Karlsruher Institut für Technologie
Institut für Algebra und Geometrie
PD Dr. T. Arens
Dipl.-Math. S. Heumann

Karlsruhe, den 18.11.2010

Inverse Probleme – Matlab Übungen

Vorbereitungen

Öffnen Sie ein Terminalfenster. Erzeugen Sie mit `mkdir uebung2` ein Verzeichnis und wechseln Sie mit `cd uebung2` hinein. Kopieren Sie vorbereitete Programmdateien:

```
cp /misc/arens/uebung2/* .
```

Starten Sie nun Matlab mit `matlab &`

Numerisches Differenzieren

Wir betrachten die Operatorgleichung $Kx = y$ mit $K \in \mathcal{K}(L^2(0,1))$ definiert durch

$$Kx(t) = \int_0^t x(s)\,\mathrm{d}s\,, \qquad t \in [0,1]\,.$$

Als rechte Seiten werden wir die Funktionen

$$y_1(t) = \begin{cases} \frac{2}{3}\,t^3 - t, & 0 \le t \le \frac{1}{2}, \\ -\frac{2}{3}\,t^3 + 2\,t^2 - 2\,t + \frac{1}{6}, & \frac{1}{2} < t \le 1\,, \end{cases} \qquad y_2(t) = \begin{cases} \frac{1}{2}\,t^2, & 0 \le t \le \frac{1}{2}, \\ -\frac{1}{2}\,t^2 + t - \frac{1}{4}, & \frac{1}{2} < t \le 1\,, \end{cases}$$

$$y_3(t) = \begin{cases} t, & 0 \le t \le \frac{1}{2}, \\ \frac{1}{2}, & \frac{1}{2} < t \le 1\,, \end{cases}$$

verwenden. Die zugehörigen exakten Lösungen sind

$$x_1(t) = \begin{cases} 2\,t^2 - 1, & 0 \le t \le \frac{1}{2}, \\ -2\,t^2 + 4\,t - 2, & \frac{1}{2} < t \le 1\,, \end{cases} \qquad x_2(t) = \begin{cases} t, & 0 \le t \le \frac{1}{2}, \\ -t + 1, & \frac{1}{2} < t \le 1\,, \end{cases}$$

$$x_3(t) = \begin{cases} 1, & 0 \le t \le \frac{1}{2}, \\ 0, & \frac{1}{2} < t \le 1\,. \end{cases}$$

Es gilt übrigens $x_1 \in K^*K(L^2(0,1))$ und $x_2 \in K^*(L^2(0,1))$.

Der Operator K besitzt das singuläre System $(\mu_n, v_n, w_n)_n$ mit $\mu_n = \frac{2}{(2n-1)\,\pi}$,

$$v_n = \sqrt{2}\,\cos\left(\frac{\cdot}{\mu_n}\right) \quad \text{und} \quad w_n = \sqrt{2}\,\sin\left(\frac{\cdot}{\mu_n}\right), \qquad n \in \mathbb{N}\,.$$

Damit stellt sich die Spectral-Cutoff-Regularisierung $R_\alpha : L^2(0,1) \to L^2(0,1)$ dar als

$$R_\alpha y(t) = \sum_{\mu_n^2 \ge \alpha} \frac{2}{\mu_n} \int_0^1 y(s)\,\sin\left(\frac{s}{\mu_n}\right)\mathrm{d}s\,\cos\left(\frac{t}{\mu_n}\right), \qquad t \in [0,1]\,.$$

Die in der Datei `test_convergence.m` implementierte Funktion überprüft die erzielte Konvergenzrate bei fehlerbehafteten Daten. Sie leistet folgendes:

- Es werden im Vector `t` Sampling-Punkte gespeichert. In diesem Punkten werden die rechte Seite `y` und die exakte Lösung `x_correct` ausgewertet. Diese sind in der als Parameter übergebenen Funktion `rhs1` implementiert.

- Es werden relative Fehler-Niveaus definiert und entsprechende Störungen zu `y` addiert. Die Ergebnisse werden als Spalten in der Matrix `y_delta` gespeichert.

- Für jede so erhaltene gestörte rechte Seite wird die Näherungslösung mit Spectral-Cutoff berechnet und in einer Spalte der Matrix `x_approx` gespeichert. Dabei wird $\alpha = \delta$ gewählt.

- Die Ergebnisse werden ausgedruckt: In jeder Tabellenzeile das relative Fehlerniveau, der Fehler δ in einer diskreten L^2-Norm, die Anzahl N von Gliedern in der Summe in R_α, der Fehler zwischen `x_correct` und `x_approx` in der diskreten L^2-Norm und dieser Fehler geteilt durch $\sqrt{\delta}$. Dieser Wert sollte ungefähr konstant sein, wenn eine Konvergenzrate von $\sqrt{\delta}$ erzielt wird.

Aufgabe 1 Öffnen Sie die Datei `test_convergence.m` im Editor (Menü *File* $\rightarrow$ *Open*) und vollziehen Sie die einzelnen Schritte nach.

Aufgabe 2 Überprüfen Sie die Strategie für Lösungen aus $K^*(L^2(0,1))$. Rufen Sie dazu Funktion `test_convergence` auf:

`[x_approx, x_correct, y, y_delta, t] = test_convergence(@rhs`n`);`

mit $n = 1, 2, 3$. Einen Plot der exakten und der Näherungslösungen erhalten Sie mit

`plot(t, x_correct, t, x_approx(:,`n`))`

wobei n eine Zahl zwischen 1 und der Spaltenzahl von `x_approx` ist.

Aufgabe 3 Sichern Sie eine neue Version `test_convergence2.m` der ursprünglichen Datei. Modifizieren Sie diese so, dass die Strategie für Lösungen aus $K^*K(L^2(0,1))$ überprüft werden kann. Rufen Sie wieder mit `rhs1`, `rhs2` und `rhs3` auf.

Aufgabe 4 Öffnen Sie auch die Datei `spectral_cut_off.m` und vollziehen Sie die Implementierung von R_α nach.

Aufgabe 5 Öffnen Sie ferner die Datei `difference_quotients.m`, die eine Implementierung des Differenzenquotienten-Operators aus Beispiel 2.12 enthält. Vollziehen Sie auch diese nach. Erzeugen Sie eine neue Version der Test-Funktion, die den Differenzenquotienten-Operator verwendet. Überprüfen Sie, dass die Konvergenzrate $\sqrt{\delta}$ für $x \in K^*(L^2(0,1))$ erreicht wird.

Aufgabe 6 Versuchen Sie, mit dem Differenzenquotienten-Operator für $x \in K^*K(L^2(0,1))$ eine höhere Konvergenzrate als $\sqrt{\delta}$ zu erreichen.