

Blatt 5

Hinweise: Sie können Ihre Lösungen mit Kommentaren und/oder weiteren Fragen innerhalb von zwei Wochen nach Ausgabe der Aufgaben per Email an Ihren Übungsleiter schicken.

Jens Harting: j.harting@ica1.uni-stuttgart.de

Aufgabe 11. Diffusionsbeschränktes Wachstum

Entwickeln und implementieren Sie einen Algorithmus zur Simulation von DLA (*diffusion limited aggregation*). Wir stellen uns etwa vor, dass Atome unter thermischem Einfluß auf einer Kristalloberfläche umherwandern können. Diese Wanderung modellieren wir durch einen *random walk*. Ein Atom (*random walker*) kann sich jetzt an einen Wachstumskeim anlagern, d.h. einen der Nachbarplätze (*growth sites*) eines bereits besetzten Gitterplatzes belegen, der zu Programmbeginn im Zentrum eines Gitters angelgt wurde. Dadurch kann z.B. zweidimensionales Kristallwachstum simuliert werden.

Da ein aus dem Unendlichen kommendes Teilchen einen Kreis mit minimalem Radius um diesen 'Kristall' an einer zufälligen Stelle durchstößt, beginnen wir die *random walks* direkt auf diesem kleinstmöglichen Kreis. Es ist jedoch möglich, dass sich der *walker* wieder vom Aggregat entfernt und es kann unter Umständen lange dauern kann, bis er wieder dorthin zurückkehrt. Daher wird er verworfen und neu gestartet, sobald er sich zu weit vom wachsenden Aggregat entfernt hat (wählen Sie als Grenze den doppelten Aggregatradius). Stellen Sie einige Aggregate graphisch dar.

Aufgabe 12. Fraktale Dimension

Erweitern Sie Ihr Programm für die Berechnung der Ausdehnung des Objekts, gegeben durch den Gyrationradius

$$R_g = \sqrt{\frac{1}{N} \sum_{i=1}^N (\vec{r}_i - \langle \vec{r} \rangle_N)^2}$$

und geben Sie in jedem Wachstumsschritt die Masse M (= Teilchenzahl N) und den Radius R_g in eine Datei aus. Tragen Sie in doppelt-logarithmischer Darstellung die Masse des wachsenden Objektes als Funktion des Radius auf. Bestimmen Sie dann die fraktale Dimension d_f als asymptotische Steigung (in logarithmischer Darstellung $\log M$ über $\log R_g$) für ausreichend große Radien: $M \sim R_g^{d_f}$

Aufgabe 13. Zellularautomaten

(a) Schreiben Sie ein Programm für einen eindimensionalen Zellularautomaten. Der einfachste Weg alle Zellen gleichzeitig zu aktualisieren ist, die Zustände der Zellen zum Zeitpunkt t und $t+1$ in zwei separaten Feldern zu speichern. Implementieren Sie die „Regel 90“, d.h. der Zustand einer Zelle zur Zeit $t+1$ ist die Summe modulo 2 der Zustände der beiden Nachbarzellen zur Zeit t . Als Anfangszustand setzen Sie eine Zelle auf den Wert 1 und alle anderen auf 0. Es genügt die Untersuchung der Zeitentwicklung für etwa 20 Zeitschritte. Zeigen die entstehenden Strukturen Selbstähnlichkeit? Können Sie eine fraktale Dimension zuordnen?

(b) Untersuchen Sie die Eigenschaften der „Regel 150“, d.h. der Zustand einer Zelle zur Zeit $t+1$ ist die Summe modulo 2 der Zustände der beiden Nachbarzellen plus der Zustand der Zelle selbst zur Zeit t .

(c) Wählen Sie nun zufällige Anfangszustände, wobei Sie die Zellen zur Zeit $t = 0$ mit Wahrscheinlichkeit $p = \frac{1}{2}$ besetzen. Untersuchen Sie die Zeitentwicklung der Zustände bei Anwendung der Regeln 18, 73, 90, 136 und 150. Es genügt, etwa 80 Zellen (mit periodischen Randbedingungen) über 60 Zeitschritte zu verfolgen. Wie sensitiv hängen die beobachteten Strukturen von Änderungen des Anfangszustands ab? Klassifizieren Sie qualitativ die Zeitentwicklung der Zellen bei Anwendung der verschiedenen Regeln durch das Auftreten (i) eines konstanten homogenen Zustands, (ii) separater periodischer Strukturen, (iii) chaotischer, aperiodischer Strukturen oder (iv) komplexer, lokalisierter Strukturen.