

Aufgabenblatt 3

mpgi4@cg.tu-berlin.de

WiSe 2013/2014

Allgemeine Hinweise:

- Die Aufgaben sollen in Gruppen bestehend aus zwei bis drei Personen bearbeitet werden. Ausnahmen müssen mit dem jeweiligen Tutor abgesprochen werden.
- Bitte reichen Sie Ihre Lösungen in Form einer ZIP Datei bis **Sonntag, den 22.12.2011, um 23:55 Uhr** auf der ISIS Webseite der Vorlesung ein. Wählen Sie dazu für Ihre Gruppe eine/n Sprecher/in, welche/r die Abgabe durchführt, und tragen Sie in jede abgegebene Datei Name und Email-Adresse aller Gruppenmitglieder ein.
- Wenn eine Aufgabe die Abgabe einer Grafik verlangt, dann muss ein vollständig funktionsfähiges Programm in der Lösung enthalten sein, welches bei der Ausführungen die Grafik erstellt. Wenn zum Erstellen einer Grafik bereits eine Funktion gegeben ist, dann darf diese nicht verändert werden.
- Tragen Sie in die Datei README.txt Name und Email-Adresse aller Gruppenmitglieder sowie einige kurze Kommentare zu Ihrer Abgabe ein. Weisen Sie auf Fehler oder Unvollständigkeiten in Ihrer Abgabe hin, dann gibt es Teilpunkte.

Aufgabe 1: Interpolation der Runge-Funktion (4 Punkte)

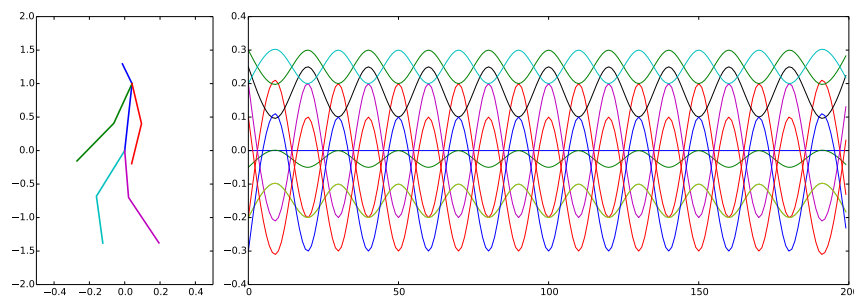
Die Runge-Funktion ist definiert durch:

$$f(x) = \frac{1}{1 + x^2} \quad (1)$$

- a) Erstellen Sie einen Plot der Runge-Funktion für das Intervall $[-5, 5]$ und geben Sie das Ergebnis in Form einer PDF Datei ab.
- b) Unterteilen Sie das Intervall $[-5, 5]$ äquidistant mit 3, ..., 11 Stützstellen. Bestimmen Sie unter Verwendung der Lagrangeschen Interpolationsformel für die Stützstellen das Interpolationspolynom, welches an den Stützstellen die Runge-Funktion interpoliert. Erstellen Sie für jede Wahl der Stützstellen einen Plot, welcher die Runge-Funktion, die Datenpunkte und das Interpolationspolynom zeigt. Nutzen Sie dazu die `subplot` Funktion von Matplotlib unter Verwendung eines 3×3 Rasters und geben Sie das Ergebnis in Form einer PDF Datei ab.
- c) Unterteilen Sie das Intervall $[-5, 5]$ äquidistant mit 8 Stützstellen und bestimmen Sie eine aus 7 kubischen Polynomen bestehende Interpolationsfunktion, welche die Funktionswerte und ersten Ableitungen der Runge-Funktion an den Stützstellen interpoliert. Plotten Sie die Runge-Funktion, die Datenpunkte und die kubischen Polynome und geben Sie das Ergebnis in Form einer PDF Datei ab.
- d) Unterteilen Sie das Intervall $[-5, 5]$ äquidistant mit 8 Stützstellen und bestimmen Sie einen aus 7 kubischen Polynomen bestehenden zweimal stetig differenzierbaren Spline (mit natürlichen Randbedingungen), welcher die Runge-Funktion an den Stützstellen interpoliert. Plotten Sie die Runge-Funktion, die Datenpunkte und den berechneten Spline und geben Sie das Ergebnis in Form einer PDF Datei ab.

Nutzen Sie für die Implementierung der Aufgaben die Dateien `aufgabe1a.py`, ..., `aufgabe1d.py` und speichern Sie erstellte Plots im Unterverzeichnis `./figures/`.

Aufgabe 2: Keyframe Animation (2 Punkte)

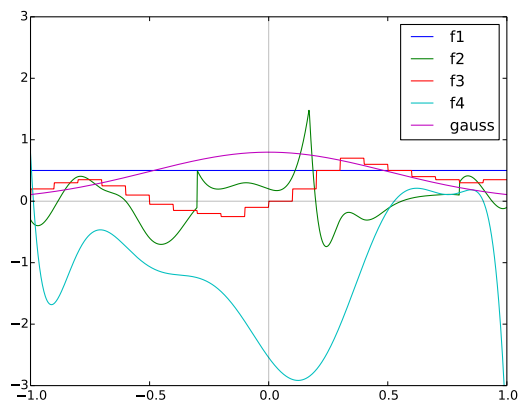


- Ersetzen Sie in der vorgegeben Keyframe Animation die lineare Interpolation durch einen kubischen Spline mit natürlichen Randbedingungen. Erstellen Sie einen Plot der interpolierten Parameter und geben Sie das Ergebnis in Form einer PDF Datei ab.
- Ersetzen Sie den kubischen Spline aus Aufgabenteil (a) durch einen kubischen Spline mit periodischen Randbedingungen. Erstellen Sie einen Plot der interpolierten Parameter und geben Sie das Ergebnis in Form einer PDF Datei ab.

Nutzen Sie für die Implementierung der Aufgaben die Dateien `aufgabe2a.py` und `undaufgabe2b.py` und speichern Sie erstellte Plots im Unterverzeichnis `./figures/`.

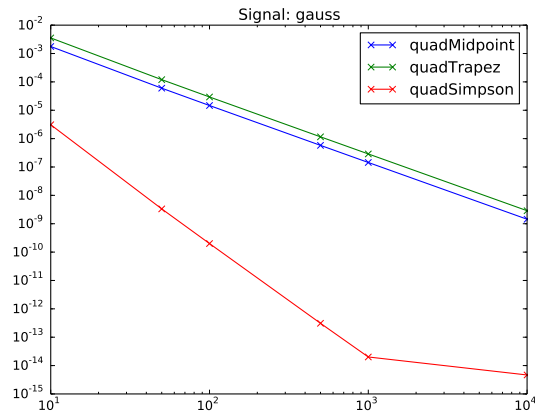
Aufgabe 3: Quadratur (3 Punkte), (jetzt auch Bonus!)

In dieser Aufgabe sollen Sie das Konvergenzverhalten von Quadraturverfahren für Funktionen mit verschiedenen Eigenschaften, wie zum Beispiel stetig und nicht-stetig differenzierbar, untersuchen. Gegeben sind dazu die folgenden fünf Funktionen:



Basierend auf dem vorgegebenen Code in `quadratur.py` sollen die Trapez- und Simpsonregel implementiert werden. Anschließend sollen die Integrale der fünf Testfunktionen für 10, 50, 100, 500, 1000 und 10000 Stützstellen berechnet werden und Grafiken mit dem Konvergenzverhalten erzeugt werden.

- Implementieren Sie die Trapezregel in der Funktion `quadTrapez()`.
- Implementieren Sie die Simpsonregel in der Funktion `quadSimpson()`.
- Verwenden Sie die Funktion `plotErrors()` um die Fehler bei der Integration in Abhängigkeit von der Anzahl der Stützstellen darzustellen (für die konstante Funktion soll keine Grafik erzeugt werden). Zum Beispiel, für die Gauss-Funktion erhalten wir folgendes Konvergenzverhalten:



Die Funktion `plotErrors()` erzeugt Grafiken im Verzeichnis `./figures/`, welche mit dem Code einzureichen sind.

Aufgabe 4: Bonus Aufgabe (3 Bonus Punkte)

Erstellen Sie eine aus mindestens sechs Keyframes bestehende Animation, welche eine komplexe Bewegung des Strichmännchens aus Aufgabe 2 beschreibt. Die Bewegung kann z.B. ein gut gemachter Laufzyklus, ein Sprung, oder ähnliches sein. Die Besten Animationen werden in der Vorlesung vorgestellt.