応用数理実験

連立 1 次方程式に対する CG 法の数値実験

桂田 祐史

1996 年 6 月 11 日

1 先週の講義の内容の復習

代表的な反復法として、**共役勾配法**(きょうやくこうばいほう)(**conjugate gradient method**, 略して **CG 法**) を取り上げる[1]。これは A が正定値対称行列である場合に利用可能な方法である[2]。

CG 法のアルゴリズム:

初期ベクトル $\mathbf{x_0}$ をとる ; 目標とする相対残差 ε を決める ;
$\mathbf{r_0} := \mathbf{b} - A\mathbf{x_0};\ \mathbf{p_0} := \mathbf{r_0};$
for $k := 0, 1, \cdots$ **until** $\|\mathbf{r_k}\| \le \varepsilon\|\mathbf{b}\|$ **do**
  begin

$$\alpha_k := \frac{(\mathbf{r_k}, \mathbf{p_k})}{(\mathbf{p_k}, A\mathbf{p_k})};$$

$$\mathbf{x_{k+1}} := \mathbf{x_k} + \alpha_k \mathbf{p_k};$$

$$\mathbf{r_{k+1}} := \mathbf{r_k} - \alpha_k A\mathbf{p_k};$$

$$\beta_k := -\frac{(\mathbf{r_{k+1}}, A\mathbf{p_k})}{(\mathbf{p_k}, A\mathbf{p_k})};$$

$$\mathbf{p_{k+1}} := \mathbf{r_{k+1}} + \beta_k \mathbf{p_k};$$

  end

ここに掲げた「古典的な」CG 法では、実際に解ける問題がかなり限定される。近年は**前処理** (**preconditioning**) と呼ばれるテクニックを併用した**前処理つき共役勾配法** (**preconditioned CG method**, 略して **PCG 法**) が広く使われている。前処理でやっていることは、与えられた問題を、固有値が密集している係数行列を持つ問題に変換することであると言えるが、具体的にどのような前処理を採用すべきかは、係数行列 A の性質に強く依存する。(現段階では扱わないことにしよう。)

2 本日の課題

CG 法は初めてという人が多いと思われるので、「古典的」CG 法で連立 1 次方程式を解くプログラムを作って簡単な実験をしてもらうことにする。

[1]CG 法では丸め誤差がなければ有限 (未知数の個数 N 以下) 回の反復で真の解が得られるので、直接法的な性格も持っていると言える。しかし実際上扱う問題では、N よりかなり小さな回数の反復で、十分な精度の解が得られる (またそれ以上反復しても精度は改善されない) ので、反復法的な性格の方が強い、と見るのが妥当である。

[2]係数行列 A が正定値でない場合にも適用できる同様の方法が開発されている。

(1) 前節の記述や前回の講義内容を参考にして、CG 法で連立 1 次方程式を解くプログラムを作る。連立 1 次方程式を解く部分はなるべく副プログラム (FORTRAN ならば subroutine, C ならば function) として実現すること。反復の各段階において、誤差、残差のノルム等を出力して収束の様子が見られるようにする。

(2) 未知数の個数 N がごく小さな問題を自作して、プログラムの動作を確認する。

(3) 係数行列 A が 3 重対角行列である問題を解くためのプログラムを作る (1. で作ったプログラムの中の、行列とベクトルの積を計算するところだけ書き換えれば良い)。

(4) 対角線上の成分の値が 4, その両隣りの成分が 1 である 50 次の三重対角行列を A' とする:

$$A' = \begin{pmatrix} 4 & 1 & & & \\ 1 & 4 & 1 & & \mathbf{0} \\ & \cdots & \cdots & & \\ & & \cdots & \cdots & \\ & & 1 & 4 & 1 \\ \mathbf{0} & & & 1 & 4 \end{pmatrix}.$$

さらに適当な解ベクトル $x^* \in \mathbf{R}^{100}$ を選んで、以下の実験をして、その結果を分析する。

(4-i)

$$A^{(1)} \equiv \begin{pmatrix} A' & O \\ O & A' \end{pmatrix}, \qquad b^{(1)} \equiv A^{(1)} x^*$$

として方程式 $A^{(1)}x = b^{(1)}$ を解け。

(4-ii) (4-i) の方程式のうち、最初の 50 行を c 倍 ($c = 10$, あるいは 100) した問題、すなわち

$$A^{(2)} \equiv \begin{pmatrix} cA' & O \\ O & A' \end{pmatrix}, \qquad b^{(2)} \equiv A^{(2)} x^*$$

として作った方程式 $A^{(2)}x = b^{(2)}$ を解け。

(4-iii) (4-i) の方程式のうち、すべての行を c 倍 ($c = 10, 100$) した問題、すなわち

$$A^{(3)} \equiv \begin{pmatrix} cA' & O \\ O & cA' \end{pmatrix}, \qquad b^{(3)} \equiv A^{(3)} x^*$$

として作った方程式 $A^{(3)}x = b^{(3)}$ を解け。

解説: 実は上記の実験 (4) は、最も原始的な前処理の一つであるスケーリング[3]の効用を納得するためのものである。もともと (4-i) のような方程式は固有値があまり散らばっていないため、CG 法にとっては比較的扱い易い問題であると言えるが[4]、(4-ii) のような問題に変換してしまうと、固有値の存在範囲が広がってしまい、収束が悪くなる (反復回数が増える) はずである。一方で (4-iii) になると、固有値の広がり具合はまた (4-i) と同等なものに戻り、反復回数も同程度になると予想される。

[3] これが有効な場合があることは古く ('70 年代以前) から分かっていた。

[4] とはいえ、このような三項方程式で Gauss の消去法と競争したら勝てないであろう。あくまで実験用の問題。