有限要素法による管網計算

九州産業大学　正　員　加納正道
福　岡　大　学　正　員　黒木健実
九州産業大学　正　員　赤坂順三
九州産業大学　　　　　木下義博

1. まえがき　管網計算を行うプログラムの汎用化および入力データの簡素化のために、有限要素法を使った管網計算を試みました。管網計算に有限要素法を用いるためには、各管内のまさつ損失水頭および流量を各節点における値におきかえる必要があります。また、流量とまさつ損失水頭が非線形な関係になっている流量式を線形化することも重要となります。

2. 理論と計算手法

2.1 基礎方程式　管網計算の基礎式として以下に示す4方程式が用いられます。

流量式：　$h=rQ^m$　(1.1)　ここに、r は抵抗係数、m は指数を示します。

節点方程式：　$\sum_{j=1}^{s} Q_{ij}=Q_{i\ out}$　(1.2)　ここに、Q_{ij} は任意の節点 i に流入する j 番目の管の流量、s は節点 i に流入する管の数、$Q_{i\ out}$ は節点 i における流出量を示します。

閉管路方程式：　$\sum h = 0$　(1.3)　即ち、まさつ損失水頭の和はゼロになります。

流出入平衡式：　$\sum_{i=1}^{p} Q_{i\ out} = 0$　(1.4)　即ち、与えられた管網全体に流入する量と流出する量は平衡しなければなりません。ここに、p は節点の数を示します。

2.2 有限要素法による定式化

2.2.1 節点値へのおきかえ　この問題を有限要素法で解くためには図1に示すように、管網を4個の節点と4個の管とに分解して考えます。

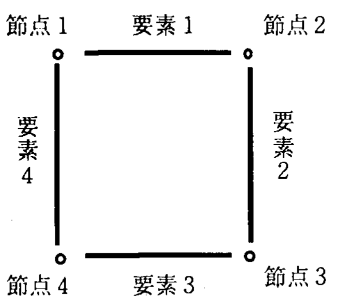

図1　節点および要素番号づけ

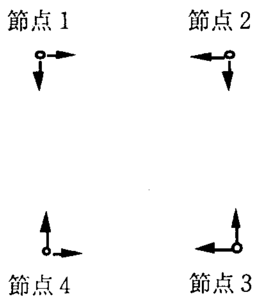

図2　流出入方向

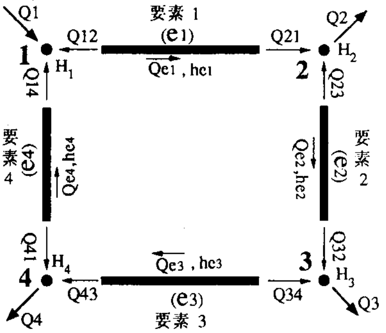

図3　一閉管路の流量収支

1本の管を一つの要素とみなし、要素と節点には別々に通し番号を打ちます。節点番号と要素番号は通し番号をつけるということ以外には番号の付け方に制約はありません。次に節点への流入量や節点からの流出量について考えます。いま水は図2に示すような方向に流れていると仮定してみましょう。図3に示すように第1の要素の両端の節点 1、2 における水頭をH_1, H_2、流出量をQ_{12}, Q_{21}とおき、要素についてのまさつ損失水頭h_{e1}と流量Q_{e1}を次式のように節点の水頭H_1, H_2と流量Q_{12}, Q_{21}でおきかえてみます。

$$h_{e1} = H_1 - H_2 \quad , \quad Q_{e1} = -Q_{12} = Q_{21} \tag{1.5}$$

2.2.2 線形化　次に流量式を次のように線形化します。

$$h_{e1} = H_1 - H_2 = rQ_{e1}\hat{Q}_{e1}^{(m-1)\gamma}, \hat{\hat{Q}}_{e1}^{(m-1)(1-\gamma)} \tag{1.6}$$

ここに右辺の2番目と3番目の流量はそれぞれ1要素における繰り返し計算の1回前と2回前の既知流量を意味し、γ=0.8 はそれらを取り込む割合を示しています。すると水頭(H_1, H_2)を使っ

て管の両端の流量(Q_{12}, Q_{21})を次式のように表示できます。

$$Q_{12}=\frac{1}{\left[r\hat{Q}_{e1}^{(m-1)\gamma}\ \hat{\hat{Q}}_{e1}^{(m-1)(1-\gamma)}\right]}(H_2-H_1) \quad , \quad Q_{21}=\frac{1}{\left[r\hat{Q}_{e1}^{(m-1)\gamma}\ \hat{\hat{Q}}_{e1}^{(m-1)(1-\gamma)}\right]}(H_1-H_2) \tag{1.7}$$

ここで図３に示す閉管路について流量計算する場合を例にあげて有限要素法プログラムによる管網計算の手順を説明しましょう。

1)全ての管の第一次仮定流量を１.０とします。

2)節点ごとに流出入量の釣り合い式(1.4)を考えます。即ち

$$Q_{12}+Q_{14}=Q_1 \ :\ 節点\ 1 \quad , \quad Q_{21}+Q_{23}=Q_2 \ :\ 節点\ 2$$
$$Q_{32}+Q_{34}=Q_3 \ :\ 節点\ 3 \quad , \quad Q_{43}+Q_{41}=Q_4 \ :\ 節点\ 4 \tag{1.8}$$

次に流量式 (1.1) について、節点における水位(H)と流量(Q)の関係におきかえ、HとQの線形化を次式のように行います。

$$Q_{12}=\frac{1}{r_{e1}\left[\hat{Q}_{e1}^{(m-1)\gamma}\ \hat{\hat{Q}}_{e1}^{(m-1)(1-\gamma)}\right]}(H_2-H_1) \quad \sim \quad Q_{41}=\frac{1}{r_{e4}\left[\hat{Q}_{e4}^{(m-1)\gamma}\ \hat{\hat{Q}}_{e4}^{(m-1)(1-\gamma)}\right]}(H_1-H_4) \tag{1.9}$$

ここに$\hat{Q}_{e1}^{(m-1)\gamma}, \hat{\hat{Q}}_{e1}^{(m-1)(1-\gamma)}$は１要素における既知流量(仮定値)の項を表わします。

また、$1/\{r_{e1}\hat{Q}_{e1}^{(m-1)\gamma}\ \hat{\hat{Q}}_{e1}^{(m-1)(1-\gamma)}\}$は１要素の既知項の意味をもつのでこれを$k_{e1}$とおけば式(1.9)の１～４行目はそれぞれ次式(1.10)のように示されます。

$$-k_{e1}\cdot H_1+k_{e1}\cdot H_2=Q_{12} \ \sim\ k_{e4}\cdot H_1-k_{e4}\cdot H_4=Q_{14} \tag{1.10}$$

さらに式(1.10)を式(1.8)に代入して次式(1.11)が得られます。

$$-k_{e1}\cdot H_1+k_{e1}\cdot H_2+k_{e4}\cdot H_1-k_{e4}\cdot H_4=Q_1 \ ,\ k_{e1}\cdot H_1-k_{e2}\cdot H_2-k_{e1}\cdot H_2+k_{e2}\cdot H_3=Q_2$$
$$k_{e2}\cdot H_2-k_{e3}\cdot H_3-k_{e2}\cdot H_3+k_{e3}\cdot H_4=Q_3 \ ,\ k_{e4}\cdot H_1+k_{e3}\cdot H_3-k_{e4}\cdot H_4-k_{e3}\cdot H_4=Q_4 \tag{1.11}$$

これを$H_1\sim H_4$に関する連立一次方程式としてマトリックス表示すれば、

$$\begin{bmatrix} -k_{e1}-k_{e4} & k_{e1} & 0 & k_{e4} \\ k_{e1} & -k_{e2}-k_{e1} & k_{e2} & 0 \\ 0 & k_{e2} & -k_{e3}-k_{e2} & k_{e3} \\ k_{e4} & 0 & k_{e3} & -k_{e4}-k_{e3} \end{bmatrix}\begin{bmatrix} H_1 \\ H_2 \\ H_3 \\ H_4 \end{bmatrix}=\begin{bmatrix} Q_1 \\ Q_2 \\ Q_3 \\ Q_4 \end{bmatrix} \tag{1.12}$$

式(1.12)が得られ、これを解けば$H_1\sim H_4$が求まります。

以上、手計算により$H_1\sim H_4$に関する連立一次方程式の作り方を述べましたが、プログラム中では、有限要素法の考え方によってこの式を作成します。

なお、閉管路方程式($\Sigma h=0$)は式(1.5)の第１式により自動的に成り立っています。

3)式(1.12)を解いて各節点の水頭H_1～H_4が得られます。

4)各管の新流量$Q_{e1}\sim Q_{e4}$を次式(1.13)より求める。

$$Q_{e1}=\frac{1}{r\left[\hat{Q}_{e1}^{(m-1)\gamma}\ \hat{\hat{Q}}_{e1}^{(m-1)(1-\gamma)}\right]}(H_1-H_4) \sim Q_{e4}=\frac{1}{r\left[\hat{Q}_{e4}^{(m-1)\gamma}\ \hat{\hat{Q}}_{e4}^{(m-1)(1-\gamma)}\right]}(H_4-H_1) \tag{1.13}$$

5) Q_{e1}～Q_{e4} を $\hat{Q}_{e1}\sim\hat{Q}_{e4}$と入れ換えて、手順 2) から 5) までを流量の相対誤差の和が許容誤差以下となる(許容誤差をεとして $\Sigma(Q_{ei}-\hat{Q}_{ei})/\hat{Q}_e<\varepsilon$, $i=1,2...4$ を満足する)まで繰り返します。

3. まとめ 管網計算へ有限要素法を適用することにより、管網数が１個から数十個と増えても入力データを変えるだけで同じプログラムで解析できる汎用性が得られました。また通常のハーディ・クロス法などでは第１次仮定流量を定める場合に多大の労力を必要としますが、本法ではプログラムの中で自動的に１.０とすることで対応できます。