Institut für Informatik
Dr. Tobias Neckel
Christoph Kowitz
Alfredo Parra

Technische Universität München
WS 2015/2016

Einführung in die wissenschaftliche Programmierung

Übungsblatt 12

1.) LR Zerlegung

Oft will man ein lineares Gleichungssystem $Ax = b$ für verschiedene rechte Seiten $b_0, b_1, \ldots$ lösen. Dann bietet es sich an, den Gaußalgorithmus als LR Zerlegung zu interpretieren, d.h. man schreibt die Schritte während der Gaußelimination in einer linken unteren Dreiecksmatrix L mit. Man erhält dann eine Zerlegung von A in eine linke untere Dreiecksmatrix L und eine rechte obere Dreiecksmatrix R, z.B.

$$\begin{pmatrix} 3 & 2 & 1 \\ 6 & 6 & 3 \\ 9 & 8 & 6 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ 3 & 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} 3 & 2 & 1 \\ 0 & 2 & 1 \\ 0 & 0 & 2 \end{pmatrix} = L \cdot R$$

Will man dann das System $Ax = b$ lösen, so kann zuerst durch Vorwärtssubstitution $Ly = b$ und anschließend durch Rückwärtssubstitution $Rx = y$ bestimmt werden. Beides ist in $\mathcal{O}(n^2)$ Schritten möglich, und damit billiger als der Gaußalgorithmus.

Wandeln Sie den Gaußalgorithmus der Vorlesung so um, dass die LR Zerlegung berechnet wird. Schreiben Sie dann eine Funktion um mit den beiden Matrixen L und R das Gleichungssystem $Ax = b$ zu lösen.

Hausaufgabe

Beim Eliminieren der Einträge einer Spalte der Matrix A kann es vorkommen, dass man durch Null (oder eine sehr kleine Zahl) teilt. Um diese Problematik zu veranschaulichen, versuchen Sie das System $Ax = b$ mit

$$A = \begin{pmatrix} 10^{-20} & 1 \\ 1 & 1 \end{pmatrix}, \quad b = \begin{pmatrix} 1 \\ 2 \end{pmatrix},$$

mit Hilfe von einer normalen LR Zerlegung zu lösen. Nennen wir die numerische Lösung x^*. Wie groß ist das Residuum $r = b - Ax^*$?

Um dieses Problem zu überwinden, sucht man den betragsmäßig größten Eintrag der Spalte (das *Pivotelement*) und vertauscht die entsprechenden Reihen der Matrix (Vorsicht: die Reihen der rechten Seite b müssen auch vertauscht werden!). Implementieren Sie diese Pivotstrategie und lösen Sie nochmal das obere System. Überprüfen Sie nochmal, dass $b - Ax^* \to 0$.

2.) Verfahren des steilsten Abstiegs

Das Verfahren des steilsten Abstiegs ist eine der bekanntesten iterativen Methoden zum Lösen von linearen Gleichungssystemen $Ax = b$ mit symmetrisch positiv definiter (spd) Matrix $A \in \mathbb{R}^{N \times N}$ und $b, x \in \mathbb{R}^N$. Beim Lösen der Wärmeleitungsgleichung mit Finite Differenzen wie in der Vorlesung entsteht eine spd Matrix.

Ist die Matrix A des Gleichungssystems $Ax = b$ spd, dann entspricht das Lösen des Gleichungssystems der Minimierung der quadratischen Funktion

$$f(x) = \frac{1}{2} x^T A x - b^T x.$$

Das Verfahren sucht nach dem Minimum von $f(x)$ in Richtung des Residuums $r = b - Ax$, welches der Richtung des steilsten Abstiegs entspricht (warum?). Ohne erstmal auf weitere Details einzugehen[1], ergibt sich der Algorithmus

i) Wähle Startwert x_0 und setze $r_0 = b - Ax_0$, $k = 0$

ii) While $k < k_{\max}$ und $\|r_k\| > \epsilon$

   (a) Berechne Schrittweite
   $$\alpha_k = \frac{r_k^T r_k}{r_k^T A r_k}$$

   (b) Berechne neuen Ort x_{k+1}
   $$x_{k+1} = x_k + \alpha_k r_k$$

   (c) Update Residuum
   $$r_{k+1} = b - Ax_{k+1}$$

   (d) Update
   $$k = k + 1$$

[1] Sollten Sie diese Verfahren noch nicht kennen bzw. an weiteren Details interessiert sein, so bietet `http://www.cs.cmu.edu/ quake-papers/painless-conjugate-gradient.pdf` eine gute Einführung.

Implementieren Sie diesen Algorithmus und lösen Sie damit die Wärmeleitungsgleichung wie in der Vorlesung besprochen. Setzen Sie beispielweise

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

und $k_{\max} = 100$, $\epsilon = 10^{-8}$. Verwenden Sie die Norm $\|r\| \equiv \sqrt{\frac{1}{N}\sum_{i=0}^{N} r_i^2}$.

Hausaufgabe

Erhöhen Sie nun die Dimension des oberen Systems: testen Sie Ihren Algorithmus für $N = 8, 16, 32, \ldots$ Wie verhält sich die Anzahl der Iterationen hinsichtlich der Problemgröße?