

Projekt: Diffusionsbeschränktes Wachstum (Fraktale)

Physikalische Fragestellung

Das diffusionsbegrenzte Anlagern beschreibt Phänomene wie Lichtenberg-Figuren oder die Anlagerung von Farbpigmenten. Diese Prozesse und Phänomene sind eng mit Fraktalen verbunden.

Das diffusionsbegrenzte Wachstum geht von einem Keim aus. Aus dem Unendlichen kommen nun Teilchen diffundiert. Kommt ein solches Teilchen zu einem nächsten Nachbarplatz eines bereits belegten Platzes, dann bleibt es dort haften. Aus einem solchen Keim wachsen nun fraktale Strukturen. Selbstverständlich kann man dies auch für räumlich kontinuierliche Systeme formulieren, jedoch ist es hier einfacher, mit Gittersystemen zu arbeiten.

Die physikalische Frage ist die nach den fraktalen Eigenschaften solcher Strukturen. Man möchte die fraktale Dimension einer Struktur verstehen, die multifraktalen Eigenschaften untersuchen. Gegenüber den deterministischen Fraktalen, z.B. Sierpinski Gitter, Koch-Kurve etc., kommt hier die Zufälligkeit mit ins Spiel.

Methode

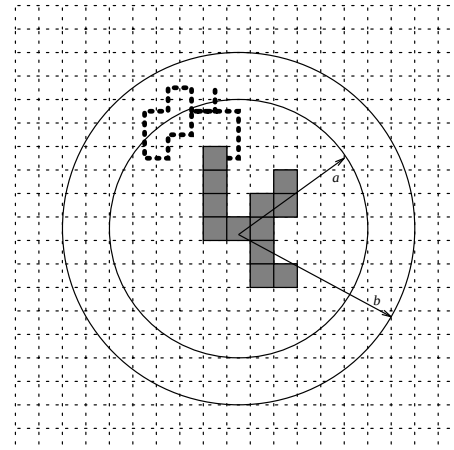
Die Methode, um das diffusionsbegrenzte Wachstum zu simulieren und zu studieren, ist die Monte-Carlo-Methode. Hier tritt sie in der einfachsten Weise auf. Jede Konfiguration eines DLA-Clusters wird unabhängig voneinander erzeugt (Simple Sampling-Verfahren). Zur Erzeugung eines Clusters benutzt man Zufallszahlen, die einen Random Walk (Irrfahrt) generieren. Man erhält dann eine Realisierung eines Clusters aus dem Raum der möglichen. Es tritt keinerlei Wichtung der erzeugten Konfigurationen auf. Jede hat das gleiche statistische Gewicht, wenn man Observablen für einen Cluster, z.B. seinen Radius, seine Masse, berechnet.

Algorithmus

Die Erzeugung eines Clusters geschieht durch Random Walks. Sie werden „weit“ von dem bereits erzeugten Cluster aufgesetzt. Dies soll das Kommen aus dem Unendlichen simulieren. Wie weit „weit“ ist, hängt im wesentlichen von der verfügbaren Rechenzeit ab. Je weiter der Aufsetzpunkt von dem nächstliegenden besetzten Gitterplatz entfernt ist, desto länger

muss man warten, bis der Random Walker zu einem Platz gelangt, an dem er haften bleibt.

Um weitere Rechenzeit einzusparen, werden Walks entfernt, die sich von dem Objekt entfernt haben. Es werden also isotrope Aufsetzpunkte und Entfernepunkte auf dem Gitter definiert, die sich mit Anwachsen der Struktur verändern müssen. Dies ist in der nachfolgenden Figur verdeutlicht:



Die von dem Prozess erzeugte Struktur muss in einer geeigneten Datenstruktur abgelegt sein. Dies wird günstigerweise als eine verkettete Liste geführt, da man einerseits alle besetzten Plätze besuchen und andererseits auch Nachbarschaftsbeziehungen erkennen muss. Auch hier gilt, wie für den Leath-Algorithmus, dass durch die fraktale Eigenschaft des Objekts für große Objekte ein Gitter als Datenstruktur nicht vernünftig ist. Die Anzahl der unbesetzt bleibenden Plätze wird mit ansteigender linearer Dimension des Objekts um ein Vielfaches größer als das Objekt selbst.

DLA Algorithmus

Starte mit einem besetzten Platz.
Wähle einen Punkt auf einem Kreis um den Schwerpunkt des Clusters mit dem Radius R_1 .
Starte einen Random Walker von diesem Punkt.
Wenn der Random Walker sich außerhalb eines Kreises um den Schwerpunkt des Clusters mit dem Radius $R_2 > R_1$ befindet, dann gehe zurück zur Auswahl eines Aufsetzpunktes.
Wenn der Random Walker einen Gitterplatz erreicht, dessen nächster Nachbar besetzt ist, dann bleibt er haften.
Berechne den minimalen Abstand d zum Kreis R_1 .

Projekt: Diffusionsbeschränktes Wachstum (Fraktale)

Wenn $d <$ als ein Schwellwert, dann erhöhe $R1$ und $R2$.
Gehe zurück zum Aufsetzen.

Implementierung

Zur Implementierung des obigen Algorithmus ist es notwendig, zunächst eine geeignete Datenstruktur zu entwickeln. Sie muss in der Lage sein, ein schnelles Suchen von möglichen besetzten Nachbarn zu erlauben. Es kommen in Frage:

- eine mehrfach verkettete Liste oder
- eine Hashtabelle.

Sie können sich für die von Ihnen als günstigste wahrgenommene Möglichkeit entscheiden. Bedenken Sie zum einen den Speicherbedarf und zum anderen die Zugriffszeit bis Sie entscheiden können, ob ein Random Walker haften bleibt oder nicht. Dazu müssen Sie in der Liste suchen.

Den Random Walker verwirklichen Sie wie folgt. Sie starten auf einem Platz. Da der Walker ein Random Walker ist, „vergisst“ er sofort, wo er hergekommen ist. Nehmen wir an, alles spiele sich auf einem einfach quadratischen Gitter ab. Jeder Gitterpunkt hat vier nächste Nachbarn. Jeder dieser Gitterpunkte hat die Wahrscheinlichkeit $\frac{1}{4}$, dass er beim nächsten Zug von dem Walker besetzt wird (Diffusion!). Er kann also sofort wieder auf die Stelle zurückkehren, von der er gekommen ist.

Sie ziehen eine Zufallszahl r , die aus dem Intervall $(0,1)$ ist. Diese reelle Zahl r wandeln Sie in eine natürliche Zahl aus der Menge $\{0,1,2,3\}$ um. Sie bilden also das Intervall $(0,1)$ auf eine diskrete Menge ab. Tip: Umwandlung `float` in `int`, sowie `floor` Funktionen.

Aufgabe

1. Entwickeln Sie eine geeignete Datenstruktur für das DLA-Problem
2. Schreiben Sie ein Programm zur Simulation eines DLA-Clusters

Literatur

1. T.A. Witten und L.M. Sander, Diffusion-limited aggregation, *Physical Review B* 27, 5686, 1983

2. H.E. Standley, *Kinetics of Aggregation and Gelation*, eds F. Family and D. Landau, North Holland, Amsterdam, 1984
3. D.W. Heermann, *Computer Simulation Methods of Theoretical Physics*, Springer Verlag, 2nd Edition, 1992
4. K. Binder und D.W. Heermann, *Introduction to the Monte Carlo Method*, Springer Verlag, 4th Edition 2002