緩やかな過渡水理現象の数値解析手法として剛性モデル理論による非定常流解析が有効であることを明らかにした。しかしながら，現在，この手法は十分に活用されていない。その理由として，送配水パイプラインシステムの状態方程式を求めることが非常に面倒であることが挙げられる（鬼塚，1998）。以下に，この面倒な作業を解決するために既往の研究において行われた成果を整理してみる。

剛性モデル理論による非定常流解析とは，管路系に発生する緩やかな非定常水理現象であるサージング現象を，水の圧縮性や管体の弾性変形を考慮しない剛体水柱理論で近似できる範囲の現象として定式化し，数値解析するものである（鬼塚，1981）。剛性モデル理論による非定常流解析として，接続解析（島田，1991）と閉路解析（鬼塚，1977）が知られている。

接続解析は，パイプラインシステムの流量と運動量の連続性を表す 1 階の連立常微分方程式（システム方程式と呼ぶ）から状態方程式を導き，ルンゲ・クッタ法などによって時間積分を行う手法である。内藤ら（1983）は，パイプラインシステム内のある水槽から他の水槽まで運動方程式を管の接続状態に従って足し合わせて経路を消去する操作（消去経路と呼ぶ）を提案した。これは，操作後の経路の数が独立変数の数と等しくなることを利用して，状態方程式の独立変数を決定し，足し合わせた経路の情報を元に状態方程式を導出する方法である。しかし，大規模なパイプラインシステムでは接続情報が複雑になるため，そもそも消去経路の操作を技術者が行うこと自体が煩雑な作業であり，機械的な操作が必要とされた。そこで，島田（1991）は，パイプラインシステムでは流量の従属変数の個数が流量の境界条件の個数と等しい特性を利用して，独立変数を決定する方法を提案し，流量の独立変数と接続情報から状態方程式を導出する方法を提案した。

他方，閉路解析は接続解析よりも技巧的な作業が必要である。つまり，システム方程式の導出には，エネルギー基準面との仮想的な経路をパイプラインシステムに追加する作業が必要である。しかし，パイプラインシステムをモデル化したシステムグラフについて全域木と補木の辺の情報を得ることができれば，自動的に流量の独

水利工学研究領域　水路システム担当
平成 23 年 11 月 4 日受理
キーワード：パイプライン，グラフ理論，状態方程式，非定常流解析，Ternary network flow 法

立変数と従属変数が判明し，独立変数と同じ数の状態方程式が得られる利点がある。ただし，大規模なパイプラインシステムでは閉路情報を得る作業は煩雑になり，この解析手法を技術者が利用しにくい状況であった。例えば，パイプラインシステムにおいて末端バルブを完全閉鎖するときの過渡現象を解析する際には，バルブ開度を絞った際に stiff な状態方程式になる問題があるため，末端バルブを完全閉鎖する直前に，末端バルブと上流側の節点間の接続を切断する対処方法が行われる（内藤ら，1983：島田，1988）。閉路解析では，接続関係を変更したことに伴い閉路情報を更新する作業が必要であった。本稿では，全域木探索法と Ternary network flow 法（Doris and Stephan, 1981）を用いて閉路情報を素早く得る方法を提案して，技術者が剛性モデル理論による閉路解析を容易に解析できるように整備することを目的とした。

Ⅱ　剛性モデル理論の概説

剛性モデル理論についての説明として，単一管路における運動方程式の導出例から始めることにする。

水の圧縮性と管体の弾性変形を無視すると，**Fig.1** に示すように，上流端と下流端の断面の位置が i と j である単一管路 k の運動方程式は次式になる（鬼塚，1982）。

$$\frac{l}{g}\dot{v}_k = \left(\frac{p_i}{\omega} + z_i + \frac{v_i^{\,2}}{2g}\right) - \left(\frac{p_j}{\omega} + z_j + \frac{v_j^{\,2}}{2g}\right) - f_j \frac{l_j}{d_j}\frac{v_j^{\,2}}{2g} \quad (1)$$

ただし，v_k のドットは単一管路 k の流速についての時間微分であることを表している。また，l_k は管路延長，g は重力加速度，p_i と p_j は圧力強度，ω は水の単位重量（$\omega = \rho g$），z_i と z_j は基準面から上流端と下流端までの管軸高さ，v_k は管内平均流速，f_k は摩擦損失係数，d_k は管の直径である。

上流端 i と下流端 j の地点における全水頭 h_i と h_j は（2）と（3）式で表されるので，（1）式は（4）式に書き直すことができる。

$$h_i = p_i + z_i + \frac{v_k^{\,2}}{2g} \quad \cdots\cdots (2)$$

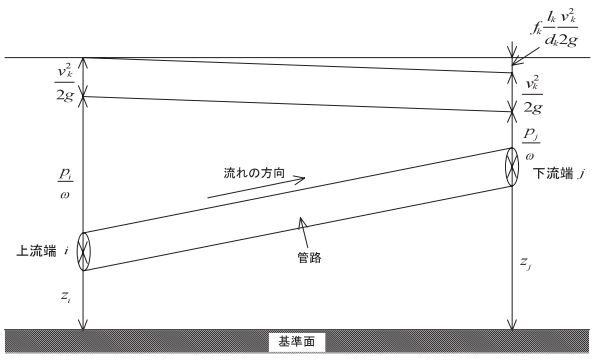

Fig.1　単一管路におけるエネルギーの模式図
Schema of various energies in single pipeline

$$h_j = p_j + z_j + \frac{v_k^{\,2}}{2g} \quad \cdots\cdots (3)$$

$$\frac{l}{g}\dot{v} = \left(h_i - h_j\right) - f_k \frac{l_k}{d_k}\frac{v_k^{\,2}}{2g} \quad \cdots\cdots (4)$$

分岐がなく管路断面積が変化する複数管路は，管径の異なる単一管路が直列に接続したパイプラインシステムであるため，システム方程式は，各単一管路において異なる流速 v_k で表記せずに一定である流量 q_k を用いれば表記が簡単になる。さらに，管路内の流れが逆向きに流れる場合にも対処するために，流れの向きを特定の方向に決めておき，その方向を正とし，逆方向を負としておくと，（5）式で表すことができる。

$$\frac{l}{A_k g}\dot{q}_k = \left(h_i - h_j\right) - f_k \frac{l_k}{d_k}\frac{|q_k|q_k}{2gA_k^{\,2}} \quad \cdots\cdots (5)$$

ここで，q_k のドットは単一管路 k の流量 q についての時間微分であることを表している。また，A_k は管路断面の面積を表している。

さらに，単一管路 k の慣性定数を L_k，流量抵抗係数をダルシー・ワイズバッハの式に基づいて K_k とすると，（5）式は（6）式に変形できる。

$$L_k \dot{q}_k = \left(h_i - h_j\right) - K_k |q_k| q_k \quad \cdots\cdots (6)$$

$$L_k = \frac{l_k}{A_k g} \quad , \quad K_k = \frac{f_k l_k}{2gd_k A_k^{\,2}} \quad \cdots\cdots (7)$$

パイプラインシステムの非定常流解析では，システム内の節点間や節点と境界条件を表す点との間の各単一管路区間について水の慣性効果を表す運動方程式を立て，各々の節点における流量の連続式を組み合わせることによってシステム方程式を作成する。

浜口・鬼塚（1980）は，末端バルブの開閉操作による過渡現象について，剛性モデルと水の圧縮性や管体の弾性変形を考慮して弾性体と見なした弾性体モデルとについて比較を行っており，剛性モデルは，弾性体モデルよりも非常に大きくて急激な圧力上昇が発生するために圧力変動については慎重な取り扱いが必要であるが，流量変動については波動モデルによる結果の平均値となるために適用範囲が高いことを明らかにしている。よって，剛性モデルは送配水パイプラインシステムの流量輸送の穏やかな過渡現象を解析するのに利用できる。

Ⅲ　閉路解析の概説と問題点の整理

剛性モデル・閉路解析において，パイプラインシステムの接続関係をシステムグラフとして見た立場から全域木と補木の辺を指定できれば，容易に状態方程式を導出できることを具体例を挙げて説明する。手順として，はじめにグラフ理論に基づいてパイプラインシステムを複数の閉路で表す方法を説明する。次に，システムグラフ

の辺を全域木と補木に分解すれば，接続情報と閉路情報が得られることを説明する。最後に，接続情報と閉路情報を利用した流量連続とエネルギー連続条件から状態方程式を導出する。この導出過程を通じて，全域木と補木の辺を指定することをプログラムにより支援することが必要であることを説明する。

1　パイプラインシステムのモデル化

パイプラインシステムの接続情報や閉路情報の構造はグラフ理論に基づく回路図で表すことができる（鬼塚，1998）。回路図は節点とそれらを連結する辺で構成されており，1 方向の辺で構成されるグラフを有向グラフという。具体的に **Fig.2** のようなパイプラインシステムをモデル化して，以後，状態方程式の導出に利用する。ここで，水頭の基準面を記号 D で表し，既知の水頭境界を H_0，H_5，中間調整池の水頭を h_1，ファームポンドの水頭 h_4，既知の流量境界を q_{10}，その他の q_i を管の流量とした。

パイプラインシステムを有向グラフで表したシステムグラフは **Fig.3** である。節点は，①ファームポンドや中間調整池など未知の水頭をもつ水槽，②既知の流量境界，③既知の水頭境界を表す境界条件を含む。このシステムグラフを作図するには，境界条件となる節点と基準面 D とを辺でつなげる作業が必要である。この作業によって，グラフは複数の閉路（システムグラフ内のある節点を出発して幾つかの節点を経由して，出発した節点に戻った時の経路）で構成されることになる。

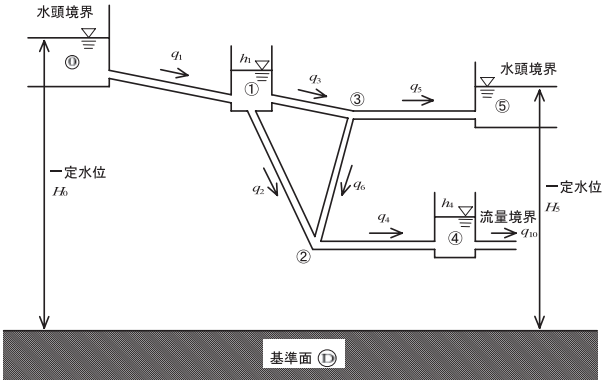

Fig.2　パイプラインシステム例
Example of pipeline system

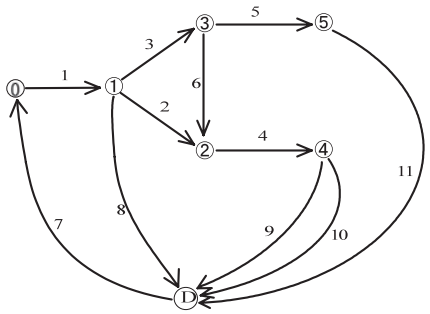

Fig.3　パイプラインシステムのシステムグラフ
System graph of pipeline system

システムグラフは全域木の辺と補木の辺に分解することができる。ここで，全域木とは，閉路を 1 つも持たないが，すべての点に接続している辺の組み合わせである。他方，補木とは全域木の辺に追加することによって閉路を作ることができる辺の集合である。この補木の辺の数は，システムグラフを構成する辺の数を n 本，節点の数を m 個とすると，$(n - m + 1)$ 本である。

ここで，補木の辺を $(n - m + 1)$ 本だけ指定して，それぞれを一本だけ含む閉路を考えてみる。**Fig.3** では，辺の数 11 本，節点の数 7 個であるので，補木の辺の数は 5 本である。全域木を辺 {3，4，7，8，9，11} としたとき，補木は辺 {1，2，5，6，10} になり，補木の辺を 1 本だけ含むように閉路を選ぶと以下の通りになる。ここで，負号は全域木の辺の向きが閉路の向きと逆方向であることを示している。**Fig.4** には，全域木と補木の辺をそれぞれ点線と実線で描画し，閉路を構成する辺の経路を細い実線で記している。

補木の辺 1 を指定した場合の閉路 1：{補木 1，全域木 8，全域木 7}
補木の辺 2 を指定した場合の閉路 2：{補木 2，全域木 4，全域木 9，－（全域木 8)}
補木の辺 5 を指定した場合の閉路 3：{補木 5，全域木 11，－（全域木 8)，全域木 3}
補木の辺 6 を指定した場合の閉路 4：{補木 6，全域木 4，全域木 9，－（全域木 8)，全域木 3}
補木の辺 10 を指定した場合の閉路 5: {補木 10，－（全域木 9)}

剛性モデル・閉路解析では，補木の辺を 1 本だけ含むように全域木の辺と接続した閉路においてエネルギーが保存されていることを利用して運動方程式を立て，各々の節点や境界条件を表す点における流量の連続式を組み合わせることによってシステム方程式を作成する。よって，システム方程式を構成するある 1 つの運動方程式では，流量の独立変数は閉路に 1 本だけ含まれる補木の辺の流量とする必要がある。つまり，補木の辺と全域木の辺を決定することは，辺に割り当てる流量をそれぞれ独立変数と従属変数とに指定していることに他ならない。

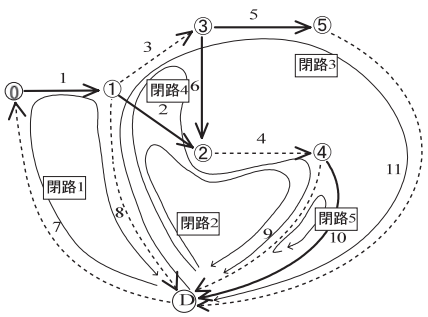

Fig.4　全域木と補木の辺の決定によって得られた閉路情報
Closed circuit information got by the decision of edges of spanning tree and cotree

補木の辺に指定する際には，節点から基準面 D へ接続するように追加した辺については，指定方法が様々であるので, 以下に箇条書きすることに注意が必要である。

①未知の水頭をもつ水槽から基準面 D へ接続するように追加した辺は全域木の辺にする。
なぜなら，この辺は水槽における流量の連続性を満たすために，水槽の貯留量変化を流量と見なすことによって得られた辺である。この流量は水槽の水位によって変化するため，独立変数になり得ないためである。

②既知の流量境界を表すように基準面 D へ追加した辺は必ず補木の辺にする。
なぜなら，既知の流量境界は取り出し流量を与える必要があるため，この流量は従属変数には成り得ないためである。

③既知の水頭境界から基準面 D へ接続するように追加した辺は全域木の辺にする。
なぜなら，既知の水頭境界を表す節点における流入量は, 流出量によって連続条件が決まるためである。

2　基本閉路行列の定義

補木の辺を 1 本しか含まない閉路を基本閉路という。よって，補木の辺の数と基本閉路の数は等しい。**Fig.2** において補木の辺 {1，2，5，6，10} を選択して行列の成分 b_{ij} の値を次のとおりに分類すると，(8) 式の基本閉路行列 **B** を作成できる。

$$b_{ij}=\begin{cases} 1 & \cdots \text{辺 } j \text{ が閉路 } i \text{ に含まれ，かつ，同じ向きの場合。} \\ -1 & \cdots \text{辺 } j \text{ が閉路 } i \text{ に含まれ，かつ，逆向きの場合。} \\ 0 & \cdots \text{辺 } j \text{ が閉路 } i \text{ に含まれない場合。} \end{cases}$$

$$\mathbf{B}=\begin{array}{c} \text{辺}\ 3\ \ 4\ \ 7\ \ 8\ \ 9\ \ 11\ \ 1\ \ 2\ \ 5\ \ 6\ \ 10 \\ \left[\begin{array}{cccccc|ccccc} 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 1 \end{array}\right] \\ \qquad B_t \qquad\qquad B_c \end{array} \begin{array}{l} \\ \text{閉路1} \\ 2 \\ 3 \\ 4 \\ 5 \end{array}$$

……（8）

ここで，**B** は行方向（横方向）に全域木の辺を若い番号順に並べ，次に，補木の辺を若い番号順に並べ替えている。さらに，**B** の列方向（縦方向）の大きさは基本閉路の個数であり，その並べ方は基本閉路に含まれる補木の辺の若い順である。つまり，**B** を構成する木の部分を行列 B_t，補木の部分を行列 B_c で表すと，$\mathbf{B}=[B_t,\ B_c]$ で表せる。

3　基本カットセット行列の定義

B_t の転置行列をマイナスにした行列 $-B_t^{\mathrm{T}}$ を全域木の辺の個数の大きさの単位行列 I に追加した行列 $\mathbf{A}=[I,\ -B_t^{\mathrm{T}}]$ は，次式である。**A** は基本カットセット行列と呼ばれている。基本カットセット行列の各行は個々の基本カットセットである。1 つの基本カットセットは，1 つのシステムグラフを 2 つに分離する辺の集合であり，全域木の辺を 1 つだけ含んでいる。**A** の行方向（横方向）の順番は **B** と同じであるが, 列方向（縦方向）は基本カットセット内に含まれる全域木の辺が若い番号順に並べている。

$$\mathbf{A}=\begin{array}{c} \text{辺}\ 3\ \ 4\ \ 7\ \ 8\ \ 9\ \ 11\ \ 1\ \ 2\ \ 5\ \ 6\ \ 10 \\ \left[\begin{array}{cccccc|ccccc} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & -1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & 0 & -1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 \end{array}\right] \\ \qquad I \qquad\qquad -B_t^{\mathrm{T}} \end{array} \begin{array}{l} \\ \text{カットセット1} \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{array}$$

……（9）

ここで，成分 a_{ij} の値は，以下のような意味がある。

$$a_{ij}=\begin{cases} 1 & \cdots \text{辺 } j \text{ が基本カットセット } i \text{ に含まれ，かつ, 向きが全域木と同じである時。} \\ -1 & \cdots \text{辺 } j \text{ が基本カットセット } i \text{ に含まれ，かつ, 向きが全域木と反対である時。} \\ 0 & \cdots \text{辺 } j \text{ が基本カットセット } i \text{ に含まれていない時。} \end{cases}$$

4　流量連続条件

状態方程式は，システムグラフの **A**，**B** を元にして，流量とエネルギーの連続性から導出される。

パイプラインシステム内の辺の流量ベクトル **Q** は次式で表すことができる。行方向の成分の順番は **B** と同じである。

$$\mathbf{Q}=[q_3\ \ q_4\ \ q_7\ \ q_8\ \ q_9\ \ q_{11}\ \ q_1\ \ q_2\ \ q_5\ \ q_6\ \ q_{10}] \quad \cdots\cdots(10)$$

ただし，水槽 1，4 と基準面 D とをつなぐ辺の流量 q_8, q_9 は，貯留変化量であるので，水槽 1，4 の水面の面積を A_1，A_4 とし，未知の水頭の記号 h の上のドットは時間微分を表すと，次式になる。

$$\left.\begin{array}{l} q_8=A_1\dot{h}_1 \\ q_9=A_4\dot{h}_4 \end{array}\right\} \quad \cdots\cdots(11)$$

よって，流量ベクトル **Q** は，

$$\mathbf{Q}=[q_3\ \ q_4\ \ q_7\ \ A_1\dot{h}_1\ \ A_4\dot{h}_4\ \ q_{11}\ \ q_1\ \ q_2\ \ q_5\ \ q_6\ \ q_{10}] \quad \cdots\cdots(12)$$

流量の連続条件は，$\mathbf{AQ}^{\mathrm{T}}=0$ であるので，以下のシステム方程式が導かれる。

$$\left.\begin{aligned} A_1\dot{h}_1 &= q_1 - q_2 - q_5 - q_6 \\ A_4\dot{h}_4 &= q_2 + q_6 - q_{10} \end{aligned}\right\} \quad \cdots\cdots(13)$$

同時に，0，2，3 および 5 の節点における流量条件が導かれる。

$$\left.\begin{aligned} q_3 &= q_5 + q_6 \\ q_4 &= q_2 + q_6 \\ q_7 &= q_1 \\ q_{11} &= q_5 \end{aligned}\right\} \quad \cdots\cdots(14)$$

（13）式を行列形式に書き直すと次式になる。

$$\begin{bmatrix} \dot{h}_1 \\ \dot{h}_4 \end{bmatrix} = \begin{bmatrix} \frac{1}{A_1} & 0 \\ 0 & \frac{1}{A_4} \end{bmatrix} \begin{bmatrix} q_1 - q_2 - q_5 - q_6 \\ q_2 + q_6 - q_{10} \end{bmatrix} \quad \cdots\cdots(15)$$

5　エネルギー連続条件

損失水頭ベクトル $\Delta\mathbf{H}$ は次式で表すことができる。ただし，行方向の成分の順番は $\mathbf{B}$ と同じである。

$$\Delta\mathbf{H} = \begin{bmatrix} h_1 - h_3 & h_2 - h_4 & -H_0 & h_1 & h_4 & H_5 & H_0 - h_1 \\ & h_1 - h_2 & h_3 - H_5 & h_3 - h_2 & h_4 - 0 \end{bmatrix} \quad \cdots\cdots(16)$$

節点 i と節点 j における単一管路 k についての運動方程式はⅡ章の（6）式と同じ式で表されるので，損失水頭ベクトルは次式になる。

$$\Delta\mathbf{H} = \begin{bmatrix} L_3\dot{q}_3 + K_3|q_3|q_3 \\ L_4\dot{q}_4 + K_4|q_4|q_4 \\ -H_0 \\ h_1 \\ h_4 \\ H_5 \\ L_1\dot{q}_1 + K_1|q_1|q_1 \\ L_2\dot{q}_2 + K_2|q_2|q_2 \\ L_5\dot{q}_5 + K_5|q_5|q_5 \\ L_6\dot{q}_6 + K_6|q_6|q_6 \\ h_4 \end{bmatrix} \quad \cdots\cdots(17)$$

エネルギーの連続条件 $\mathbf{B}\Delta\mathbf{H}^{\mathrm{T}} = 0$ における従属変数の流量に（16）式を代入すると，独立変数の流量について，以下のシステム方程式が得られる。

$$\begin{bmatrix} L_1 & 0 & 0 & 0 \\ 0 & L_2+L_4 & 0 & L_4 \\ 0 & 0 & L_3+L_5 & L_3 \\ 0 & L_4 & L_3 & L_3+L_4+L_6 \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_5 \\ \dot{q}_6 \end{bmatrix} = \begin{bmatrix} H_0 - h_1 \\ h_1 - h_4 \\ h_1 - H_5 \\ h_1 - h_4 \end{bmatrix} - \begin{bmatrix} K_1' & 0 & 0 & 0 \\ 0 & K_2'+K_4' & 0 & K_4' \\ 0 & 0 & K_3'+K_5' & K_3' \\ 0 & K_4' & K_3' & K_3'+K_4'+K_6' \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \\ q_5 \\ q_6 \end{bmatrix} \quad \cdots(18)$$

ただし，

$$\left.\begin{aligned} K_1' &= K_1|q_1| \\ K_2' &= K_2|q_2| \\ K_3' &= K_3|q_3| = K_3|q_5 + q_6| \\ K_4' &= K_4|q_4| = K_4|q_2 + q_6| \\ K_5' &= K_5|q_5| \\ K_6' &= K_6|q_6| \end{aligned}\right\} \quad \cdots\cdots(19)$$

（18）式の左辺の L_i（i=1，2，5，6）で構成される行列を **L** 行列と呼ぶことにする。

$$\mathbf{L} = \begin{bmatrix} L_1 & 0 & 0 & 0 \\ 0 & L_2+L_4 & 0 & L_4 \\ 0 & 0 & L_3+L_5 & L_3 \\ 0 & L_4 & L_3 & L_3+L_4+L_6 \end{bmatrix} \quad \cdots\cdots(20)$$

L 行列の逆行列 $\mathbf{L}^{-1}$ を（18）式の両辺にかけると一般的な形式は次式になる。

$$\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_5 \\ \dot{q}_6 \end{bmatrix} = \mathbf{L}^{-1} \begin{bmatrix} H_0 - h_1 \\ h_1 - h_4 \\ h_1 - H_5 \\ h_1 - h_4 \end{bmatrix} - \mathbf{L}^{-1} \begin{bmatrix} K_1' & 0 & 0 & 0 \\ 0 & K_2'+K_4' & 0 & K_4' \\ 0 & 0 & K_3'+K_5' & K_3' \\ 0 & K_4' & K_3' & K_3'+K_4'+K_6' \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \\ q_5 \\ q_6 \end{bmatrix} \quad \cdots\cdots(21)$$

6　状態方程式の導出

したがって，導出した（15）式と（21）式をまとめた（31）式は，力学系の状態方程式表示と呼ばれる現代制御理論における標準的な形式である。

$$\left.\begin{aligned} \begin{bmatrix} \dot{h}_1 \\ \dot{h}_4 \end{bmatrix} &= \begin{bmatrix} \frac{1}{A_1} & 0 \\ 0 & \frac{1}{A_4} \end{bmatrix} \begin{bmatrix} q_1 - q_2 - q_5 - q_6 \\ q_2 + q_6 - q_{10} \end{bmatrix} \\ \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_5 \\ \dot{q}_6 \end{bmatrix} &= \mathbf{L}^{-1} \begin{bmatrix} H_0 - h_1 \\ h_1 - h_4 \\ h_1 - H_5 \\ h_1 - h_4 \end{bmatrix} - \mathbf{L}^{-1} \begin{bmatrix} K_1' & 0 & 0 & 0 \\ 0 & K_2'+K_4' & 0 & K_4' \\ 0 & 0 & K_3'+K_5' & K_3' \\ 0 & K_4' & K_3' & K_3'+K_4'+K_6' \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \\ q_5 \\ q_6 \end{bmatrix} \end{aligned}\right\} \quad \cdots\cdots(22)$$

この状態方程式を時間積分すれば，未知の水頭 h_1，h_4 と流量の独立変数 q_1，q_2，q_5 および q_6 が求まる。さらに，これらの流量を（14）式に代入すれば，従属変数 q_3，q_4，q_7 および q_{11} が求まる。よって，パイプラインシステムの全ての状態を求めることができる。このように，閉路解析ではシステムグラフ内の辺を全域木と補木とに分離することができれば，流量とエネルギーの連続条件を表したシステム方程式から状態方程式を導出することができる。ただし，状態方程式を導出する過程は，代入や移項などの数式処理が行われており，接続関係が複雑なパイプラインシステムの例に適用する場合には解析者の負担が大きくなる。そこで，これらの数式処理をプログラムによって自動化すると，本手法を容易に適用することができるようになるが，これについては同じ技報（田中ら，2012）に詳細に記す。

7 全域木と補木の辺の決定支援の必要性

本稿において，これまで状態方程式が容易に導出できた理由は，全域木と補木の辺を分離することが流量の独立変数を選定することであり，あらかじめ全域木と補木の辺が指定されていたためである。

全域木と補木の辺を指定する際の条件をまとめると，以下のとおりである。

①3種類の境界（未知の水頭をもつ自由境界，既知の流量境界，既知の水頭境界）と基準面Dは辺で連結する。

②未知の水頭をもつ水槽から基準面Dへの辺は，全域木の辺に含める。

③既知の流量境界から基準面Dへの辺は，補木の辺に含める。

④既知の水頭境界から基準面Dへの辺は，全域木の辺に含める。

⑤補木の辺の数は，辺の数を n 本，節点の数を m 個とすると，$(n - m + 1)$ 本である。

⑥補木の辺の数と基本閉路の数は等しい。

この条件を満たして全域木と補木の辺に指定できる組み合わせは複数ある。例えば，1節で指定した異なる辺の番号 {1，3，4，6，10} や {1，2，3，6，10} などの組み合わせが補木の辺として指定することができる。よって，補木の辺の指定は，ある程度自由度のある探索であると言える。しかし，大規模なパイプラインシステムにおいては，技術者が条件②～④はチェックできても，$(n - m + 1)$ 本の補木の辺を選び，基本閉路を探索する作業は煩雑である。そのため，条件①～④の段階まで入力したデータから，条件⑤と⑥を素早く解決できるように全域木と補木の辺の指定を支援する探索アルゴリズムが必要である。

Ⅳ 全域木と補木の辺の決定支援方法の提案

Ⅲ章7節においてまとめた全域木と補木の辺を指定する際の条件のうち①～④は機械的な作業であるので，プログラムにてデータの入力支援をするように実装することもできるが，煩雑な処理にプログラムによる支援が必要とされている条件⑤と⑥に焦点を当てたいので，条件①～④は技術者によって入力されたデータがあることを前提条件とする。この章では，条件⑤と⑥を満たして全域木と補木の辺を指定することを支援するプログラムについて提案する。

1 方針

条件⑤，⑥とは，補木の辺を1本だけ含む閉路を $(n - m + 1)$ 本探索することである。この条件を満たす補木の辺の組み合わせは複数あるが，大規模なパイプラインシステムをモデル化したシステムグラフから補木の辺を1本だけ含む閉路を指定することは煩雑な作業である。そこで，補木の辺であるかの判断は条件②～④を満たすように行ったデータ入力作業までを行い，残りの補木の判断は探索プログラムによって解決したい。

辺の数が n 本，節点の数が m 個のシステムグラフには，$(n - m + 1)$ 本の補木の辺がある。よって，システムグラフは補木の辺と全域木の辺で構成されているので，全域木の辺の数は $(m - 1)$ 本である。この時点で，システムグラフ内のどの辺が補木または全域木の辺であるかについては指定していないとする。

このシステムグラフに，未知の水頭を持つ自由境界 m_1 個，既知の流量境界 m_2 個，既知の水頭境界 m_3 個を設定した場合に条件②～④を満たすと，$(n - m + 1 - m_2)$ 本の補木の辺と $(m - 1 - m_1 - m_3)$ 本の全域木の辺が指定されていない状態である。指定されているのは，補木の辺 m_2 本と全域木の辺 $(m_1 + m_3)$ 本のみである。この指定を初期の入力データとする。

補木の決定支援プログラムは以下の2つのアルゴリズムを繰り返して，初期の入力データを満たすようにする。

①全域木探索：システムグラフの辺を全域木と補木に分解する。

②閉路解決：全域木探索によって得られた補木の辺を1本だけ含む閉路を探索する。

2 全域木探索

全域木は，すべての節点がつながっており，かつ，閉路を1つも持たない辺の組み合わせである。よって，全域木探索とは，ある節点から到達可能なすべての節点を探索することである（例えば，猪股・益崎，1994）。全域木探索のアルゴリズムとしては，深さ優先探索（depth-first search，以下，DFSと呼ぶ）と幅優先探索（breadth-first search，以下，BFSと呼ぶ）がある。どちらも，すべての接点を探索して，形は異なるが全域木を得ることができるので，DFSのみについてアルゴリズムを説明することにする。

着目している節点につながった複数の辺について，それらの辺の反対側にあるすべての節点を「子」と呼び，ある節点についてすべての子を得る操作を「展開」と呼ぶことにすると，DFSは，ある節点を展開して得た子から1個選択し，それをさらに展開することを繰り返す探索方法である。**Fig.5** に示したシステムグラフを例に探索順序を説明する。

はじめに，節点0を探索開始の節点とする。ある節点を展開した子とつながる辺の番号が若い方を選択する方針で探索を続けると，ひとまず節点 {0，1，2，4，6} の順に探索できる（**Fig.6(a)**）。**Fig.6** 中の実線が補木の辺であり，点線が全域木の辺である。次に節点6を展開した場合，子の一つは節点0であるが，節点0を選択すると閉路を持つことになるので，節点5を選択しなければならない。よって，探索した節点の順番は {0，1，2，4，6，5，3} となり，探索の際に通過した辺が全域木と

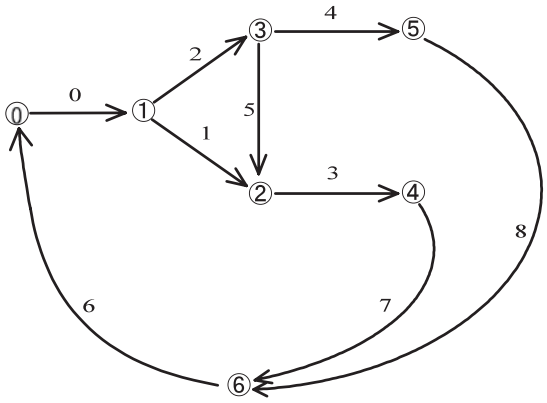

Fig.5　システムグラフの例
Example of a system graph

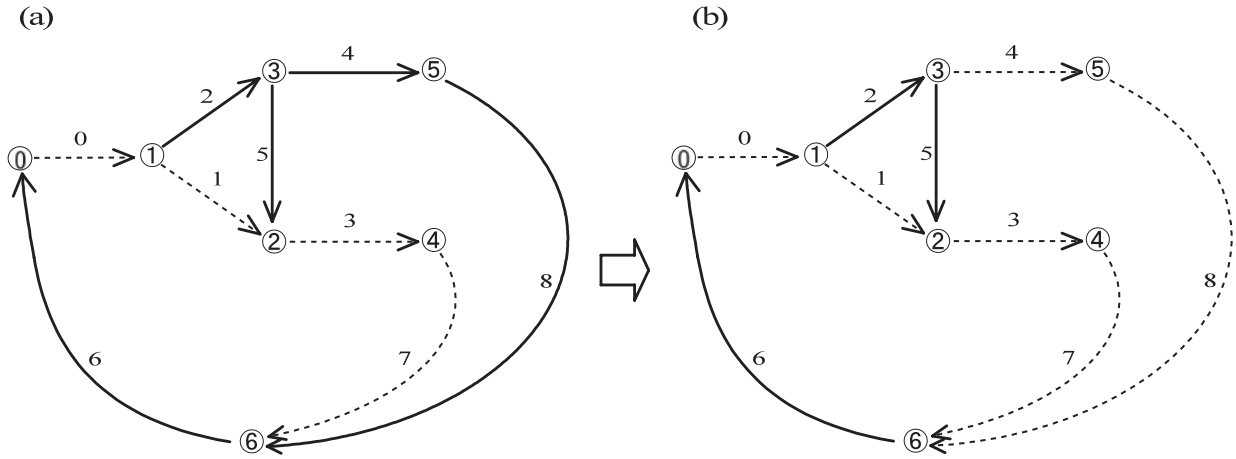

Fig.6　DFS による節点の選択順序
Selection order of the node by DFS

なる。したがって，全域木は辺 {0，1，3，4，7，8} である（**Fig.6(b)**）。システムグラフのすべての辺から全域木の辺を差し引けば，補木が辺 {2，5，6} と得られる。

全域木探索では補木の辺を指定することが可能である。それは，全域木探索の際に指定の辺は通過しないという制限を与えることによって可能になる。例えば，辺 1 を通過しないと指定した場合，探索した節点の順番は {0，1，3，2，4，6，5} になる。得られた全域木は辺 {0，2，3，5，7，8}，補木は辺 {1，4，6} である。ただし，子が 1 つしかない場合は，その間の辺は通過しないという制限を指定すると節点が孤立してしまうために，補木の辺として指定できない。他方，全域木探索では全域木の辺を指定しても，必ずしも満足するとは限らない。なぜなら，それは指定の辺を必ず通過しなければならないという制限を与えることであるが，全域木の辺として指定された辺が探索した最後の節点の子に属していた場合は，指定を満たすことが出来ないためである。そこで，全域木の辺の指定を完全に満たすために，本章 3 節の閉路解決と連携してアルゴリズムを構築した。その方法は本章 4 節で説明する。

3　閉路解決

全域木探索によって補木の辺が得られた時点では，基本閉路を構成する辺の経路は見つけられていなかった。この閉路情報を得るために，補木の辺が 1 本だけ含まれる閉路を探索することが本節で用いる Ternary network flow 法である。この方法は network flow 理論と balanced ternary number に基づく手法である。ここで，network flow 理論とは，システムグラフにおいて flow という数値をある節点に割り振った場合に，各辺にどのように伝播するか，また，伝播がどのような保存則に従うか，などを規定した理論である。また，balanced ternary number（以下，b.t.n と呼ぶ）とは三進数の一種で，各桁において − 1，0，1 のいずれかの値をとる。例えば，b.t.n と十進数との対応を 0 から 10 まで **Table 1** に示す。ここで，− 1 は $\underline{1}$ と表記した。

以下，Ternary network flow 法の手順を箇条書きにて記す（Doris and Stephan, 1981）。

手順①　すべての辺の flow に 0 を割り当てて初期化する。

手順②　補木の中の辺には順番を付けておく。i 番目（i = 1，2，3・・・）の補木の辺の始点に -3^{i-1}

Table 1　b.t.n と十進数の関係
Relationship between decimal number and b.t.n

十進数	変換式	b.t.n
0	0×3^0	0
1	1×3^0	1
2	$1 \times 3^1 - 1 \times 3^0$	$1\,\underline{1}$
3	$1 \times 3^1 + 0 \times 3^0$	1 0
4	$1 \times 3^1 + 1 \times 3^0$	1 1
5	$1 \times 3^2 - 1 \times 3^1 - 1 \times 3^0$	$1\,\underline{1}\,\underline{1}$
6	$1 \times 3^2 - 1 \times 3^1 + 0 \times 3^0$	$1\,\underline{1}\,0$
7	$1 \times 3^2 - 1 \times 3^1 + 1 \times 3^0$	$1\,\underline{1}\,1$
8	$1 \times 3^2 + 0 \times 3^1 - 1 \times 3^0$	$1\,0\,\underline{1}$
9	$1 \times 3^2 + 0 \times 3^1 + 0 \times 3^0$	1 0 0
10	$1 \times 3^2 + 0 \times 3^1 + 1 \times 3^0$	1 0 1

を割り当て，終点に 3^{i-1} を割り当てる。負号は節点からの出ていく量であることを表している。補木の辺すべてについて同様に割り当てる。

手順③　すべての節点における flow の合計が 0 になるように，全域木の辺にも flow を割り当てて調整する。

手順④　全域木の辺に割り当てられた flow の値を b.t.n に変換する。この i 桁目の値 c_i は全域木の辺と補木の辺との関係を示している。つまり，flow が割り当てられた全域木の辺は，i 番目の補木の辺との間に以下の関係がある。ただし，3^0 の位を 1 桁目，3^1 の位を 2 桁目，3^2 の位を 3 桁目として右から桁数を数えることにする。

$$c_i \begin{cases} 1 \cdots i \text{ 番目の補木の辺が含まれた基本閉路に含まれ，かつ，同じ向きである。} \\ -1 \cdots i \text{ 番目の補木の辺が含まれた基本閉路に含まれ，かつ，逆向きである。} \\ 0 \cdots i \text{ 番目の補木の辺が含まれた基本閉路に含まれていない。} \end{cases}$$

例えば，本章 2 節 **Fig.5** に対して全域木探索で得られた結果，全域木の辺 ｛0，1，3，4，7，8｝と補木の辺 ｛2，5，6｝について，Ternary network flow 法を適用した。手順②，③を **Fig.7** に図示した。図中の四角で囲んだ数字が flow である。手順④によって得られた結果を **Table 2** に整理した。**Table 2** を見ると，Ⅲ章 7 節の条件②～④を満たす 3 個の基本閉路が以下の通り見つかったことが分かる。ただし，負号は全域木の辺の向きが補木の辺の向きと逆方向であることを表している。

1 番目の補木 2・・・
　基本閉路 ｛－（全域木 1），－（全域木 3），全域木 4，－（全域木 7），全域木 8｝
2 番目の補木 5・・・
　基本閉路 ｛全域木 3，－（全域木 4），全域木 7，－（全域木 8）｝
3 番目の補木 6・・・
　基本閉路 ｛全域木 0，全域木 1，全域木 3，全域木 7｝

4　補木の決定支援アルゴリズム

Ternary network flow 法による閉路の探索は，システムグラフを全域木と補木に分解することが出来れば適用することができる。全域木探索では，補木の辺を指定することが可能であったが，全域木の辺の指定を必ずしも満足することは出来なかった。これを解決する方針として，はじめはすべての辺を「全域木の辺」に指定しておき，Ⅲ章 7 節の条件②～④が指定された辺の入力データに整合する閉路が得られるまで「補木の辺」を増やしてくこ

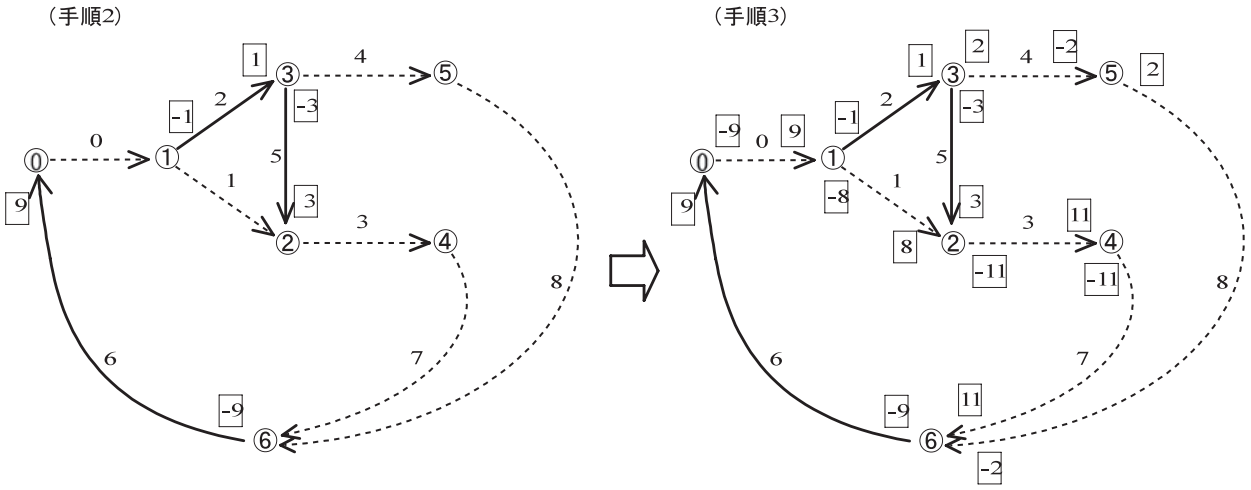

Fig.7　Ternary network flow 法における手順 2 と 3
Procedures 2 and 3 in the Ternary network flow method

Table 2　Ternary network flow 法における手順 4 の結果
Result of procedures 4 in the Ternary network flow method

全域木の辺	flow	変換式	b.t.n	補木 6	補木 5	補木 2
0	9	$1 \times 3^2 + 0 \times 3^1 + 0 \times 3^0$	1 0 0	1	0	0
1	8	$1 \times 3^2 + 0 \times 3^1 - 1 \times 3^0$	1 0 $\underline{1}$	1	0	− 1
3	11	$1 \times 3^2 + 1 \times 3^1 - 1 \times 3^0$	1 1 $\underline{1}$	1	1	− 1
4	− 2	$- 1 \times 3^1 + 1 \times 3^0$	$\underline{1}$ 1	0	− 1	1
7	11	$1 \times 3^2 + 1 \times 3^1 - 1 \times 3^0$	1 1 $\underline{1}$	1	1	− 1
8	− 2	$- 1 \times 3^1 + 1 \times 3^0$	$\underline{1}$ 1	0	− 1	1

とによって，全域木の辺の指定を満足するようにする。そのアルゴリズムのフローチャートを **Fig.8** に示し，以下，ステップごとに説明する。

ステップ①：入力ファイルからデータを読み込む。その際に，辺の指定が「全域木の辺」,「補木の辺」，または「どちらも指定しない」分類される。Ⅲ章 7 節の条件②〜④が指定された辺の入力データは，境界条件と基準面との間の辺のみに全域木か補木かの指定がされ，その他の辺はどちらも指定されていない。

ステップ②：システムグラフに含まれる辺について 1 本ずつ補木となりうるか検討を行う。その際，辺の番号が若い順から補木の辺の候補とする。

ステップ③：システムグラフ内のすべての節点と辺の属性を初期化する。節点の属性は，全域木探索において探索済みかどうかのフラグを未完にし，閉路解決において使用する flow を 0 にする。辺の属性は，節点同様に flow を 0 とし，入力データにおいて指定されていない辺は「全域木の辺」と指定して初期化する。

ステップ④：全域木探索を行う。辺の指定が「補木の辺」であった場合，指定の辺は通過しないという制限を与えて全域木探索を行い，全域木を得る。

ステップ⑤：システムグラフから全域木を差し引くことによって，補木を得る。

ステップ⑥：得られた補木の辺を用いて，補木の辺を一つだけ含む閉路を構成する辺の経路を探索する。

ステップ⑦：入力データに基づく辺の指定と本アルゴリズムによって得られた辺の状態が，一致するかどうか確認する。一致した場合は，適切な全域木と補木が得られる。

ステップ⑧：妥当な閉路が得られなかった場合，検討した辺 $i - 1$ は補木の辺として適切でなかった可能性があるため，「どちらでもない」という指定に変更する。他方，補木の辺として良い場合はそのままにしておく。

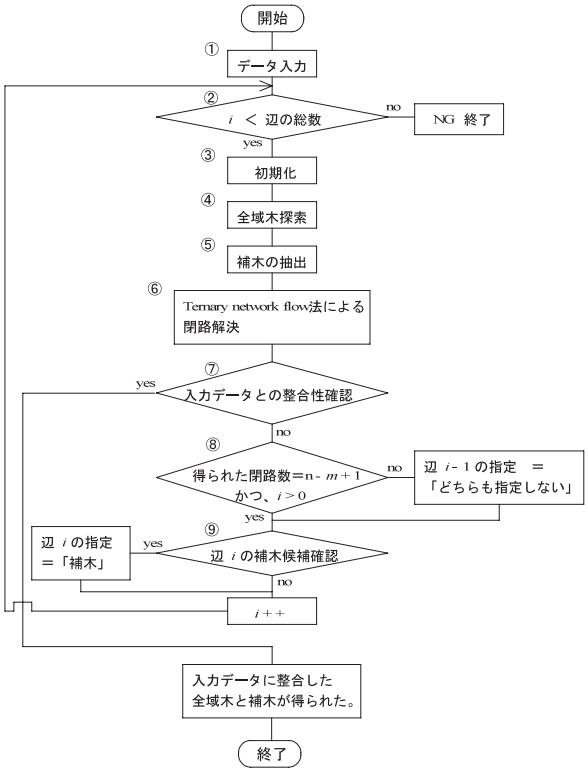

Fig.8　全域木と補木の決定支援アルゴリズムのフローチャート
Flowchart of the algorithm which supports that apanning tree and cotree are designated

ステップ⑨：次に検討する辺 i を選び，それが補木に含める辺としての以下の条件を満たしていれば，「補木の辺」と指定する。

条件①：始点と終点が「補木の辺」でない辺に自身を除き最低 1 本つながっている。

条件②：入力データにおいて「全域木の辺」として指定されていない。

ステップ①から⑨までを入力データの辺の指定と整合するまで繰り返す。

5　解析例

前節で提案したアルゴリズムの動作を確認する。確認することは，全域木の辺について一部を入力データとして指定した場合に，この指定を満たした全域木と補木の辺の分離ができて，最終的に閉路情報を得ることができるかを試みることである。**Fig.5** のシステムグラフに **Fig.9** に記したパイプラインシステムの物理的な条件を適用する。Ⅲ章 7 節の条件①～④を満たすように入力データを作成すると，節点 0，4 および 5 は既知の水頭境界を表す点なので，これらと基準面 D とを結ぶ辺 ｛6，7，8｝ は全域木の辺として指定することにする。

本章 4 節のアルゴリズムに従って，入力データにおける辺の指定と一致する閉路情報を得るまでの過程について，探索の手順を以下に説明する。

まず，補木の候補として番号が最も若い辺 0 を補木の辺とする。よって，辺 0 以外の辺はすべて全域木の辺とする。節点 0 を起点として深さ優先探索に基づく全域木探索を行うと，辺 0 は補木の辺なので通過しない。展開した子につながる辺の番号が若い方を選択する方針で探索を続けると，全域木の辺は ｛6，7，3，1，2，4｝，補木の辺は ｛0，5，8｝ となる。入力データでは，辺 8 を全域木の辺として指定していたが，探索の結果では補木の辺となってしまったので，この場合は入力データを満足する結果ではない。

次に，辺 0 は補木の辺のままに，辺 1 も補木の辺に追加する。節点 0 を起点として深さ優先探索を行うと，全域木の辺は ｛6，7，3，5，2，4｝，補木の辺は ｛0，1，8｝ となる。やはり，辺 8 は補木の辺となる結果が得られ，またも入力データを満たすことができなかった。

次に，辺 0 と 1 は補木の辺のままに，新たに辺 2 を補木の辺に追加したいのだが，辺 2 は本章 4 節のステップ⑨における条件①を満たさないので，補木の辺にできない。よって，次の順番である辺 3 を補木の辺に追加して，節点 0 から深さ優先探索を行うと，全域木の辺は ｛6，7，8，4，2，5｝，補木の辺は ｛0，1，3｝ となる。この探索結果は，入力データで指定した全域木の辺 ｛6，7，8｝ を満足している。その時の基本閉路行列 **B** は，Ternary network flow 法による閉路探索によって次式の通り求められる。

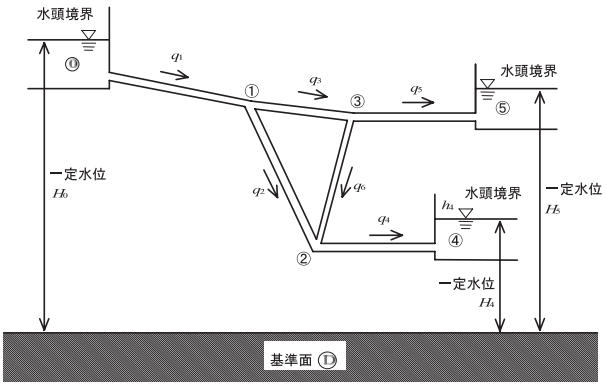

Fig.9　パイプラインシステム例 2
Example of pipeline system 2

$$\mathbf{B} = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ -1 & 0 & -1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & -1 & 1 & 0 & 1 & -1 & 0 & 0 & 1 \end{bmatrix} \quad \cdots\cdots\cdots(23)$$

なお，基本カットセット行列 **A** は，（9）式のように基本閉路行列 **B** から変形して容易に求めることができる。

Ⅴ　結　言

本稿を整理すると，以下のように要約できる。

①閉路解析における状態方程式の導出過程を具体例に基づいて行い，閉路情報からシステムグラフを全域木と補木とに分離できれば，流量の独立変数と従属変数が判明し，式を変形することにより状態方程式が得られることを説明した。

②パイプラインをモデル化したシステムグラフについて全域木と補木の辺を指定する際の条件を整理し，技術者が機械的に行える作業と繁雑で面倒な作業に分離した。後者はシステムグラフを全域木と補木とに分離する作業であり，これをプログラムで支援するように試みた。

③全域木探索法は，補木を指定して分離することは可能であるが，欠点として，全域木を指定しても完全に分離できない問題点があった。そこで，全域木を指定して分離できるアルゴリズムを構築することによって，システムグラフ内の境界条件における辺の指定を入力データとして与えるだけで，これを満たすように，システムグラフ内の補木と全域木の辺を分離して，基本閉路を見つける方法を提案した。

参考文献

1）Doris R. Ryan and Stephen Chen（1981）：A Comparison of Three Algorithms for Finding Fundamental Cycles in a Directed Graph，Networks，**11**，1-12

2）浜口憲一郎・鬼塚宏太郎（1980）：管水路における非定常現象の弾性水柱理論界と剛性水柱理論界の比較について，農土学会大会講演要旨集，22-23

3）猪股俊光・益崎真治（1994）：Scheme による記号処理入門，森北出版，100-110

4）内藤克美・小山潤・岩崎和己（1983）：管路系の”ゆるやかな過渡水理現象”解析のための汎用プログラムの開発，農土誌，51，3，215-227

5）鬼塚宏太郎（1977）：状態空間解析による枝分かれ管路のサージング減衰特性の評価，土木学会論文報告集，262，79-88

6）鬼塚宏太郎（1981）：農業用パイプラインの非定常解析に関する研究，東京農工大学農学部学術報告，

第 22 号
7）鬼塚宏太郎（1982）：パイプラインの水理設計（その 7）－剛性モデルによる過渡現象解析－，農土誌，50，3，259-268
8）鬼塚宏太郎（1998）：送配水システム解析入門，技報堂出版，61-85
9）島田正志（1988）：接続行列による剛性モデル・パイプライン非定常流解析理論，農土論集，134，77-83
10）島田正志（1991）：パイプライン非定常流の剛性モデル・接続行列法における状態変数の決定法，農土論集，156，17-22
11）田中良和・中田達・樽屋啓之（2012）：パイプライン非定常流の剛性モデル・閉路解析におけるオブジェクト指向プログラミングによる数式処理の自動化，農村工学研究所技報，212，13-28

Study of Supporting Method that Spanning Tree and Cotree are Designated Using Rigid Water Column Model of Analyzing Slow Transients in Pipelines

TANAKA Yoshikazu, NAKADA Toru and TARUYA Hiroyuki

Summary

The rational water management which coincides with water-use plan in proportion to the change of recent water demand has been required. Numerical method for analysis is important in order to examine the plan which carries out improvement of the elastic water management method and improvement of facilities. Rigid water column model - closed circuit analysis for slow transients in pipelines are one of effective numerical hydraulic analysis technique. In rigid water column model - closed circuit analysis, the state equation is obtained shortly, because independent variable and dependent variable of flow rate are automatically obtained, if to separate system graph which modeled pipeline to the edges of spanning tree and cotree is possible. But the work which separates a system graph to edges of spanning tree and cotree is not easy for engineers. Then, computer program which supported that work has been developed. The method is an algorithm using spanning tree search and Ternary network flow method. By the benefit of this program, it is expected that automatically deducing the state equation in rigid water column model - closed circuit analysis is possible.

Keywords : pipeline system, graph theory, state equation, unsteady flow analysis, Ternary network flow method