Computational Physics — SoSe 2022
5. Übungsblatt

Ausgabe: 12.05.2022 — Prof. H. Jelger Risselada
Abgabe: 19.05.2022 12 Uhr

Aufgabe 0: Verständnisfragen — 0 Punkte

1) Welche Verfahren zum Lösen einer zeitunabhängigen partiellen Differentialgleichung kennen Sie? Welche Vorteile bieten diese jeweils?

2) Wie unterscheidet sich das FTCS-Schema von dem Crank-Nicolson-Schema? Was müsste man bei einer Implementierung des Crank-Nicolson-Schemas zusätzlich beachten?

Aufgabe 1: Poisson-Gleichung — 12 Punkte

Sie haben in einer geeigneten Elektrodynamik-Vorlesung bereits die Poisson-Gleichung kennengelernt, mit Hilfe derer ein elektrostatisches Potential aus einer statischen Ladungsdichte berechnet werden kann. Bis auf wenige Ausnahmefälle kann diese Gleichung nicht analytisch gelöst werden. Deshalb sollen Sie in dieser Aufgabe ein Verfahren implementieren, um die Poisson-Gleichung für beliebige Ladungsdichten numerisch zu lösen.

Lösen Sie die zweidimensionale Poisson-Gleichung ($\epsilon_0 = 1$)

$$\left(\partial_x^2 + \partial_y^2\right) \phi(x, y) = -\rho(x, y)$$

mit Hilfe der Gauß-Seidel-Iteration für folgendes System: Wir betrachten ein Quadrat mit den Abmaßen $Q = [0, 1]^2$ und Dirichlet-Randbedingungen mit vorgegebenem Potential ϕ. Im Inneren des Quadrates befinden sich diskrete Ladungen q_i an den Orten $\vec{r}_i$, sodass für die Ladungsdichte gilt

$$\rho(\vec{r}) = \sum_i q_i \delta(\vec{r} - \vec{r}_i)\,.$$

Verwenden Sie in der gesamten Aufgabe eine Diskretisierung mit $\Delta = 0.05$. Nutzen Sie als Abbruchkriterium für die Gauß-Seidel-Integration eine Fehlerschranke von $\kappa = 10^{-5}$. Stellen Sie in jedem Aufgabenteil Ihre Ergebnisse graphisch dar.

Gehen Sie bei Ihrer Implementierung wie folgt vor:

a) Diskretisieren Sie das System und implementieren Sie die Gauß-Seidel-Iteration. Bei jeder Iteration soll der Algorithmus einmal jeden Gitterplatz im Inneren aktualisieren, ohne dabei die Ränder zu verändern. Schreiben Sie außerdem eine Ausgaberoutine für $\phi(\vec{r})$ und das elektrische Feld $\vec{E}(\vec{r}) = -\nabla\phi(\vec{r})$. Wählen Sie als Anfangsbedingungen $\phi(x, y) = 1$ im Inneren und testen Sie den Algorithmus für $\rho = 0$ (keine Quellen) für die Dirichlet-Randbedingung $\phi = \text{const} = 0$.

b) Lösen Sie die Poisson-Gleichung für $\rho(x, y) = 0$ im Inneren und mit den Randbedingungen

$$\begin{aligned}&\phi(x = 0, y) = \phi(x = 1, y) = \phi(x, y = 0) = 0,\\ &\phi(x, y = 1) = 1.\end{aligned}$$

Vergleichen Sie das analytische Ergebnis mit Ihrem numerischen Resultat. Die analytische Lösung für $\phi(x, y)$ lautet

$$\Phi(x, y) = \sum_{n=1}^{\infty} \frac{2(1 - \cos(n\pi))}{n\pi \sinh(n\pi)} \sin(n\pi x) \sinh(n\pi y).$$

Für engagierte Studierende: Leiten Sie die analytische Lösung über eine Fourier-Zerlegung oder einen Separationsansatz her.

c) Wählen Sie wieder $\phi = \text{const} = 0$ auf allen Rändern und setzen Sie nun eine Ladung $q_1 = +1$ in die Mitte von Q. Berechnen Sie $\phi(\vec{r})$ und $\vec{E}(\vec{r})$ im Inneren durch Iteration.

d) Wählen Sie auch hier wieder $\phi = \text{const} = 0$ und setzen Sie nun zwei positive Ladungen mit $q = +1$ auf die Punkte $\vec{r}_1 = (0.25, 0.25)^\top$ und $\vec{r}_2 = (0.75, 0.75)^\top$ und zwei negative Ladungen mit $q = -1$ auf die Punkte $\vec{r}_3 = (0.25, 0.75)^\top$ und $\vec{r}_4 = (0.75, 0.25)^\top$. Berechnen Sie auch hier $\phi(\vec{r})$ und $\vec{E}(\vec{r})$ im Inneren durch Iteration.

Aufgabe 2: Diffusionsgleichung — 8 Punkte

Während Sie sich in der letzten Aufgabe mit der Diskretisierung eines statischen Problem auseinandergesetzt haben, sollen Sie sich nun ein Problem mit zusätzlicher Zeitabhängigkeit anschauen. Zur numerischen Lösung der Diffusionsgleichung muss neben den räumlichen Freiheitsgraden auch die Zeit diskretisiert werden. In dieser Aufgabe sollen Sie eines der einfachsten Verfahren dazu betrachten, das sogenannte FTCS (forward in time, centered in space) Schema.

Implementieren Sie zum Lösen der Diffusionsgleichung

$$\partial_t u(x,t) = D\, \partial_x^2 u(x,t)$$

das explizite FTCS-Schema. Wir betrachten ein eindimensionales System auf dem endlichen Stück $L = [0,1]$ mit einer räumlichen Diskretisierung $\Delta x = 0.01$ in der gesamten Aufgabe. Die Ränder seien dabei "isolierend", sodass hier keine Konzentration $u(x,t)$ von Außen zu- bzw. nach Außen abfließen kann. Setzen Sie außerdem in der gesamten Aufgabe die Diffusionskonstante auf $D = 1$. Stellen Sie alle Ihre Ergebnisse graphisch dar, zum Beispiel mit Hilfe einer Animation mit *FuncAnimation* aus *matplotlib.animation*.

a) Schreiben Sie einen Algorithmus, der auf L eine beliebige Anfangsbedingung $u(x,0)$ erhält und mit einer Zeitdiskretisierung Δt das System propagiert. Beachten Sie dabei auch die Randbedingungen. Prüfen Sie Ihre Implementierung für ein System mit der Anfangsbedingung

$$u(x,0) = \text{const} = 1.$$

b) In der Vorlesung wurde für das FTCS-Schema das Stabilitätskriterium

$$\frac{2D\Delta t}{\Delta x^2} < 1$$

hergeleitet. Überprüfen Sie dies für die Anfangsbedingung (δ-Peak)

$$u(x,0) = \delta(x - 0.5),$$

indem Sie Ergebnisse für eine Zeitdiskretisierung knapp über und knapp unter dem Kriterium miteinander vergleichen.

c) Wählen Sie nun eine zuverlässige Schrittweite und bestimmen Sie die zeitabhängige Lösung der Diffusionsgleichung für die Anfangsbedingungen (δ-Peak, Heaviside, Dirac-Kamm)

$$\begin{aligned} u_1(x,0) &= \delta(x - 0.5), \\ u_2(x,0) &= \theta(x - 0.5), \\ u_3(x,0) &= \frac{1}{9}\sum_{n=1}^{9} \delta(x - 0.1n). \end{aligned}$$

Für welche Anfangsbedingung nähert sich das System am schnellsten seinem Gleichgewichtszustand an? Wie verhält sich das Raumintegral der Konzentration $u(x,t)$ über die Zeit?