

1. まえがき

多元連立一次方程式を帯行列法を用いて効果的に解こうとする場合、対象行列の帯幅を小さくする必要がある。今日まで例えばCutkhill-McKee Algorithm (C-M法)¹⁾、Gibbs-Poole-Stodolmayr Algorithm (G-P-S法)²⁾等多くの方法が提案されているが、一般に、帯幅減少法に要求される条件として、1) より少ない演算時間²⁾、2) より小さい帯幅が得られること、が挙げられる。一方、構造解析の分野においては、今日、有限要素法(FEM)、差分法(FDM)が解析法の主流を占め、解の精度の理由により、あるいは系の巨大化・複雑化により、解がねばならない連立一次方程式もまた、非常に多くの変数を有するものとなりつつあり、従って上記諸法がしばしば利用されるようになった。しかしながら、これらの諸法は、汎用性のあるものであるが故に、逆にFEM、FDMといったメッシュ系固有の特性、例えば a) 節点は内部節点と境界上の節点に分類でき、b) メッシュパターンが比較的簡単なものが多い、等を有効に利用できない。上述した帯幅減少法に要求される2条件を更に十分に満たすには、これらの特性を利用する必要がありと考える。一方、白石・谷口は³⁾今日まで、a)の特性を導入した帯幅減少法を初めて提案している。本研究では、彼らの方法に、更にb)の特性の導入を図り、より効果的な帯幅減少法の提案を行う。

2. 帯幅減少法

分散化モデルにおける荷重ベクトル P と、変位ベクトル U との間に下式が成り立つ。

$$P = K \cdot U \quad (1)$$

ただし、 K は剛性行列であり、通常対称性($k_{ij} = k_{ji}$)を有し、更に、多くの要素が零である場合が多く、これを ∞ と規定する。行列 K の半帯幅 HBWは、

$$HBW = \max_i (j-i), \text{ ただし } j > i \text{ かつ } k_{ij} \neq 0. \quad (2)$$

行列 K ($n \times n$)に対し、1, 2, ..., m と番号付けた m 個の点を準備し、 K の上三角行列内の全2の k_{ij} ($\neq 0$)に対し、 i 点と j 点(=それぞれ U_i, U_j を示す)を線で結ぶ(この関係を“隣接”と呼ぶ)という操作を行うことにより、 K を1つのグラフ: $G(m)$ で表現できる。 U_i, U_j が隣接関係にあることを $\text{adj. } U_i \rightarrow U_j$ で示せば、(2)式は、

$$HBW = \max |j-i|, \text{ ただし } U_j \in \text{adj. } U_i \quad (3)$$

最小半帯幅(= $\min HBW$)を求めたいには、(2)式では行列の入れ換え、(3)式では、点番号の付け換えによる必要がある。これを新たな言葉の定義を行う。 $G(X)$ のレベル構造⁴⁾

$$L = \{L_0, L_1, L_2, \dots, L_l\} \quad (4)$$

とは、 G の点 X を次の2式を満たすように部分集合 L に分割することである。

$$\text{adj.}(L_i) \subseteq L_{i-1} \cup L_{i+1} \quad (5)$$

$$\text{adj.}(L_0) \subseteq L_1, \quad \text{adj.}(L_l) \subseteq L_{l-1} \quad (6)$$

G をレベル構造 L で表わした時、 G の幅(w)は下式で定義される。

$$w = \max \{ |L_i| \mid L_i \in L \} \quad (7)$$

文献5) において提案されている図式帯幅減少法より、

$$HBW = w \quad (8)$$

すなわち、 $\min HBW$ はレベル構造の幅の最小値 $\min w$ に近似し、従って $\min HBW$ を求めたいには、 w の最小化を行えばよいことがわかる。

例えば、C-M法、G-P-S法では、グラフGの長軸軸を見出し、その端点よりレベル構造を作り、そのレベル構造の幅をより小さくするという手順により $\min W$ を達成しようとする。一方、白石・谷はの方法は、 $\min W$ の位置をまず見出し、そのWを構成する節点を採出し、その位置より、レベル構造を作り上げるという。上記2法とは逆の手順を行う(図-1参照)。従って、前者は $\min W$ を間接的にしか行いえないのに対し、後者は直接 $\min W$ を求める。このように、 $\min HBW$ を支配するWを直接見出すことも可能ならしめたのが、この構造系の境界上に位置した節点の活用である。一方、FEM等の数値解析で扱われる系のグラフにおいても、境界点と内部点を区別することは可能であることより、上記2法は、この数値解析モデルに対して利用可能である。

図-1に示されるアルゴリズムよりも明らかに、 $\min HBW$ を求める手順の短縮が可能。レベル構造の作成が出来る。従って、白石・谷の方法を改良しようとするには、このレベル構造作成法を改良するのが最善であり、その節点においては、まづがきで示したFEMメッシュ系固有の特性b)のレベル構造作成法の導入を図る。なお、文献3)のアルゴリズムにおいて、 L_i より L_{i+1} を作成する方法は、 L_i に含まれる全2の点の隣接性を保ち、 L_{i+1} を求めるのである。

3. 新たなレベル構造作成法の提案

ここでは、2次元連続体の有限要素解析がよく利用される三角形要素と四角形要素を対象として考察を加える。図-2は、この要素が整然と配置されている場合を示し、このような配列をレギュラーメッシュと呼ぶことにする。図-2よりも明らかに、グラフがレギュラーである。

例えば、 L_i より L_{i+1} を求める場合、

- ① 三角形では 1点のみ
- ② 四角形では 2点のみ

の L_i 内の点より採せばよい(図-2、aおよびb参照)。従って、対象がレギュラーメッシュであれば、レベル構造作成の手順は、白石・谷の方法と比較して、

- ① 三角形では $1/2$ 、
- ② 四角形では $1/3$

に減少すると云える。以後、この新たな手順のうち前者をESVS(Every Second Vertex Searching Method)、後者をETVS(Every Third Vertex Searching Method)と呼ぶことにする。

通常、FEMで扱われるメッシュ系は、図-2のようなレギュラーメッシュではなく、メッシュがイレギュラーに配置されている。従って、このようなイレギュラーメッシュ系に対しても有効なように、上記ESVS、ETVSの修正を以下で考へてみる。ただし、以下の文中 u は L_i に、また v は L_{i+1} に属する点を示し、 $adj. u_d$ と u_p なる点 u_p を採り出しうるように、修正を行う。

3-1. 三角形メッシュ系

三角形要素のみであり、 u_d 、 u_p の関係は、基本的には図-3に示す2つの場合しか存在しない。

Case 1 : E CVS により、 u_d または u_d の1つが u_p が E CVS

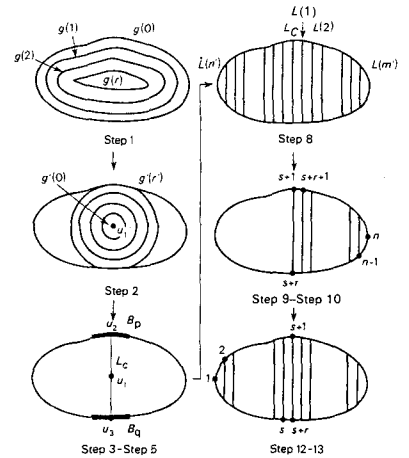


図-1. アルゴリズム

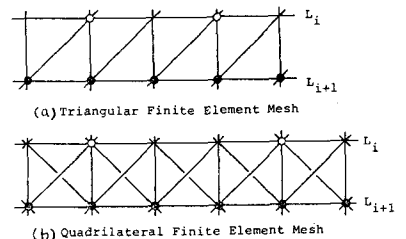


図-2. レギュラーメッシュ

用いられなければならない。従って、 U_p は必ず捨られる。

Case 2: もし、 U_d が $ESVS$ に用いられる場合は、 $U_p \in \text{adj. } U_d$ は必ず捨られる。

従って、問題となるのは、 U_d が SVS に用いられる場合であり、この場合を更に詳細に分類すれば、図-4に示す3ケースが考えられる。Case 2-1 では U_d が $ESVS$ に含まれる以上、 $W = U_{d-1}$ が $ESVS$ に利用され、従って U_p は捨られる。Case 2-2 では、 L_{i+1} との隣接関係を有する L_i の点を $ESVS$ より除くという条件を付加すれば、 U_p は捨られることになる。いま $\{U_i \mid U_i \in L_{i+1}, U_i \in \text{adj. } U_d\}$ であり、いまは捨られる点の数を $R\text{-deg. } U_d$ と示す。次のような修正を要することになる。

[修正 1: $R\text{-deg. } U_d = 0$ とする点 U_d は $ESVS$ より除く。]

Case 2-1 の場合、 U_p が U_d の隣接点より決まる点である。従って、 U_{d-1} は、 L_i に入る。また、 U_p (必ずしも) は捨られる可能性を持つことになり、 U_p を捨る際には次のような修正が必要となる。

[修正 2: $R\text{-deg. } U_d \geq 2$ の点 U_d は $ESVS$ に含める。]

よって、残された問題は、 $R\text{-deg. } U_d = 1 (= U_p)$ となる点 U_p は、修正した $ESVS$ で必ず捨られるのか? という点である。この点の捨られるか残るかは容易に見い出されることになり(図-5参照)、更に次の修正3が必要となる。ただし $\text{Deg. } U_d$ とは、 U_d の L_i, L_{i+1} への隣接点数を示すものである。

[修正 3: $\text{Deg. } U_d \geq 5$ 、かつ、 $R\text{-deg. } U_d = 1$ とする点 U_d は $ESVS$ に含める。]

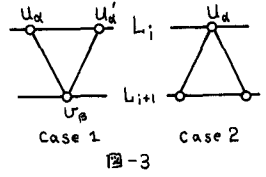


図-3

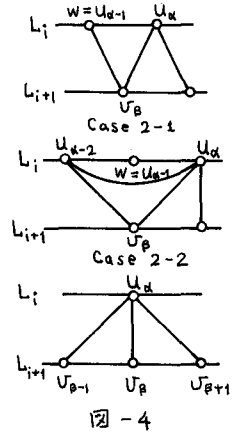


図-4

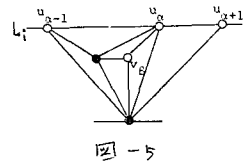


図-5

3-2. 四角形メッシュ系

上述の三角形メッシュの場合と同様に考える。 $U_d \in L_i, U_p \in L_{i+1}, \text{adj. } U_d \cup U_p$ の関係にある U_d, U_p は1つの四角形要素に含まれる2点であることになり、基本的にはそれらの位置関係は、図-6に示される3ケースしかあり得ないことになり得る。

Case 1: $ETVS$ により、 U_d, U_d', U_d'' のうち1つは必ず $ETVS$ に含まれることになり、 U_p は必ず捨られる。

Case 2: U_d, U_d' のいずれもが $ETVS$ に含まれる場合が1つ問題となる。この場合、図中の W, W' が $ETVS$ に含まれることになり、 U_p, U_p' が捨られる。このとき、 U_d が、例えば U_{p-2} に隣接していることになり得る(図-7参照)。 U_{p-2} は、 W より捨られる可能性を有することと考慮すれば、次の2つの修正を行うことになり得る。 (Case 2 の場合の U_p, U_p' は必ず捨られることになり得る。

[修正 1: $R\text{-deg. } U_d = 0$ の点 U_d は $ETVS$ より除く。]

[修正 2: $R\text{-deg. } U_d \geq 3$ の点 U_d は $ETVS$ に含める。]

上記2つの修正を行うと未だ考慮されていないのは $R\text{-deg. } U_d = 2$ の点 U_d である。この点の捨られるか残るかは容易に見い出されることになり、次の修正が必要となる。これにより、Case 3 の U_p は必ず捨られる。

[修正 3: $\text{Deg. } U_d \geq 7$ 、かつ $R\text{-deg. } U_d = 2$ or 1 の点 U_d は $ETVS$ に含める。]

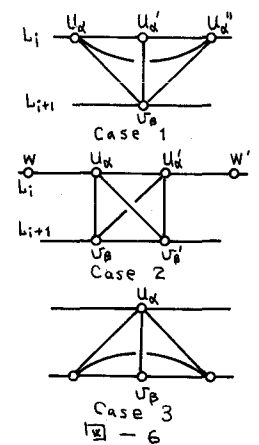


図-6

3-3. レベル構造を有するメッシュ系

$L_i = \{U_1, U_2, \dots, U_j, \dots, U_d\}$ であり $L_{i+1} = \{U_1, U_2, \dots, U_k, \dots, U_p\}$ と仮定する場合を考える。三角形メッシュでは $X=2, Y=5$ 、四角形メッシュでは $X=3, Y=7$ である。

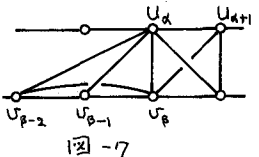


図-7

- Step 1: $j = 1$. Find out $\{v | v \in \text{adj. } U_j\}$.
- Step 2: $C = 0$.
- Step 3: $j = j + 1$. If $j = \alpha$, find out $\{v | v \in \text{adj. } U_j\}$ and go to Step 10. Otherwise, go to Step 4.
- Step 4: Calculate $\text{Deg. } U_j$ and $R\text{-deg. } U_j$.
- Step 5: If $R\text{-deg. } U_j = 0$, go to Step 3. Otherwise, go to Step 6.
- Step 6: $C = C + 1$. If $C = X$, find out $\{v | v \in \text{adj. } U_j\}$ and go to Step 2. Otherwise, go to Step 7.
- Step 7: If $R\text{-deg. } U_j \geq X$, find out $\{v | v \in \text{adj. } U_j\}$ and go to Step 2. Otherwise, go to Step 8.
- Step 8: If $\text{Deg. } U_j \geq Y$, go to Step 9. Otherwise, go to Step 3.
- Step 9: Find out $\{v | v \in \text{adj. } U_j\}$ and go to Step 2.
- Step 10: L_{i+1} is obtained.

4. 数値実験 および 考察

第2節に図示した白石-谷口の方法の中でのレベル構造作成法E上に示した新たなアルゴリズム4で置き換え、更に、数個所の改良を行ったアルゴリズム4E。三角形要素よりなる3例に適用し、表-1のよう結果を得た。また、図られた節点番号の1例を図-8に示す。なおこの表中のMethod 1, 2, 3は次のとおり。

Method 1: 平均次数以下の点よりC-M法によりレベル構造を作成し、その中で $\min w$ を求める。

Method 2: 最小次数以下の点よりC-M法によりレベル構造を作成し、その中で $\min w$ を求める。

Method 3: G-E-5法により出発点を見つけ出し、その点よりレベル構造を作成し w を求める。

この数値実験より、今回提案したアルゴリズム4は、Method 3と同程度の計算時間しか要せず、また半帯幅値に限りてはほとんども最小の値を満足していることより、現存するアルゴリズム4中の最良のものと考えられる。なお、今回のアルゴリズム4と、元のアルゴリズム4を比較すれば、計算時間に関しては約 $1/2$ となり、理論的考察の結果と一致している。従って、四角形メッシュ系について $1/3$ の計算時間に短縮されるものと予想される。

参考文献

- 1). E. Cuthill & J. McKee, Proc. ACM National Conference (1969), 157-172
- 2). N.E. Gibbs et al, SIAM J. Numer. Anal., 13 (1976), 236-250
- 3). 白石成人・谷口健男, 土木学会論文報告集, 314 (1981)
- 4). A. George, Lecture Notes in Mathematics 572 (1976), 53-101
- 5). I. Konishi et al, J. Struct. Mech., 4(2) (1976), 197-226

	New Algorithm	Method 1	Method 2	Method 3
Ex. 1 HBW	9	9	10	10
T(sec)	0.011	0.350	0.020	0.008
Ex. 2 HBW	12	13	18	16
T(sec)	0.024	3.709	0.030	0.025
Ex. 3 HBW	22	21	24	25
T(sec)	0.050	44.832	0.180	0.060

Table 1. Comparison of Results

[注意: Ex. 1, 2, 3の節点数はそれぞれ42, 99, 193]

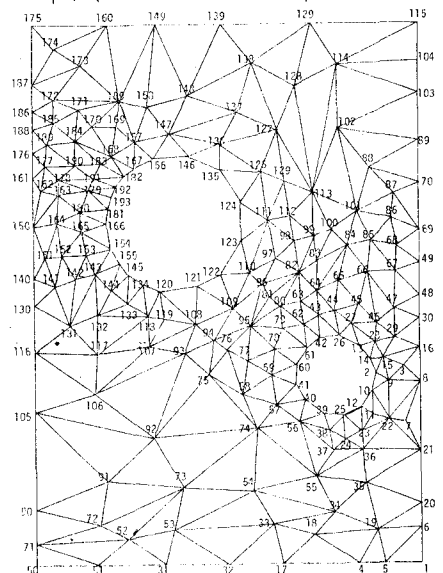


図-8. 例題3の結果