I-94　スーパーコンピュータに適した線形代数方程式の解法

岡山大学大学院　学生員　○藤原　浩二
岡山大学工学部　正会員　　谷口　健男

1 まえがき

大規模構造解析の需要により、線形代数方程式の解法の見直しが要求されている。ここで注目されているのが演算時間、容量、精度等に優れた反復解法の一種である修正共役傾斜法である。この解法の特徴としては、より良いプレコンディショナーを開発すればその収束性の向上が期待できる点にある。本研究ではジェニングスらの提案したプレコンディショナーの欠点である２つの項目（①不完全コレスキー分解の計算時にルート計算が計算不能になる。②計算機の容量及び演算時間を決定するパラメータの設定が必要であるが、実行以前に容量の予測ができない）の排除により利用しやすい修正共役傾斜法の提案を行う。

2 Preconditioner

AX=bという方程式を直接取り扱う代わりにある正則行列Ｂを両辺に左から掛けて修正した方程式ＢAX=Ｂbに共役傾斜法を適用するのが修正共役傾斜法であり、行列Ｂは積ＢＡの収束性を向上させることを目的としている。この行列ＢのことをPreconditionerと呼び、今日様々な行列Ｂが提案されているが、なかでも最も多く利用されているのが不完全コレスキー法の利用であって、具体的には行列Ｂを２つの三角行列で表し、行列Ａの近似逆行列となるような行列Ｂを探す方法である。ここではPreconditionerとしてこの不完全コレスキー分解を使用する。

3 数値実験及びその結果

3-1数値実験モデル

解析モデルとして60cm四方の正方形、板厚1cm、荷重1kg/cm^{2}の分布荷重、ヤング率2.0×10^6kg/cm^2、ポアソン比0.3の平板モデルを使用する。なお境界条件は、対辺固定、対辺単純支持、一辺固定とし１節点自由度はZ方向のたわみ、X,Y軸回りの曲げの3自由度である。また要素としては三角形板要素を使用する。

3-2 修正及び結果

修正 1）ジェニングスらが提案した不完全コレスキー分解を基本とした手法（以後Robust法と呼ぶ)は

For k = 2,3, ..,n

$$a_{kk} = \left(a_{kk} - \sum_{j=1}^{k-1} a_{kj}^2 \right)^{1/2} \quad \text{—— ①}$$

$$a_{ik} = \left(a_{ik} - \sum_{j=1}^{k-1} a_{ij}a_{kj} \right)/a_{kk} \quad \text{—— ②}$$

Where k < n and i = k+1,k+2, ..,n

このコレスキー分解をおこなう過程で対角項へ下記の修正を加える。ただしa_{ij}^*は分解過程の非対角要素

$$\text{IF} \quad a_{ij}^{*2} < \psi^2 a_{ii} a_{jj} \quad \text{—— ③} \quad \text{THEN}$$

$$a_{ii} = a_{ii} + \sqrt{a_{ij}^{*2} a_{ii}/a_{jj}} \quad \text{—— ④}$$

$$a_{jj} = a_{jj} + \sqrt{a_{ij}^{*2} a_{jj}/a_{ii}} \quad \text{—— ⑤}$$

つまりパラメータψによりコレスキー分解される非対角要素数が決定される。パラメータψと収束回数(IT)及び不完全コレスキー分解に用いた非零要素数の完全コレスキー分解での要素数に対する割合(IC)を表１に示す。表１に示すとおり$0\leqq\psi\leqq0.1$においての数値実験結果パラメータの値によっては分解不能（表1のX印）となった。この原因としては完全コレスキー分解では分解過程で生じるfill-inを全て考慮するのに対して不完全コレスキー分解ではある程度無視されること、またこの手法では、不完全コレスキー分解の過程でa_{jj}は常に修正されるのに対してa_{ii}はもしその行の最後の要素が排除されなければa_{ii}は修正されないという不公平さがあるという２点が考えられる。これにより式①の( )$^{1/2}$の中が負になり計算不能となる。そこでこの不公平さをなくすために式④を次のように修正する。

$$a_{ii} = a_{ii} + \mathrm{MAX}\left(a_{ij}^{*2} a_{ii}/a_{jj}\right)$$

この修正の結果、各行における $\sqrt{a_{ij}^{*2} a_{ii}/a_{jj}}$ の最大値で a_{ii} を修正することにより a_{ii} が過大評価され式①での a_{kj} が小さくなる。結果として表１（修正後）が示すとおり分解不能だったパラメータ ψ においても分解可能となった。

表 1 ψ とIT,ICの関係

	対辺固定（153元）				対辺単純支持（146元）				一辺固定（304元）			
PSI	修正前のIC	IT	修正後のIC	IT	修正前のIC	IT	修正後のIC	IT	修正前のIC	IT	修正後のIC	IT
0.00	1.000	1	1.000	1	1.000	1	1.000	1	1.000	1	1.000	1
0.01	0.540	20	0.536	21	0.719	29	0.711	33	0.390	87	0.383	91
0.02	0.504	25	0.500	26	0.660	41	0.640	45	0.308	103	0.306	106
0.03	0.487	30	0.482	31	0.602	55	0.609	60	0.273	122	0.264	117
0.04	×		0.451	38	×		0.602	69	0.251	125	0.247	130
0.05	0.438	35	0.433	43	0.540	73	0.553	78	0.233	146	0.231	146
0.06	0.398	49	0.392	50	0.522	82	0.521	87	0.219	155	0.216	167
0.07	0.377	52	0.368	57	0.494	92	0.491	95	0.186	183	0.181	203
0.08	0.368	55	0.357	60	0.463	104	0.462	111	0.160	237	0.158	240
0.09	0.347	60	0.348	63	0.440	109	0.452	112	0.133	264	0.134	262
0.10	0.332	63	0.330	66	0.426	113	0.419	121	×		0.118	281

修正 2） 次に上記手法でのパラメータ ψ はユーザー側で任意決定できるが、実際には計算を行わないとコレスキー分解される非対角要素数が判らない。そこで、パラメータ ψ のかわりに各行で、$\left|a_{ij}^{*}/a_{jj}\right|$ の最大値からM個と分解する非対角要素を決めて計算を行った。図１に示されるように、ここに提案したパラメータを用いればロバスト法よりわずかに反復回数は増加するものの計算着手以前に必要な容量の推定が可能となる利点を有することになる。この手法により(M=10 対辺単純支持)において分解不能が生じたが使用可能な計算機の容量によりパラメータを設定できるという点では有効であるといえる。

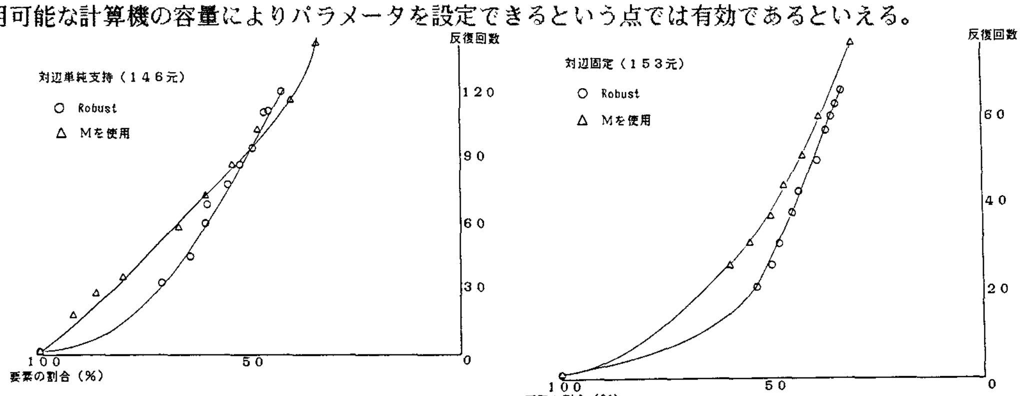

図 1 Robust法とM使用時との比較

4 あとがき

いままでに提案されている修正共役傾斜法の一つである ジェニングスらの Robust 法を修正することによって新しい修正共役傾斜法を提案した。修正１により、板曲げなどにおいて時として計算不能となっていたのを防ぎ、安定な計算を期待できることになった。つぎに修正２により、容量に関して不安のあったパラメータ ψ の排除が可能となり、計算前に容量に応じて決定できるパラメータMを導入することにより当初の目的は達成された。今後、このパラメータMを使用し分解不能を起こさないようにする検討が必要である。

参考文献）M.A.Ajiz,A.jennings､" A Robust Incomplete Choleski-Conjugate Algorithm " ; International Journal for Numerical Methods in Engineering (1983)