

Ein ZA-basiertes Modell zur Simulation der single-file-Diffusion mittels MPI in Cluster-Rechnern

U. Renner, Arnold-Sommerfeld-Gesellschaft Leipzig, Email: info@renner-uwe.de
D. Fey, Universität Jena, Institut für Informatik, Email: dietmar.fey@uni-jena.de

1. Problemstellung

In den hier vorgestellten Untersuchungen soll ein spezielles physikalisches Problem, das der single-file-Diffusion (SFD) [1]-[3] behandelt werden. Hierbei handelt es sich um ein besonderes eindimensionales Diffusionsproblem wie es in sehr engen Kanälen, so z.B. in heterogenen Katalysatoren etwa Zeolythen und anderen porösen Materialien, auftritt. Das Verhalten hierin unterscheidet sich qualitativ vom herkömmlichen eindimensionalen Diffusionsproblem, da sich die Teilchen aufgrund der geometrischen Einschränkung nicht überholen können. Es treten Staus auf, wodurch die Ausbreitung stark behindert wird und was in der geänderten Gesetzmäßigkeit für das mittlere Verschiebungsquadrat zum Ausdruck kommt.

Dieses ungewöhnliche physikalische Verhalten lässt sich durch zelluläre Automaten (ZA) [4]-[6] geeignet beschreiben. Wegen ihrer theoretisch massiv-parallelen Struktur sind solche Modellsysteme ideal geeignet für Computer-Experimente auf der Basis von Simulationsstudien. So kann einerseits jede Automatenzelle durch einen Prozessor realisiert werden, andererseits lässt sich der gesamte ZA auch auf einen einzigen Prozessor sequentiell simulieren. Die Verwendung zellulärer Automaten für dieses spezielle physikalische Problem zeigt eindrucksvoll ihre qualitative und quantitative Stärke bei der Modellierung realer Prozesse, so der single-file-Diffusion.

Für eine realistische statistische Auswertung muss man mit einer Teilchenzahl in der Größenordnung von mehreren Millionen Teilchen rechnen. Um solch groß dimensionierte Simulationssysteme in akzeptablen Rechenzeiten simulieren zu können, bieten sich Cluster-Rechner bzw. Multi-Cluster-Rechner an. Hierbei übernimmt jeder Rechnerknoten einen bestimmten Abschnitt des gesamten zellulären Systems für die Simulation.

Da die hier vorgestellten Automatenregeln sehr einfach sind und die Verarbeitungszeit pro Zelle sehr gering ist, wird man bei sehr großer Clustergröße eine Abnahme des Speed-Up aufgrund der Kommunikation für den Austausch der Randzustände erwarten. Als Programmierumgebung für einen Cluster-Rechner wurde das Message Passing Interface (MPI) [7], [8] von LAM verwendet. Es wurden verschiedene Konfigurationen für die single-file-Diffusion untersucht. Die erzielten Rechenzeiten liegen aufgrund des parallelen Rechnens in Cluster-Strukturen im Bereich weniger Minuten. Es stellte sich heraus, dass eine Multi-Cluster-Umgebung, wie ursprünglich geplant, nicht unbedingt notwendig ist. Die erzielten Ergebnisse der Simulationsexperimente bestätigen das theoretisch erwartete Verhalten.

2. Zur Theorie der single-file-Diffusion

Betrachtet man die gewöhnliche eindimensionale Ausbreitung von Teilchen, so macht man dies unter der Idealisierung, dass die anderen Dimensionen des Raumes für die quantitative Beschreibung der Ausbreitung vernachlässigbar sind, so im Modell des schmalen Stabes. Bei der herkömmlichen Beschreibung der Diffusion, kann man zumindest bei niedrigen Teilchenkonzentrationen c , jedes Teilchen einzeln betrachten und dessen statistischen Eigenschaften weitestgehend unkorreliert von den anderen beschreiben. Die Ausbreitung folgt gemäß dem FICKSchen Diffusionsgesetz einer GAUß-Verteilung. Eine charakteristische Messgröße ist das mittlere Verschiebungsquadrat $\langle \vec{r}^2 \rangle$. Für die gewöhnliche Diffusion ist dieses proportional zur Zeit t :

$$\langle \vec{r}^2 \rangle = 2dDt, \quad (1)$$

mit der Diffusionskonstanten D in der EUKLIDischen Raumdimension d .

In einigen stark porösen Systemen, etwa Zeolythen oder Aktivkohlen, beobachtet man jedoch eine Abweichung von dieser linearen zeitlichen Proportionalität, was als Diffusionsanomalität bezeichnet wird. Zum einen wird hierfür die strukturelle Beschaffenheit (verzweigte Netzwerke, fraktale Geometrien) verantwortlich gemacht, zum anderen kann diese durch starke Korrelationen der Teilchen untereinander erklärt werden. Eine große Klasse solcher Systeme lässt sich dann durch gebrochenzahlige Potenzgesetze charakterisieren. Ihr zeitlich asymptotisches Verhalten kann wie folgt beschrieben werden [9]:

$$\langle \vec{r}^2 \rangle \propto t^\gamma \text{ mit } 0 < \gamma < 1. \quad (2)$$

Der Exponent γ im angegebenen Intervall beschreibt eine gebremste anomale Diffusion und ist eine wesentliche Kenngröße des Systems. Der Wert $\gamma = 1$ hingegen liefert den Fall der gewöhnlichen Diffusion.

Im Falle der SFD wird die Anomalität durch die geometrische Einschränkung hervorgerufen, so dass die Teilchen aufgrund der Engporigkeit nicht einander passieren können und somit in ihren freien Bewegungen behindert werden. Für große Zeiten ist das Verhalten theoretisch bekannt:

$$\langle \vec{r}^2 \rangle = 2 \frac{1-c}{c} a^2 \sqrt{\frac{\Gamma t}{\pi}} \text{ mit } 0 < c < 1. \quad (3)$$

Hierbei ist a die Gitterkonstante, gemessen in geeigneten Längeneinheiten, und Γ die Sprungrate, gemessen pro Zeiteinheit. Diese Gitterkonstante sollte so gewählt werden, dass die mikroskopische Struktur hinreichend gut wiedergespiegelt wird, z.B. in 100 nm. In den Simulationen legen wir a auf 1 Gittereinheit fest. Das spezifische Teilchenverhalten kommt in der Sprungrate zum Ausdruck und hängt von den Wechselwirkungen des Teilchens mit dem Gitter, der Temperatur und effektiv auch von der Wechselwirkung der kollidierenden Teilchen ab. Sie gibt an, wie häufig das Teilchen eine Gittereinheit (im Mittel) weit „springt“, z.B. 1 Sprung in 1 μ s. In den Simulationen werden wir annehmen, dass ein solcher Sprung pro Zeiteinheit (zu 1 μ s) stattfindet.

Die makroskopische Ausdehnung, die wir durch die Simulation erfassen können, wird neben den obigen Parametern durch die Anzahl der Gitterpunkte, z. B. 10^6 , und der Endzeit der Simulation festgelegt. Aufgrund der bei der Simulation zur Vereinfachung angenommenen periodischen Randbedingungen, sollte diese Zahl viel größer als die maximal zu erwartende zurückgelegte Distanz sein. Ansonsten sind Fehler in Form von Abweichungen vom asymptotischen Verlauf zu erwarten. In (3) erkennt man deutlich den anomalen Charakter im Zeitverhalten des mittleren Verschiebungsquadrates, welches nunmehr proportional zur Wurzel der Zeit ist, d. h. $\gamma = 1/2$, und somit gegenüber der gewöhnlichen Diffusion viel stärker gebremst wird.

Die Diffusion wird hier als stochastischer Prozess verstanden werden, bei dem die Teilchen ihre Position nach statistischen mikroskopischen Gesetzen ändern. Die zeitliche Abfolge der Platzwechsel auf einem diskretisierten Gitter kann dann z.B. gemäß einem POISSON-Prozess erfolgen, mit exponentieller Wartezeitdichte für einen Sprung. Der Einfachheit betrachten wir im folgenden ZA-Modell nur zeitsynchrone Änderungen mit konstanter mittlerer Wartezeit.

3. Ein ZA-Modell für die single-file-Diffusion

Zur Beschreibung der Diffusion wird ein anschauliches mikroskopisches Bild verwendet. In diesem kann man alle erlaubten Sprünge der Teilchen in einem Zustandsautomaten allein durch kombinatorische Regeln unter Berücksichtigung der topologischen Besonderheiten und unter Zuhilfenahme des Zufalls bei Konfliktsituationen erfassen. Der dargestellte Zustandsautomat beschreibt sowohl das Vorhandensein der Teilchen und dessen Bewegung als auch die unbesetzten Gitterplätze. Die Gesamtdynamik des Systems folgt aus der parallelen Anwendung der Regeln.

Als Geometrie für das Modell der single-file-Diffusion dient ein eindimensionales Gitter mit der einheitlichen Gitterkonstanten a mit möglichst vielen Gitterpunkten. Jedem Gitterpunkt entspricht eine Zelle im ZA. Es werden zyklische Randbedingungen gewählt. Jedoch sind auch Randbedingungen mit konstanten Zuflussraten oder mit Reflexion denkbar. Die Gitterpunkte werden zufällig und gleich verteilt entsprechend der Teilchenkonzentration c besetzt. Der Besetzungszustand der Gitterpunkte wird durch zwei Zustände ($A = \text{„leer“}$, $B = \text{„besetzt“}$) beschrieben. Die zeitliche Entwicklung des Gesamtsystems erfolgt im einheitlichen Zeittakt $\tau = 1/(2\Gamma)$.

Auch die Bewegungsrichtung („links“ und „rechts“) der Teilchen wird wie bei der freien Diffusion zufällig gewählt. Da wegen der Einfachheit keine weitere äußere Einflussnahme etwa durch Felder vorliegen soll, kann die Wahrscheinlichkeit für die eine oder andere Richtung zu jedem Sprungzeitpunkt nach einer Gleichverteilung gewählt werden. D. h. im eindimensionalen Fall jeweils zu $1/2$. Die Beschreibung der Bewegungsrichtung lässt sich durch zwei weitere Zustände ($E = \text{„gehe nach links“}$, $F = \text{„gehe nach rechts“}$) erfassen. Für die endgültigen Realisierung eines Platzwechsels werden zwei weitere Zustände ($C = \text{„lösche links“}$, $D = \text{„lösche rechts“}$) benötigt, d.h. solche Zustände werden im folgenden Aktualisierungsschritt auf den Zustand „leer“ gesetzt.

Die Regeln für die Zustandsüberführung, wonach sich die Teilchen im ZA bewegen, wurden aus kombinatorischen Erwägungen erstellt, wobei als Prinzip gelten soll, dass ein Teilchen sich frei, d.h. zufällig, bewegen soll, solange es nicht behindert wird. Die Regeln sind symmetrisch für die Zustandsüberführungen hinsichtlich linker und rechter Bewegungsrichtung.

Um das zeitliche Verhalten hinreichend genau beschreiben zu können, wurde eine Nachbarschaft von jeweils zwei linken und zwei rechten nächsten Nachbarn inklusive des betrachteten Gitterplatzes gewählt. Die zu beschreibenden Regeln (s. Abb. 1) müssen alle möglichen Konstellationen und Konflikte erfassen und auflösen. Werden für bestimmte Konstellationen keine Regeln angegeben, so soll der Zustand unverändert bleiben. Ein vollständiger Aktualisierungsschritt der Zustände kann in zwei getrennte Teilschritte aufgeteilt werden, in dem jeweils immer der Automat in Abb. 1 zum Einsatz kommt. Zu Beginn sind alle Zellen entweder im Zustand A oder B . Gemäß Abb. 1 werden durch einen Zufallsgenerator alle Zustände B verlassen, so dass nur noch die Zustände A , E oder F vorkommen. Es folgt der zweite Teilschritt mit möglichen Übergängen nach A , B , C oder D . Somit kann eine Aktualisierung zum einen in 4 (A , B , C , D) und zum anderen in 3 Zuständen (A , E , F) zerlegt werden, woraus insgesamt $4^5 + 3^5 = 1267$ Regeln resultieren.

Die Erhaltung der Teilchenzahl in jedem vollen Zeitschritt wird als selbstverständliche Forderung erfüllt. Durch die Regeln wird sichergestellt, dass ein leerer Gitterplatz nur von benachbarten Teilchen besetzt werden kann, sofern er von diesen infolge seiner Sprungrichtung beansprucht wird. Ein Konfliktfall liegt z.B. vor, wenn es zwei Anwärter auf den leeren Gitterplatz gibt, einen von links und einen von rechts kommend. Hier entscheidet wegen der geforderten Isotropie wieder der Zufall gleich verteilt über die Besetzung.

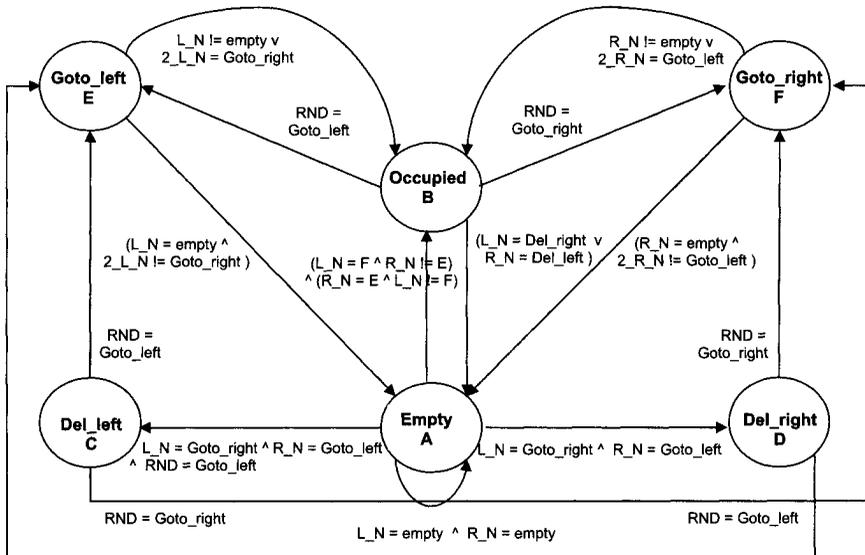


Abbildung 1: Zustandsübergänge im ZA-Modell der single-file-Diffusion (L_N , R_N entspricht linkem bzw. rechtem Nachbar, $2_L N$, $2_R N$ entspricht übernächstem linken bzw. rechten Nachbar, RND entspricht Wahrscheinlichkeit von $\frac{1}{2}$)

4. Das MPI-Interface

Zur Beschreibung des ZA in der MPI-Welt wird der Zustand und eine mit dem Teilchen verbundene Änderung der Distanz relativ zum Zeitpunkt 0 in einer Struktur `MPI_Cell` mittels `MPI_Type_commit` deklariert. Über die Zahl der zur Verfügung stehenden Prozesse, wird eine lineare Topologie mittels `MPI_Cart_create` erzeugt, die sich auf die verschiedenen Computer des Clusters aufteilen. Diese sollten möglichst gleiche Performanceeigenschaften besitzen, so dass in ihnen gleichgroße Segmente des Gesamtautomaten bearbeitet werden können und Synchronisationszeiteinbußen minimiert werden. Insbesondere in Multi-Cluster-Umgebungen kann dies nicht garantiert werden, so dass ein dynamischer Lastausgleich wünschenswert ist. In den durchgeführten Untersuchungen wurde dies noch nicht berücksichtigt. Es wurde bei der Auswahl der Rechnerknoten darauf geachtet, dass die Unterschiede in den einzelnen Knoten nicht zu groß sind.

Die zellulären Teilautomaten im jeweiligen Prozess werden mittels `MPI_Type_vector` bestehend aus dem Datentyp `MPI_Cell` erzeugt und um die Randlelemente, jeweils zwei Zellen für links und rechts, für die Kopien der Nachbarautomaten erweitert. Zum Austausch der Randdaten wird wiederum ein Vektor vom selben Datentyp deklariert, sowohl für die linken, als auch für die rechten Randdaten (Abb. 2).

Zu Beginn der Simulation werden sowohl die Zufallsgeneratoren der Teilautomaten als auch die Zustände der Zellen entsprechend der Konzentration initialisiert und die tatsächlich realisierte Teilchenzahl mittels `MPI_Reduce` und dem Parameter `MPI_SUM` im Prozess 0 aufsummiert. Außerdem muss die Prozesskommunikation für die beiden Ränder durch `MPI_Cart_rank` organisiert werden. Dadurch soll sichergestellt werden, dass die zwei Randzustände auf beiden Seiten eines jeden

Teilautomaten in die entsprechend erweiterten Position der Nachbarautomaten kopiert werden und so für den neuen Aktualisierungsschritt zur Verfügung stehen.

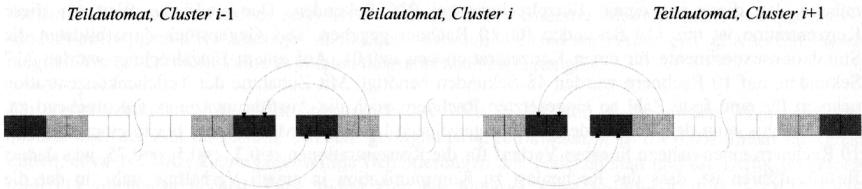


Abbildung 2: Datentransfer zwischen den Teilautomaten. Ein Teilautomat i bestehend aus den inneren Zellen (weiß) und den beiden Randzellen links und rechts (hellgrau). Aus Darstellungsgründen wird er durch eine symbolische Trennlinie unterbrochen. Die dunkelgrauen Zellen beinhalten die Kopien der Randzellen der benachbarten Teilautomaten $i-1$ bzw. $i+1$, welche für die Zustandsaktualisierung erforderlich sind.

Die Aktualisierung des Zustandes erfolgt wie zuvor beschrieben in zwei Teilschritten in einer iterativen Zeitschleife, deren Länge in jedem Teilautomaten gleich lang ist. Vor jeder Aktualisierung in den Teilautomaten erfolgt zuerst die Bestimmung des mittleren Verschiebungsquadrates und das aufsummierte Gesamtergebnis wird im Prozess 0 ausgegeben. Darauf folgt die zweistufige Zustandsaktualisierung. Diese wird eingeleitet durch den Austausch der Randzustände mittels **MPI_Sendrecv** und mit der jeweiligen Zustandstransformation abgeschlossen. Die Aktualisierung der Zustände in jedem der Teilautomaten geschieht sequentiell. Die resultierenden Zustände werden in einem gesonderten, zuvor organisierten Speicherbereich, aktualisiert. Nach Beendigung der Aktualisierungssequenz werden die Zeiger der Speicherbereiche vertauscht.

5. Experimentelle Untersuchungen

Die durchgeführten experimentellen Untersuchungen wurden auf einem Cluster-Rechner mit jeweils 2 bis 12 Rechnerknoten bei einer Gesamtzahl von 10 Millionen Teilchen mit 10.000 Aktualisierungsschritten für folgende Teilchenkonzentrationen $c=0,01$, $c=0,1$, $c=0,3$, $c=0,5$ und $c=0,75$ durchgeführt. Die Rechnerknoten waren mit Intel Pentium 4 Prozessoren mit einer Taktrate von 2 bis 3,2 GHz ausgestattet. Für den Programmlauf wurden jeweils 512 MByte pro Knoten allokiert, was weit ausreichend war. Für jede Teilchenkonzentration wurden insgesamt 5 Programmläufe durchgeführt und der Mittelwert der Ausführungszeit bestimmt, um unvermeidbare Schwankungen bei den Zeitmessungen auszugleichen. Die Ausführungszeiten wurden gemessen ohne dass zum Zeitpunkt der Messung weitere Lasten im Clusterrechner vorhanden waren.

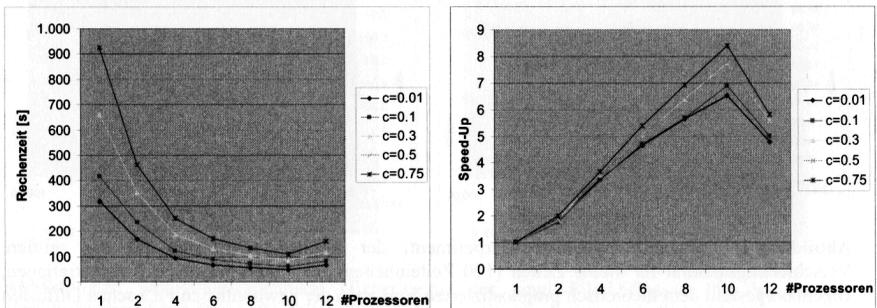


Abbildung 3: Ausführungszeiten und erzielter Speed-Up für verschiedene Teilchenkonzentrationen

Die Ausführungszeiten lagen wie Abb. 3 links zeigt durchwegs im einstelligen Minutenbereich. Für eine Teilchenkonzentration von 0,75, bei der aufgrund der höchsten Teilchenzahl die meisten Aktualisierungen in den Zuständen über alle betrachteten Konzentrationen vorgenommen werden müssen, lag diese auf einem Einzelrechner bei 926 Sekunden. Der niedrigste Wert für diese Konzentration ist mit 110 Sekunden für 10 Rechner gegeben. Das Gegenstück dazu bildeten die Simulationsexperimente für eine Konzentration von $c=0,01$. Auf einem Einzelrechner wurden 317 Sekunden, auf 10 Rechnern wurden 48 Sekunden benötigt. Mit Zunahme der Teilchenkonzentration nehmen für eine feste Zahl an eingesetzten Rechnern auch die Ausführungszeiten entsprechend zu. Abb. 3 rechts zeigt den Verlauf der zugehörigen Speed-Up-Werte. Man erkennt bis zu einer Zahl von 10 Rechnern einen nahezu linearen Verlauf für die Konzentrationen $c=0,3$, $c=0,5$, $c=0,75$, was darauf zurückzuführen ist, dass das Rechenlast zu Kommunikation in einem Verhältnis steht, in der die Rechenlast überwiegt. Dies ist bei den Konzentrationen $c=0,1$ und $c=0,01$ anders. Es zeigt sich, dass der Anstieg der Speed-Up-Kurven von 1 bis 10 Prozessoren ab 4 Prozessoren etwas einzuknicken beginnt, da aufgrund der geringeren Rechenlast der Kommunikationsanteil mehr Einfluss als bei den anderen Konzentrationen ausübt.

Auffällig ist der starke Rückgang des Speed-Ups bzw. die zugehörige Zunahme der Ausführungszeiten für alle Konzentrationen beim Übergang von 10 zu 12 Rechnern. Dies lässt sich dadurch erklären, dass bei einer Auswahl von mehr als 10 Rechnern ältere Rechnermodelle im Cluster-Rechner ausgewählt werden, die die anderen Prozessoren ausbremsen. Wie oben beschrieben, wurde ein dynamischer Lastausgleich bisher noch nicht implementiert.

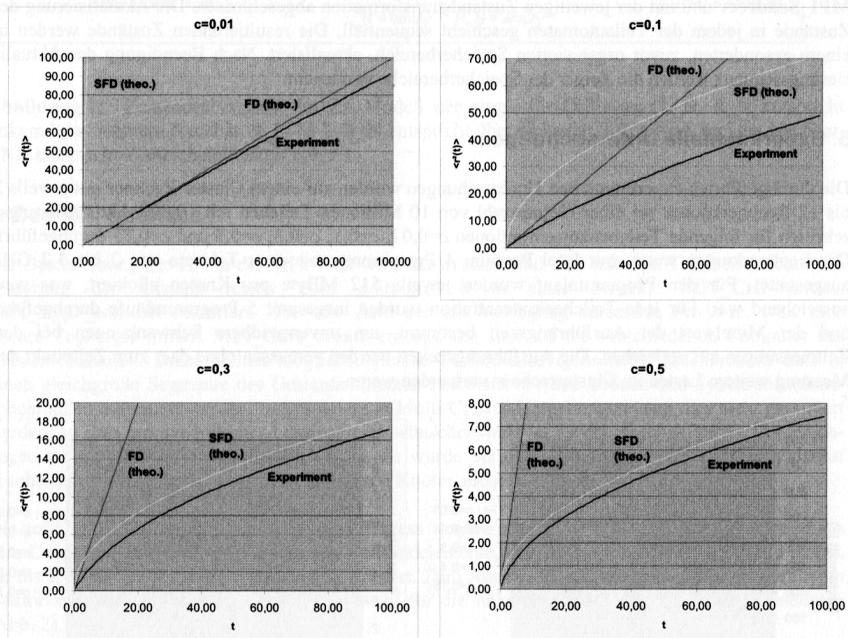


Abbildung 4: Simulationsergebnisse (Experiment) der single-File-Diffusion für das mittlere Verschiebungsquadrat für *kleine* Zeiten (100 Zeiteinheiten) mit unterschiedliche Konzentrationen, gegenübergestellt dem theoretisch prognostizierten Verhalten der gewöhnlichen FICKSchen Diffusion (FD) und der single-file-Diffusion (SFD).

6. Auswertung

Das vorgestellte ZA-Modell bestätigt die aus der Literatur bekannten Ergebnisse für den asymptotischen Zeitverlauf. Außerdem zeigt es die Fähigkeit ZA basierter Modelle nicht nur qualitative, sondern auch quantitative Aussagen zu machen. Insbesondere sind auch Aussagen im Kurzzeitbereich möglich, für die es bisher keine expliziten theoretischen Aussagen gibt.

Zur Veranschaulichung der Simulationsergebnisse wird in der Abb. 4 das mittlere Verschiebungsquadrat über der Zeit dargestellt. Bei sehr geringen Konzentrationen, z. B. $c \approx 0.01$, finden für kleine Zeiten, etwa $t < 100$, kaum gegenseitige Behinderungen der Teilchen statt und die Diffusion erfolgt nach (1) annähernd normal. Dies ändert sich bei höheren Konzentrationen, z. B. $c \approx 0.1$, schon deutlich. Weder das lineare Verhalten des mittleren Verschiebungsquadrates der gewöhnlichen Diffusion noch die angegebene asymptotische Formel der single-file-Diffusion geben eine befriedigende Erklärung. Hier besteht noch Erklärungsbedarf seitens der Theorie. Bei noch größeren Konzentrationen, z.B. $c \approx 0.3$ erkennt man schon bei kleineren Zeiten das prognostizierte asymptotische Verhalten für die single-file-Diffusion.

Die Simulationsergebnisse des mittleren Verschiebungsquadrates zeigen für große Zeiten (s. Abb. 5) eine gute Übereinstimmung mit den theoretischen Werten. Sowohl das qualitative quadratwurzelartige Verhalten in der Zeit, als auch das quantitative Verhalten für niedrige und mittlere Konzentration kann reproduziert werden. Leichte Abweichungen treten erst bei höheren Teilchenkonzentrationen $c > 0.75$ auf. Diese können u. a. auf die vereinfachenden Annahmen des Modells wie die beschränkte Nachbarschaft auf zwei nächste Nachbarn zurückgeführt werden, wodurch mögliche höhere Korrelationseffekte vernachlässigt werden.

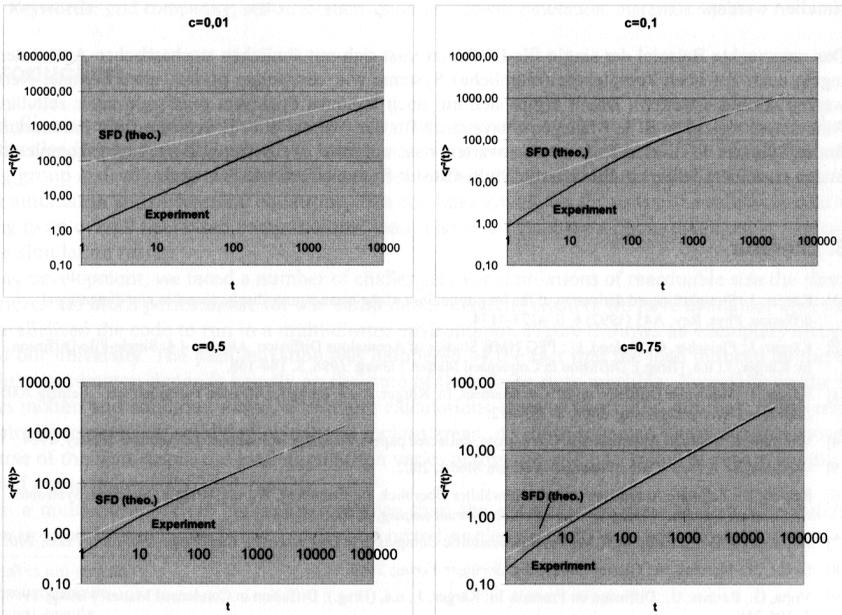


Abbildung 5: Simulationsergebnisse (Experiment) der single-File-Diffusion für das mittlere Verschiebungsquadrat für *große* Zeiten (32000 Zeiteinheiten) mit unterschiedliche Konzentrationen, gegenübergestellt dem theoretisch prognostizierten der single-file-Diffusion (SFD).

7. Ausblick

Das hier untersuchte ZA-Modell zeigt die Möglichkeiten, praxisrelevante Probleme auch quantitativ zu untersuchen. Daher sollte es nicht verwunderlich sein, wenn es zunehmend, neben den bewährten Simulationsmethoden wie Monte-Carlo oder Molekulardynamik, an Bedeutung gewinnt. Die Vorzüge des Modells liegen in dessen einfacher plausiblen Beschreibung. Es ist flexibel durch zusätzliche Regeln erweiterbar, wodurch sich weitere Anwendungsgebiete erschließen lassen.

Ein solches wichtiges praktisches Problem, bei dem etwa die Berücksichtigung der Randeffekte von Bedeutung ist, ist die single-file-Diffusion von Ionen durch eine Membran über einen engen Kanal, z.B. in einer biologischen Zelle [10], [11]. An dessen beiden Enden liegen im Allgemeinen verschiedene Konzentrationen vor. Infolge dessen erfolgt ein Stoffausgleich (Ionenstrom) durch diesen Kanal. Den Unterschied kann man über die Potentialdifferenz messen. Aber auch die oben angesprochenen Prozesse im Katalysator haben in Wirklichkeit einen solchen Stoffaustausch zum Ziel. In beiden Fällen kann man die Randeffekte im ZA-Modell durch Austauschraten zwischen einem Reservoir und den Randzellen beschreiben. Mit den für die Praxis immer wichtiger werdenden (Kohlenstoff-) Nanoröhrchen liegen gleichfalls analoge Strukturen vor, für die der hier untersuchte Stofftransport von Interesse sein könnte.

Die Zustandsüberführungen im ZA-System erfolgen bisher mit einem einheitlichen Zeittakt. Um den mikroskopischen stochastischen Charakter des Systems noch besser beschreiben zu können, besteht die Möglichkeit Sprungzeitverteilungen einführen. Dann würden Platzwechsel zusätzlich zu den bestehenden Regeln noch von durch die Wartezeitverteilung gegebenen Zusatzbedingungen abhängen. Diese lassen sich zusätzlich asymmetrisch definieren, um auch (externe) Felder erfassen zu können. Auch gerichtete Ströme von Partikeln, etwa von Fahrzeugen auf Autobahnen können auf diese Weise simuliert werden.

Das untersuchte Beispiel der single-file-Diffusion lässt sich mit ähnlichen stochastischen Automatenregeln auch auf noch komplexere (räumliche) Systeme wie verzweigte poröse, etwa fraktale, Netzwerkstrukturen erweitern. Damit ergibt sich ein noch breiteres Spektrum praxisrelevanter zellulärer Automaten, welche z. B. in Managementsystemen für den Waren- oder Energietransport Anwendung finden könnten. In diesem Falle ist zu erwarten, dass aufgrund der größeren Berechnungskomplexität in den einzelnen Zellen ein Einsatz von Multi-Cluster-Systemen erforderlich wird.

8. Literatur

- [1] Kärger, J.: Straightforward derivation of the long-time limit of the mean-square displacement in one-dimensional diffusion. *Phys. Rev. A* 45 (1992) 6, S. 4173-4174.
- [2] Kärger, J.; Fleischer, G.; Roland, U.: PFG NMR Studies of Anomalous Diffusion, Abschn. 4.4: Single-File-Diffusion. In: Kärger, J.; u.a. (Hrsg.): *Diffusion in Condensed Matter. Vieweg, 1998*, S. 144-168.
- [3] Kärger, J.: Molecular Diffusion under Confinement. In: Kärger, J. u.a. (Hrsg.): *Diffusion Fundamentals – Leipzig 2005, Leipziger Universitätsverlag, 2005*, S. 389-412.
- [4] Wolfram, S.: *Cellular Automata and Complexity, collected papers, Reading/ Massachusetts: Addison-Wesley, 1994*
- [5] Wolfram, S.: *A New Kind of Science. Wolfram Media, 2002.*
- [6] Renner, U.: Zelluläre Automaten – Ein ausgewählter Überblick. In: Eisenberg, W.; u.a. (Hrsg.): *Synergie, Syntropie, Nichtlineare Systeme. Verlag im Wissenschaftszentrum Leipzig, 2000*, S. 31-54.
- [7] Karniadakis, G. E.; Kirby II, R. M.: *Parallel Scientific Computing in C++ and MPI. Cambridge University Press, 2003.*
- [8] Bauke, H.; Mertens, St.: *Cluster Computing. Springer-Verlag, 2006.*
- [9] Vojta, G.; Renner, U.: Diffusion on Fractals. In: Kärger, J.; u.a. (Hrsg.): *Diffusion in Condensed Matter. Vieweg, 1998*, S. 306-318.
- [10] K. Heckmann: Zur Theorie der "single-file", Diffusion, I. *Zeitschrift für Physikalische Chemie, Neue Folge*, 1965, Bd. 44, S.184-203.
- [11] S. Kh. Aityan: Single-File-Diffusion of Uncharged Particles. *Gen. Physiol. Biophys.* 4 (1985), S. 3-14.