Numerische Methoden für komplexe Systeme, SoSe 2013

Dr. S. Gurevich

Übungen im WWW: `http://pauli.uni-muenster.de/tp/`

Aufgabe 1: Wettbewerbsmodell

Die Leistungsfähigkeit pseudospektraler Methoden gründet auf der effizienten Implementierung der Fouriertransformation, der als *fast fourier transform* (FFT) bekannte Algorithmus skaliert mit $N \log N$. Subroutinen zur Berechnung der Fouriertransformation sind von vielen Anbietern erhältlich und in den meisten numerischen Bibliotheken implementiert. Eine frei verfügbare FFT, die auf vielen Systemen lauffähig ist und von `C`, `C++` und `Fortran` aus aufrufbar ist, ist unter `www.fftw.org` erhältlich.

a) Besuchen Sie die homepage und machen Sie sich mit den Eigenschaften dieser numerischen Bibliothek vertraut. Versuchen Sie insbesondere, folgende Fragen zu beantworten:

- Wie installiert man diese Bibliothek?
- Wie kompiliert man ein Programm, das auf in der Bibliothek enthaltenen Subroutinen zurückgreift?
- Was ist ein *plan* und wozu dient er?
- Was ist der Unterschied zwischen *in place* und *out of place* Transformationen?
- Welche verschiedenen Arten von Transformationen werden angeboten? Was ist beispielsweise der Vorteil einer *real to real* Transformation?

b) Versuchen Sie nun ein Beispielprogramm zu erstellen, welches einen eindimensionalen *array* (reell oder komplex) in den Fourierraum transformiert. Wie kann man in der Programmiersprache Ihrer Wahl mit komplexen Zahlen umgehen? Wie sind die Fourierkoeffizienten in dem transformierten *array* angeordnet? Was ist bezüglich der Normierung zu beachten? Überprüfen Sie durch Rücktransformation, ob Sie wieder die ursprünglichen Werte erhalten.

Tipp für Frustrierte: In der Dokumentation sind zahlreiche Beispiele zu finden.

Aufgabe 2: Ableiten mit der FFT

Mit Hilfe der schnellen Fouriertransformation ist es besonders einfach, Ableitungen von Funktionen zu berechnen oder Signale einer Hoch- oder Tiefpassfilterung zu unterziehen. Dazu ist es wichtig, die Anordnung der Fourierkoeffizienten im transformierten *array* zu beachten. Sei f eine L-periodische Funktion, welche an den Stützstellen $f_1, \ldots, f_N$ gegeben ist. Falls f komplex ist, liegen im transformierten *array* die Fourierkoeffizienten in der Reihenfolge

$$\tilde{f}_0, \tilde{f}_1, \ldots, \tilde{f}_{\frac{N}{2}}, \tilde{f}_{-\frac{N}{2}+1}, \tilde{f}_{-\frac{N}{2}+2}, \ldots, \tilde{f}_{-1}$$

vor. Ist f eine reelle Funktion, so haben die Fourierkoeffizienten die Symmetrie $\tilde{f}_{-j} = \tilde{f}_j^*$. Dadurch benötigt man im Fourierraum nur einen nahezu halb so großen *array*, in dem die Fourierkoeffizienten in der Reihenfolge

$$\tilde{f}_0, \tilde{f}_1, \ldots, \tilde{f}_{\frac{N}{2}}$$

vorliegen.

Machen Sie sich klar: Um eine Ableitung zu berechnen, ist im Fourierraum jeder Fourierkoeffizient mit der entsprechenden Wellenzahl $k(j) = \frac{2\pi}{L} j$ und der imaginären Einheit zu multiplizieren. Schreiben Sie nun ein kurzes Programm, welches eine periodische Funktion im Ortsraum initialisiert, diese fouriertransformiert und im Fourierraum die Ableitung berechnet. Prüfen Sie an einfachen Beispielen, ob sich nach der Rücktransformation in den Ortsraum das richtige Ergebnis ergibt.