Prof. Dr. Karsten Urban
Dipl.-Math. oec. Klaus Stolle
Institut für Numerische Mathematik
Universität Ulm

WiSe 2014/2015
Blatt 3
31.10.2014

Angewandte Numerik 2

Abgabetermin: Freitag, 07.11.2014, **vor der Übung**

Für dieses Übungsblatt gibt es 16 Theorie- und 20 Matlab-Punkte, sowie 8 Theorie- und 8 Matlab-Zusatzpunkte. Punkte, die mit einem * gekennzeichnet sind, sind Zusatzpunkte.
Die 50-Prozent-Grenzen liegen aktuell (inklusive Blatt 3) bei 32,5 Theoriepunkten und 26 Matlabpunkten.

Aufgabe 11 *(Jakobi-Verfahren und Gauß-Seidel-Verfahren)* (8T*+4T+8T+4T Punkte)

a) Führen Sie von Hand für das lineare Gleichungssystem

$$\begin{pmatrix} 4 & 1 & 1 \\ 1 & 4 & 1 \\ 0 & 1 & 2 \end{pmatrix} x = \begin{pmatrix} 6 \\ 6 \\ 3 \end{pmatrix}$$

jeweils zwei Schritte des Jakobi- und des Gauß-Seidel-Verfahrens durch, ohne dabei Inverse von Matrizen zu berechnen. Wählen Sie als Startvektor den Vektor $x^{(0)} = (0,0,0)^T$ und geben Sie alle Ihre Rechenschritte an.

b) Können Sie für das obige Beispiel Aussagen machen, ob die beiden Verfahren konvergieren?

c) Leiten Sie für das Jakobi- und das Gauß-Seidel-Verfahren jeweils eine komponentenweise Darstellung der Iterationsvorschrift her.

d) Moderne Computer, die für Simulationen eingesetzt werden, haben mehrere Prozessoren, die wiederum jeweils mehrere Prozessorkerne haben. Bei der Parallelisierung versucht man, Berechnungen wie beispielsweise einen Iterationsschritt eines Iterations-Verfahrens auf mehrere dieser Prozessorkerne so aufzuteilen, dass die einzelnen Arbeitsschritte dort gleichzeitig und möglichst unabhängig voneinander durchgeführt werden können.
Diskutieren Sie: Welches der beiden Verfahren lässt sich besser parallelisieren?

Aufgabe 12 *(Programmieraufgabe: Jacobi- und Gauß-Seidel-Verfahren)* (15M Punkte)

In dieser Aufgabe sollen sie das Jacobi- und das Gauss-Seidel-Verfahren vergleichen. Schreiben Sie dazu ein Matlabskript `vergleichJacobiGaussSeidel` in dem Sie für eine zufällige, diagonaldominante 1000×1000 - Matrix A und einen zufälligen 1000×1 - Vektor b das Gleichungssystem $Ax = b$ mit beiden Verfahren lösen. Gehen Sie dabei wie folgt vor.

(1) Initialisieren Sie die Matrix A, den Vektor b und den Startvektor $x0$, der nur Nullen enthält.
Achtung: A soll zufällig und diagonaldominant sein.

(2) Berechnen Sie die exakte Lösung x^* mit dem Backslash-Operator `'\'`.

(3) Führen Sie jeweils 12 Schritte des Jacobi- und des Gauß-Seidel-Verfahrens durch. Berechnen Sie für jeden Schritt den Fehler $||x^{(k)} - x^*||_2$. (Sie dürfen für die Norm die Matlab-Funktion `norm` verwenden.)

(4) Plotten Sie beide Fehler über k in eine Grafik. Wählen Sie die Skalierung der Achsen so, dass näherungsweise Geraden entstehen.

Achten Sie darauf, dass ihre Iterationen effizient ausgeführt werden. Insbesondere ist es effizienter, die Iterationsmatrizen für die beiden Verfahren vor der Iteration nur einmal zu berechnen und nicht in jedem Iterationsschritt neu.
Was lässt sich über das Konvergenzverhalten der beiden Verfahren sagen?

Aufgabe 13 *(Programmieraufgabe: Richardson-Verfahren)* (5M+6M*+2M* Punkte)

In dieser Aufgabe sollen Sie den optimalen Wert für den Parameter $\omega = \frac{1}{\gamma}$ des Richardson-Verfahrens

$$x^{(k+1)} = (I - \omega A)x^{(k)} + \omega b = x^{(k)} - \omega(Ax^{(k)} - b), \qquad k = 0, 1, 2, \dots$$

mit Hilfe eines numerischen Experiments bestimmen.
Dabei können Sie folgendermaßen vorgehen:

a) Schreiben Sie eine Matlabfunktion `x = richardson(A, b, x0, omega, N)`, die `N` Schritte des Richardson-Verfahrens zur Lösung des linearen Gleichungssystems $Ax = b$ durchführt. `x0` soll dabei als Startwert verwendet werden, und `omega` ist der Parameter für das Richardson-Verfahren. Ihre Funktion soll den Näherungswert für die Lösung des linearen Gleichungssystem nach `N` Schritten des Richardson-Verfahrens zurück geben.

b) Schreiben Sie ein Matlab-Skript `runRichardson`, in dem Sie

   (1) die Matrix A, die rechte Seite b und den Startwert x_0 initialisieren. Verwenden Sie

$$A = \begin{pmatrix} 2 & -1 & 1 \\ -1 & 2 & -1 \\ 1 & -1 & 2 \end{pmatrix}, \quad b = \begin{pmatrix} 2 \\ 0 \\ 2 \end{pmatrix} \quad \text{und} \quad x_0 = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}.$$

   (2) die exakte Lösung x^* des linearen Gleichungssystems $Ax = b$ unter Verwendung des \-Operators berechnen,

   (3) `omega` als einen Vektor mit 200 Werten aus dem Intervall $[0.25, 0.45]$ initialisieren,

   (4) für jeden Wert von `omega` die Näherungslösung $x^{(20)}$ des Richardson-Verfahrens nach `N = 20` Iterationsschritten sowie den Fehler $||x^{(20)} - x^*||_2$ dieser Näherungslösung berechnen (für die Norm können Sie die Matlabfunktion `norm` verwenden) und

   (5) die Fehler über `omega` plotten.

c) Welches ist der optimale Wert für den Parameter `omega`?

Hinweise:
Die Programmieraufgaben sind in Matlab zu erstellen. Senden Sie alle Files in einer E-mail mit dem Betreff **Loesung-Blatt03** an **angewandte.numerik@uni-ulm.de** (Abgabetermin jeweils wie beim Theorieteil). Drucken Sie zusätzlich allen Programmcode sowie die Ergebnisse aus und geben Sie diese vor der Übung ab. Der Source Code sollte strukturiert und, wenn nöbig, dokumentiert sein.