Algorithmische Mathematik II

Sommersemester 2010
Prof. Dr. Mario Bebendorf
Jos Gesenhues

universitätbonn

Übungsblatt 12.

Lösung erscheint am **Mittwoch, 21.07.2010**.

Aufgabe 1. Gegeben sei das lineare Gleichungssystem $Ax = b$, wobei

$$A = (a_{i,j})_{i,j=1,\dots,1000}, \quad a_{i,j} = \begin{cases} 100, & i = j \\ 1, & |i-j| = 1 \\ 0, & |i-j| > 1 \end{cases}$$

und

$$b = (b_i)_{i=1\dots,1000}, \quad b_i = 100,\ i = 1, \dots, 1000.$$

Bestimme einen Vektor $\hat{x}$, der in der Maximumsnorm $\|\cdot\|_\infty$ von der exakten Lösung des Systems um weniger als $1/10$ abweicht.

Lösung. Wir wählen das Jacobi-Verfahren, also

$$T_{i,j} = \left(-A_D^{-1}(A_L + A_R)\right)_{i,j} = \begin{cases} -\frac{1}{100}, & |i-j| = 1 \\ 0, & \text{sonst.} \end{cases}$$

Dann gilt $q = \|T\|_\infty = 2/100 < 1$ und nach Satz 12.9 (i) konvergiert das Verfahren für jeden Startwert $x^{(0)} \in \mathbb{R}^{1000}$. Sei nun $x^{(0)} = 0 \in \mathbb{R}^{1000}$. Ein Schritt des Jacobiverfahrens liefert

$$x^{(1)} = -A_D^{-1}(A_L + A_R)x^{(0)} + A_D^{-1}b = A_D^{-1}b = (1, \dots, 1)^T$$

und es gilt

$$\|x^{(1)} - x^{(0)}\|_\infty = 1.$$

Satz 12.9 (ii) besagt

$$\|x - x^{(k)}\|_\infty \le \frac{q^k}{1-q}\|x^{(1)} - x^{(0)}\|_\infty = \frac{1}{49 \cdot 50^{k-1}},$$

was zeigt, dass $\|x - x^{(1)}\|_\infty < 1/10$ gilt.
Das heißt, $\hat{x} = (1, \dots, 1)^T$ erfüllt das Geforderte. □

Aufgabe 2. a) Zeige, dass durch

$$6x_{n+1} = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \end{pmatrix} x_n + \begin{pmatrix} 3 \\ 2 \\ 2 \\ 3 \end{pmatrix}$$

eine für alle $x_0 \in \mathbb{R}^4$ konvergente Iteration gegeben ist.

b) Gib ein lineares Gleichungssystem an, gegen dessen Lösung das angegebene Verfahren konvergiert.

c) Schätze die Anzahl der Iterationen ab, die bei $x_0 = (0,0,0,0)^T$ benötigt werden, bis

$$\|x_k - x\|_\infty \leq 10^{-5}$$

gilt. (Dabei ist x die Lösung des Gleichungssystems aus b.)

Lösung.

a) Sei $T = 1/6 \cdot \left(\begin{smallmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \end{smallmatrix}\right)$. Dann gilt $q = \|T\|_\infty = 2/3 < 1$ und nach Satz 12.9 (i) ist das Verfahren konvergent.

b) Es sei $T = I - M^{-1}A$ und $M^{-1}b = c = 1/6 \cdot (3,2,2,3)^T$. Der Einfachheit halber wählen wir $M = I$. Dann gilt

$$A = I - T = \frac{1}{6}\begin{pmatrix} 5 & -1 & -1 & 0 \\ -1 & 5 & -1 & -1 \\ -1 & -1 & 5 & -1 \\ 0 & -1 & -1 & 5 \end{pmatrix} \quad \text{und} \quad b = \frac{1}{6}\begin{pmatrix} 3 \\ 2 \\ 2 \\ 3 \end{pmatrix}$$

und das gesuchte Gleichungssystem lautet $Ax = b$.

c) Es gilt

$$\|x_k - x\|_\infty \leq \frac{q^k}{1-q}\|x_1 - x_0\|_\infty = 3 \cdot \frac{2^k}{3^k} \cdot \frac{1}{6} \cdot 3 = \left(\frac{2}{3}\right)^{k-1} \overset{!}{\leq} 10^{-5}.$$

Daraus folgt

$$\frac{\left(\frac{2}{3}\right)^{k-1}}{\log_{10}\left(\frac{2}{3}\right)} \geq \frac{10^{-5}}{\log_{10}\left(\frac{2}{3}\right)}$$

und damit

$$\frac{\log_{10}\left(\frac{2}{3}\right)^{k-1}}{\log_{10}\left(\frac{2}{3}\right)} = \log_{2/3}\left(\frac{2}{3}\right)^{k-1} = k - 1 \geq \frac{\log_{10} 10^{-5}}{\log_{10}\left(\frac{2}{3}\right)} = \frac{-5}{\log_{10}\left(\frac{2}{3}\right)} \approx 28,39.$$

Also muss $k \geq 30$ gelten.

□

Aufgabe 3. Bestimme die Lösung des linearen Gleichungssystems $Ax = b$ mit

$$A = \begin{pmatrix} 1 & 1 \\ 1 & 3 \end{pmatrix}, \; b = \begin{pmatrix} 0 \\ 2 \end{pmatrix},$$

mit Hilfe des Verfahrens der konjugierten Gradienten mit dem Startvektor $(0,0)^T$ von Hand.

Lösung. In der Notation der Vorlesung ergibt sich folgende Rechnung.

$$x_0 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad r_0 = b - Ax_0 = \begin{pmatrix} 0 \\ 2 \end{pmatrix} - \begin{pmatrix} 1 & 1 \\ 1 & 3 \end{pmatrix}\begin{pmatrix} 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 2 \end{pmatrix} = p_0, \quad \alpha_0 = \frac{p_0^T r_0}{p_0^T A p_0} = \frac{1}{3}$$

$$x_1 = x_0 + \alpha_0 p_0 = \begin{pmatrix} 0 \\ 2/3 \end{pmatrix}, \quad r_1 = b - Ax_1 = \begin{pmatrix} -2/3 \\ 0 \end{pmatrix}, \quad \gamma_0 = -\frac{r_1^T A p_0}{p_0^T A p_0} = \frac{1}{9},$$

$$p_1 = r_1 + \gamma_0 p_0 = \begin{pmatrix} -2/3 \\ 2/9 \end{pmatrix}, \quad \frac{p_1^T r_1}{p_1^T A p_1} = \frac{3}{2}$$

$$x_2 = x_1 + \alpha_1 p_1 = \begin{pmatrix} -1 \\ 1 \end{pmatrix}, \quad r_2 = b - Ax_2 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

□

Programmieraufgabe 1. Programmiere das SOR-Verfahren. Teste das Programm am Gleichungssystem $Ax = b$ mit

$$A = (a_{i,j})_{i,j=1,\dots,n}, \quad a_{i,j} = \begin{cases} 2, & i = j \\ -1, & |i-j| = 1 \\ 0, & |i-j| > 1 \end{cases}$$

und

$$b = (b_i)_{i=1,\dots,n}, \quad b_i = 1,\ i = 1, \dots, n,$$

für $n = 100$, $\omega = 0.6$, 1, 1.4, $k = 10$, 100, 1000 und $x_0 = (0, \dots, 0)^T$.

Lösung.

```
#include <iostream>
#include <cmath>
void sorschritt(double omega, unsigned n, double * a,//
        double * b, double * x);

int main (){
  // Dimension des lin. Gleichungssystems
  unsigned n = 100;
  // Anzahl Iterationsschritte
  unsigned k = 1000;
  // Relaxionsparameter
  double omega = 1;
  // Initialisierung des Gleichungssystems
  double * a = new double[n*n];
  double * b = new double[n];
  double * x = new double[n];
  for (unsigned i=0; i<n; i++){
    b[i]=1;
    x[i]=0;
    for (unsigned j=0; j<n; j++){
      if (i==j)
    a[n*i+j]=2;
      else if ( i-j == 1 || j-i==1 )
    a[n*i+j]=-1;
      else a[n*i+j]=0;
    }
  }

  for (unsigned i=0; i<k; i++)
    sorschritt(omega, n, a, b, x);

  for (unsigned i=0; i<n; i++)
    std::cout << x[i] << std::endl;
  return 0;
}

void sorschritt(double omega, unsigned n, double * a,//
        double * b, double * x){
  double sum;
  // Der SOR-Algorithmus wie in der Vorlesung angegeben
  for ( unsigned i=0; i<n; i++ ){
```

```
    sum=0;
    for ( unsigned j=0; j<n; j++)
      if (j!=i)
    sum+=a[n*i+j]*x[j];
    x[i]=(1.-omega)*x[i]+omega/a[(n+1)*i]*(b[i]-sum);
  }
}
```

□