Lösung von linearen Gleichungssystemen

Aufgabe 1: Gauß'sches Eliminationsverfahren

Gegeben sei ein lineares Gleichungssystem $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$, für das die Lösung $\boldsymbol{x} = \boldsymbol{A}^{-1}\boldsymbol{b}$ gesucht ist. Ermitteln Sie die Lösung numerisch mithilfe des Gauß'schen Eliminationsverfahren ohne Zeilentausch. Erstellen Sie dazu eine Funktion (**Fkt. XIV** `solveGauss`), die ausgehend von einer Matrix $\boldsymbol{A}$ und einem Vektor $\boldsymbol{b}$ die Lösung $\boldsymbol{x}$ zurückgibt.

Aufgabe 2: Gradienten-Methode

Gegeben sei ein lineares Gleichungssystem $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$, für das die Lösung $\boldsymbol{x} = \boldsymbol{A}^{-1}\boldsymbol{b}$ gesucht ist. Ermitteln Sie die Lösung numerisch mithilfe der Gradienten-Methode. Erstellen Sie dazu eine Funktion (**Fkt. XV** `solveG`), die ausgehend von einer Matrix $\boldsymbol{A}$ und einem Vektor $\boldsymbol{b}$ die Lösung $\boldsymbol{x}$ zurückgibt. Als Abbruchkriterium soll der maximale Wert der l_2-Norm des Residuenvektors $\|\boldsymbol{r}^{(k)}\|_2 < r_{\mathrm{tol}}$ sowie eine maximale Anzahl an Iteration $iter_{\mathrm{max}}$ vorgegeben werden.

Aufgabe 3: Konjugierte Gradienten-Methode

Gegeben sei ein lineares Gleichungssystem $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$, für das die Lösung $\boldsymbol{x} = \boldsymbol{A}^{-1}\boldsymbol{b}$ gesucht ist. Ermitteln Sie die Lösung numerisch mithilfe der konjugierten Gradienten-Methode. Erstellen Sie dazu eine Funktion (**Fkt. XVI** `solveCG`), die ausgehend von einer Matrix $\boldsymbol{A}$ und einem Vektor $\boldsymbol{b}$ die Lösung $\boldsymbol{x}$ zurückgibt. Als Abbruchkriterium soll der maximale Wert der l_2-Norm des Residuenvektors $\|\boldsymbol{r}^{(k)}\|_2 < r_{\mathrm{tol}}$ sowie eine maximale Anzahl an Iteration $iter_{\mathrm{max}}$ vorgegeben werden.

Lösen Sie das lineare Gleichungssystem des Finite-Elemente-Beispiels aus den Arbeitsblättern 7 und 8 mit Ihren selbst programmierten Gleichungslösern. Vergleichen Sie die Dauer der Berechnung mit der des Matlab-Lösers. (Hinweis: Funktionen `tic, toc` zur Zeitmessung)

Aufgabe 4: Vergleich der Methoden

Wenden Sie das Gauß-Verfahren, die Gradienten-Methode sowie die Konjugierte Gradienten-Methode auf das folgende lineare Gleichungssystem $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ an:

$$\boldsymbol{A} = \begin{pmatrix} 1 & -2 & 0 & 0 & \dots & 0 & 0 & 0 & 0 \\ -2 & \phi & -2 & 0 & \dots & 0 & 0 & 0 & 0 \\ 0 & -2 & \phi & -2 & \dots & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & -2 & \phi & -2 & 0 \\ 0 & 0 & 0 & 0 & \dots & 0 & -2 & \phi & -2 \\ 0 & 0 & 0 & 0 & \dots & 0 & 0 & -2 & \phi \end{pmatrix}, \qquad \boldsymbol{b} = \mathbf{1}$$

wobei $\boldsymbol{A}$ eine $(n \times n)$-Matrix und $\boldsymbol{b}$ ein Vektor der Länge n mit den Einträgen $b_i = 1$ ist. Zum Aufstellen der Matrix ist die Matlab-Funktion `diag()` hilfreich.

Für

$$\phi = [10.0, 6.0, 5.1, 5.01, 5.001, 5.00001, 5.0000001, 5.000000001, 5.00000000001]$$

soll die Lösung mithilfe der drei verschiedenen Verfahren ermittelt werden. Setzen Sie $n = 300$. Wählen Sie für die iterativen Verfahren eine geeignete Toleranz r_{tol} und begrenzen Sie die maximale Anzahl der Iterationen.

Analysieren Sie folgende Punkte und begründen Sie Ihre Ergebnisse:

- Anzahl der benötigten Iterationen für verschiedene ϕ der iterativen Verfahren.
- Lösungszeit für verschiedene ϕ aller Verfahren. (Hinweis: Funktionen `tic`, `toc` zur Zeitmessung)
- Entwicklung der Lösungszeit des Gauß-Verfahrens sowie der Konjugierten Gradienten-Methode bei Erhöhung der Systemgröße ($n > 300$, maximale Größe durch Rechenleistung begrenzt!)
- Vergleichen Sie die Lösungszeit mit der des Matlab-Lösers.

Hinweis: Manche Verfahren liefern kein Ergebnis für spezielle Varianten der Problemstellung.

Matlab-Funktionen

Folgende Funktionen sollen bei der Bearbeitung dieses Aufgabenblattes erstellt werden, da diese für spätere Aufgabenblätter wiederverwendet werden sollen. Erstellen Sie die Funktionen in Matlab und speichern Sie diese in eigenen *.m-files ab.

Für die folgenden Funktionen sind (A ... Matrix $\boldsymbol{A}$) sowie (b ... Vektor $\boldsymbol{b}$) zum Gleichungssystem $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$.

- **Fkt. XIV**: `function x = solveGauss(A,b)`

  Rückgabewert: Spaltenvektor mit Lösung $\boldsymbol{x}$

  Testen Sie die Funktion mit:

  $$([10.0, 2, 1; 3, 4, 4; 1, 8, 4], [1; 1; 2]) \rightarrow [0.051282051282051; 0.275641025641026; -0.064102564102564]$$

Für die folgenden Funktionen sind Eingabeparameter wie folgt definiert:
A ... Matrix $\boldsymbol{A}$ des Gleichungssystems $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$,
b ... Vektor $\boldsymbol{b}$ des Gleichungssystems $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$,
x0 ... Startvektor für die Lösung $\boldsymbol{x}_0$,
rtol ... Zu erreichende Toleranz der L2-Norm des Residuumsvektors $\boldsymbol{r} = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}$,
itermax ... Maximale Anzahl an Iteration bei der das Verfahren gestoppt wird.

- **Fkt. XV**: `function x = solveG(A,b,x0,rtol,itermax)`

  Rückgabewert: Vektor mit Lösung $\boldsymbol{x}$

  Teste die Funktion mit:

  $$([10.0, 2.0, 10.0; 2.0, 40.0, 8.0; 10.0, 8.0, 60.0], [1.0; 1.0; 2.0], [0.0; 0.0; 0.0], 1e\text{-}7, 1000)$$
  $$\rightarrow [0.078600816156673; 0.017489712225719; 0.017901236448326]$$

  nach 55 Iterationen

- **Fkt. XVI**: `function x = solveCG(A,b,x0,rtol,itermax)`

  Rückgabewert: Vektor mit Lösung $\boldsymbol{x}$

  Teste die Funktion mit:

  $$([10.0, 2.0, 10.0; 2.0, 40.0, 8.0; 10.0, 8.0, 60.0], [1.0; 1.0; 2.0], [0.0; 0.0; 0.0], 1e\text{-}7, 1000)$$
  $$\rightarrow [0.078600823045267; 0.017489711934156; 0.017901234567901]$$

  nach 3 Iterationen