

Computerpraktikum
zur Vorlesung <Theoretische Physik III>
im Wintersemester 2006/2007
*Fraktales Wachstum –
Diffusion-Limited Aggregation*

Dozent: Prof. Dr. LUIS SANTOS · Betreuer: MARTIN PAECH

Ausgabedatum: 6. November 2006

Abgabedatum: 26. Januar 2007

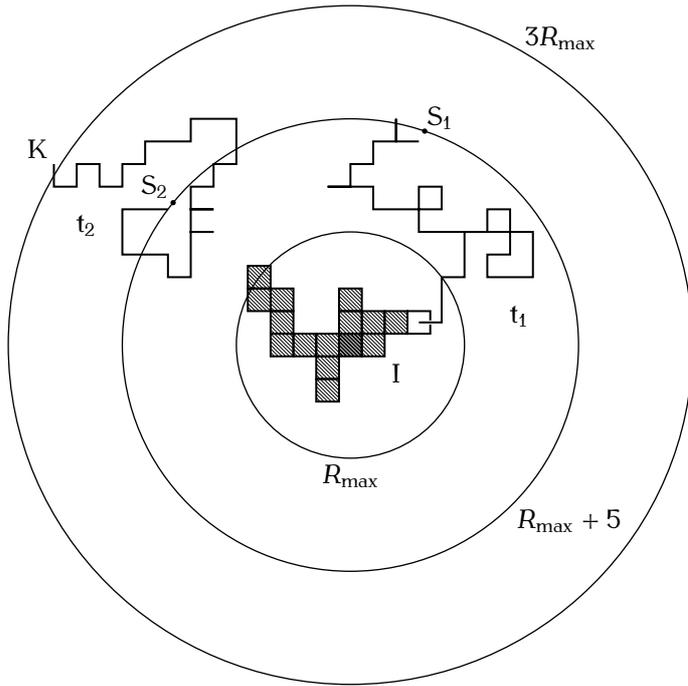
Grundlagen

Viele makroskopische Objekte in der Natur bilden sich durch die Vereinigung (Aggregation) kleinerer Subeinheiten. Beispiele dafür sind Schneeflocken, Blitze und das Wachstum von Bakterienkolonien. Unter Nichtgleichgewichtsbedingungen führen solche Wachstumsprozesse häufig zu komplexen *fraktalen* Strukturen.

Die Suche nach einem besseren Verständnis dieser Phänomene, welche außer ihrer Struktur wenig gemeinsam haben, wird seit Beginn der achtziger Jahre mit der Computersimulation von Wachstumsmodellen betrieben. Eines der erfolgreichsten Modelle mit vielen Anwendungen von der Biologie bis zur Astrophysik ist die diffusionskontrollierte Aggregation, DLA (*diffusion-limited aggregation*), die speziell fraktales Wachstum beschreiben kann. Dabei läßt man Teilchen sukzessive in einem vorgegebenen Gebiet ziellos umherwandern (*random walk*), bis sie auf die Oberfläche eines aus einem Keim (*seed particle*) gewachsenen Objektes treffen und sich dort anlagern. Die namensgebende Veröffentlichung "*Diffusion-Limited Aggregation, a Kinetic Critical Phenomenon*" von T. A. WITTEN und L. M. SANDER, *Physical Review Letters* 47 (19), 1400 (1981), gehört zu den zehn meistzitierten Arbeiten dieser einflußreichen Zeitschrift (siehe auch <http://www.aps.org/apsnews/topten.cfm>).

Der DLA-Algorithmus

Der Modellraum zum Wachsen eines DLA-Clusters bestehe aus einem zweidimensionalen quadratischen Gitter mit einem einzigen besetzten Gitterplatz in der Mitte als Wachstumskeim. Ein Aggregatteilchen wird in *weitem* Abstand davon losgelassen und beginnt einen *random walk*, der es mit gleicher Wahrscheinlichkeit zu den jeweils vier nächsten Nachbarplätzen führt. Ist ein Nachbarplatz schon von einem anderen Teilchen besetzt, so beendet es seinen Irrweg, und sein aktueller Platz wird als zum Cluster gehörend markiert. Dieser Prozess wird viele Male wiederholt, bis sich ein großer Cluster mit M Teilchen gebildet hat. Wegen der möglichen langen Irrwege der Teilchen bei der Simulation größerer DLA-Cluster wird der Algorithmus weiter präzisiert (siehe Abbildung): Ein neues Teilchen startet von einem zufällig gewählten Punkt auf einem Kreis mit dem Radius $R_{\max} + 5$ um den Ursprung (Keim), wobei R_{\max} der gerade aktuelle maximale Clusterradius ist, d. h. der maximale Abstand der Clusterteilchen vom Ursprung. Sollte sich das Teilchen auf seinem Irrweg allzu weit vom Cluster entfernt haben, so wird es bei einem Abstand von $r > 3R_{\max}$ aus dem



Schematische Darstellung der diffusionskontrollierten Aggregation auf einem quadratischen Gitter: Der Keim des Aggregates I ist dunkler, die bereits in den Cluster eingeschlossenen Gitterplätze sind heller schattiert.

Beginnend von einer zufälligen Position auf dem Rand des Startkreises (Radius $R_{\max} + 5$) sind zudem zwei typische Diffusionswege eingezeichnet: Das Teilchen auf der Trajektorie t_1 startet am Punkt S_1 , erreicht möglicherweise einen unbesetzten Gitterplatz an der Cluster-Oberfläche (unausgefüllt gezeichnet) und besetzt diesen dauerhaft. Der Diffusionsweg t_2 führt vom Cluster weg; erreicht ein Teilchen auf dieser Bahn den Rand des äußeren Kreises (Radius $3R_{\max}$), Punkt K, so wird es dem Prozeß entnommen und beginnt seinen Weg erneut auf einem zufällig auf dem Rand des Startkreises gewählten Punkt.

Prozeß herausgenommen und ein neues Teilchen vom Startkreis losgeschickt. (Die genannten Zahlen sind als Richtwerte anzusehen.) Für die Simulation sehr großer Cluster existieren weitere Beschleunigungsmethoden, z. B. die zeitweilige Vergrößerung der effektiven Schrittweite.

Die konkrete Computersimulation findet auf einem Gitter mit $L \times L$ Plätzen statt. Im Prinzip reichen zwei eindimensionale Felder zur Aufnahme der x - und y -Koordinaten der N Gitterplätze aus. Bei Clustern mit $M \gtrsim 1000$ Teilchen wird die Simulation wegen der Abfrage der beiden wachsenden Listen aber extrem langsam, verglichen mit der speicheraufwendigen, indes schnellen Simulation, bei welcher ein zweidimensionales Feld der Größe $L \times L$ mitgeführt wird. Dadurch hingegen wird die Abfrage der jeweiligen Nachbarplätze auch bei großem N nicht zeitaufwendiger. Die Wahl des Startpunktes und der vier Bewegungsrichtungen geschieht mit einem Zufallszahlengenerator, den heutzutage jede (universelle) Programmiersprache aufweist.

Die fraktale Dimension

Die löchrige Struktur der DLA-Aggregate läßt vermuten, daß es sich um Gebilde mit gebrochener, also *fraktaler* Dimension handelt. Eine fraktale Dimension D wird üblicherweise anhand einer Skalierungsrelation zwischen der Masse (momentane Anzahl $N \in [1, M]$ der Teilchen des wachsenden Clusters) und einer charakteristischen Länge l im Aggregat definiert: $N \sim l^D$ für $N \gg 1$. Für ein kompakt wachsendes N -Teilchenaggregat mit dem Clusterradius R erhielte man das Potenzgesetz $N \sim R^2$ mit $D = d = 2$, wobei d die normale Dimension des Clusters ist. Die fraktale Struktur der DLA-Cluster führt dagegen zu einer fraktalen Dimension $1 < D < 2$.

Es gibt verschiedene Verfahren zur Messung bzw. Abschätzung der fraktalen Dimension. Das wohl einfachste besteht in der Messung des bei der Simulation ohnehin verwendeten maximalen Clusterradius R_{\max} als Funktion der Clustermasse N , $1 \ll N \leq M$, und folgender Auswertung des vermuteten Potenzgesetzes $R_{\max} \sim N^\gamma$ mit $D_\gamma = 1/\gamma$.

Eine in der Literatur häufig verwendete charakteristische Länge ist der *radius of gyration* R_g definiert für einen Cluster aus N Teilchen durch

$$R_g^2 = \frac{1}{N} \sum_{i=1}^N |\vec{r}_i - \vec{r}_{\text{cm}}|^2, \quad \text{mit} \quad \vec{r}_{\text{cm}} = \frac{1}{N} \sum_{i=1}^N \vec{r}_i$$

und dem Ort \vec{r}_i des i ten Teilchens im Cluster. Hier ist $R_g \sim N^\beta$ für $1 \ll N \leq M$ mit $D_\beta = 1/\beta$ auszuwerten.

Ein ganz anderes Verfahren macht Gebrauch von der Eigenschaft des sogenannten *Korrelationsintegrals*

$$C(r) = \lim_{M \rightarrow \infty} \frac{1}{M^2} \sum_{i,j=1}^M \Theta(r - |\vec{r}_i - \vec{r}_j|),$$

das sich für kleine Abstände r (größer als ein paar Gittereinheiten, aber deutlich kleiner als der Clusterradius) gemäß $C(r) \sim r^\nu$ verhält. Der Exponent ν wird als Korrelationsdimension D_c bezeichnet und nach $D_c = \lim_{r \rightarrow 0} \ln C(r) / \ln r$ bestimmt.

$C(r)$ ist die Abstandsverteilungsdichte (für eine prinzipiell unendlich ausgedehnte) Struktur, d. h. die Anzahl aller Abstände $|\vec{r}_i - \vec{r}_j|$, die kleiner als ein vorgegebener Abstand r sind. Der Vorfaktor dient nur zur Normierung von $C(r)$ auf Werte zwischen $1/M$ und 1 . Das Korrelationsintegral $C(r)$ ist im dreidimensionalen Raum definiert durch

$$C(r) = \int_0^r d^3 r' g(\vec{r}'), \text{ wobei } g(\vec{r}') = \frac{1}{M^2} \int d^3 r'' n(\vec{r}'') n(\vec{r} + \vec{r}')$$

die Dichte-Dichte-Korrelationsfunktion ist. Mit der Dichte $n(\vec{r}) = \sum_{i=1}^M \delta(\vec{r} - \vec{r}_i)$ ergibt sich die obige auswertbare Form des Korrelationsintegrals.

Aufgaben

1. Programmieren Sie den Algorithmus zur Simulation von DLA-Clustern, und lassen Sie sich damit im Computereperiment etwa fünf bis zehn Cluster mit $M \gtrsim 10000 \dots 15000$ angelagerten Aggregatteilchen wachsen. Die großen Cluster benötigen zum Wachsen bereits Gitter mit $L \approx 400$. Denken Sie auch daran, die vom Zufallszahlengenerator erzeugte (deterministische!) Folge von Zufallszahlen für jeden neuen Cluster mit einer anderen *seed* zu starten. Erstellen Sie von ein paar Clustern Graphiken, und speichern Sie am besten die relevanten Daten eines jeden Clusters zur späteren Auswertung ab.
2. Bestimmen Sie mit jedem der angegebenen Verfahren die fraktale Dimension D aller von Ihnen erzeugten DLA-Cluster, und berechnen Sie zusätzlich den Mittelwert von D (mit der zugehörigen Schwankung) für jedes Verfahren. Die gesuchten Exponenten β , γ und ν lassen sich in der doppellogarithmischen Darstellung des betreffenden Potenzgesetzes einfach aus der Ausgleichsgeraden (*least-square fit*) durch die \langle Meßpunkte \rangle gewinnen. Für die Bestimmung des *radius of gyration*-Exponenten β sollten Sie die letzten 80% der Teilchen heranziehen, welche dem Cluster während seines Wachstums hinzugefügt worden sind.

Literaturhinweise

- [1] W. KINZEL und G. REENTS: *Physik per Computer. Programmierung physikalischer Probleme mit Mathematica und C*. Heidelberg: Spektrum Akademischer Verlag, 1996.
- [2] W. H. PRESS, S. A. TEUKOLSKY, W. T. VETTERLING und B. P. FLANNERY: *Numerical Recipes in C. The Art of Scientific Computing*. Cambridge: Cambridge University Press, 1992. Siehe <http://www.numerical-recipes.com>.
- [3] K. BINDER und D. W. HEERMANN: *Monte Carlo Simulation in Statistical Physics*. Berlin: Springer, 1988 (4. Auflage 2002).

Anmerkungen zum Computerpraktikum

- ▷ Die erfolgreiche Teilnahme am Computerpraktikum ist eine Voraussetzung für die Vergabe des Übungsscheines der entsprechenden Theorievorlesung (Pflichtleistung).
- ▷ Das Computerpraktikum ist eine Computer-Hausübung. Die Aufgabe besteht darin, ein physikalisches Problem mit Hilfe des Computers zu lösen und einen Bericht auszuarbeiten.
- ▷ Zum Bericht gehören eine Zusammenfassung des Problems, eine Beschreibung der angewandten Methoden und des Programmes, einen kommentierten Code (Programmlisting), und eine Auswertung der Ergebnisse mit Graphiken und Fehlerrechnung.
- ▷ Die Durchführung der Übung und die Ausarbeitung des Berichts können durch jede(n) Studierende(n) einzeln oder in Zweiergruppen erfolgen.
- ▷ Die Übung gilt als erfolgreich bearbeitet, wenn die Ergebnisse korrekt sind und der Bericht akzeptiert worden ist. Die Studierenden müssen auch in der Lage sein, ihr Programm an einem der CIP-Rechner oder an einem mitgebrachten Laptop zu demonstrieren.
- ▷ Die endgültige Version des Berichtes muß unbedingt bis zum Ende der Bearbeitungszeit abgegeben werden.
- ▷ Als Programmiersprachen sind die für die Forschung bedeutsamsten, FORTRAN (77 und 90/95) sowie C und C++, dringend empfohlen; andere wie PASCAL/DELPHI oder JAVA nur nach Rücksprache mit dem Betreuer. Sie dürfen auch mit MATHEMATICA arbeiten, müssen aber die numerischen Verfahren selber programmieren und dürfen nicht auf die enthaltenen Funktionen und Algorithmen zurückgreifen. Zur graphischen Auswertung der Programmausgaben können Sie externe Visualisierungsanwendungen, beispielsweise GNUPLOT, verwenden.
- ▷ *In Ihrem eigenen Interesse* wenden Sie sich mit Fragen bitte *möglichst frühzeitig* an Ihren Betreuer: Neben den offiziellen Sprechzeiten, Mittwochs zwischen 15 und 17 Uhr im Raum 223, Appelstraße 2, sowie Donnerstags von 15 bis 17 Uhr im CIP-Pool, Raum 034, gerne auch nach Vereinbarung (Telefon: 762 32 68, eMail: martin.paech@itp.uni-hannover.de) – oder auf <gut Glück> – im Zimmer 223.