

Erlangen, 3. März 2008

Promotionsvortrag: Thomas Zeiser

extended version

Simulation und Analyse von durchströmten Kugelschüttungen in engen Rohren unter Verwendung von Hochleistungsrechnern

Technische Fakultät
Friedrich-Alexander-Universität
Erlangen-Nürnberg



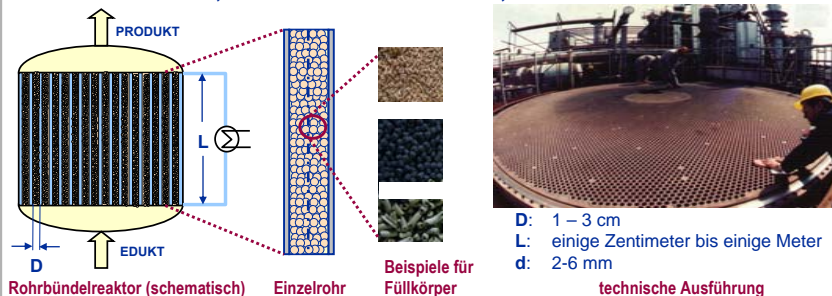
Promotionsvortrag: Thomas Zeiser Gliederung

1. Motivation und Ziel der Arbeit
2. Grundlagen der Lattice-Boltzmann-Verfahren
3. Effiziente Implementierung von Lattice-Boltzmann-Verfahren
4. Methoden zur Charakterisierung von Zufallsschüttungen: Gewinnung, Aufbereitung und Analyse der 3-D Struktur
5. Anwendungsbeispiel: Numerische Strömungsuntersuchungen als Ergänzung von komplementären NMR-Messungen
6. Zusammenfassung und Ausblick

2

1. Motivation und Ziel der Arbeit Warum durchströmte Kugelschüttung in engen Rohren?

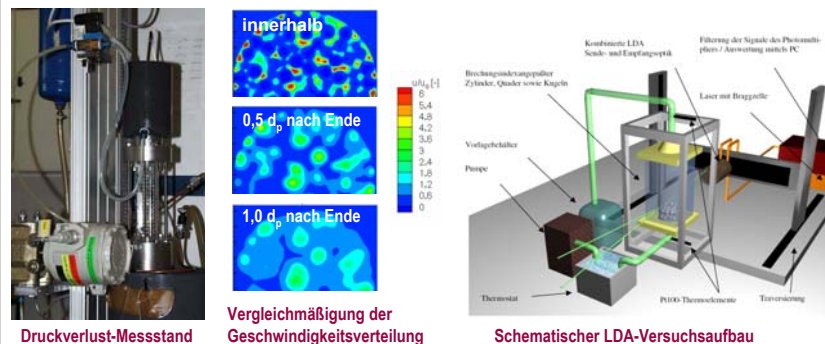
- Technisch relevante chemischen Umsetzungen laufen überwiegend katalytisch ab – meistens heterogen katalysiert.
- Typische Reaktoren für die heterogene Katalyse sind:
 - Festbettreaktoren (Vollraum-, Horden-, **Rohrbündelreaktor**)
 - Wirbelschicht; Rührkessel im Batchbetrieb, ...



- in meiner Arbeit: Beschränkung auf die **Hydrodynamik** einphasiger, **isothermer, newtonscher Fluide** im explizit aufgelösten Porenraum zwischen den als kompakt angenommenen, **kugelförmigen Füllkörpern** („Bulk-Strömung“)
- für Reaktionstechnik: + Stofftransport v.a. innerhalb der Füllkörper
+ Wärmetransport / Temperaturverteilung

1. Motivation und Ziel der Arbeit Warum orts aufgelöste numerische Untersuchungen?

- **Integrale Größen** können experimentell gut bestimmt werden.
- **Lokale Größen** lassen sich jedoch nur mit großem Aufwand messen.
- Geschwindigkeitsmessungen **hinter** der Schüttung können die Verhältnisse im Inneren nur bedingt wiedergeben.
- **Transportvorgänge** werden in entscheidendem Maße jedoch von der **Geometrie** und den **lokalen Prozessen** bestimmt.



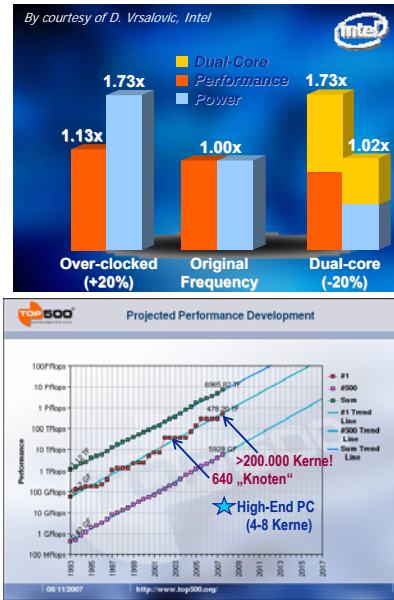
- **detaillierte, orts aufgelöste numerische Untersuchungen — komplementär zu NMR-Untersuchungen der For-338/Uni-Karlsruhe**

4

3

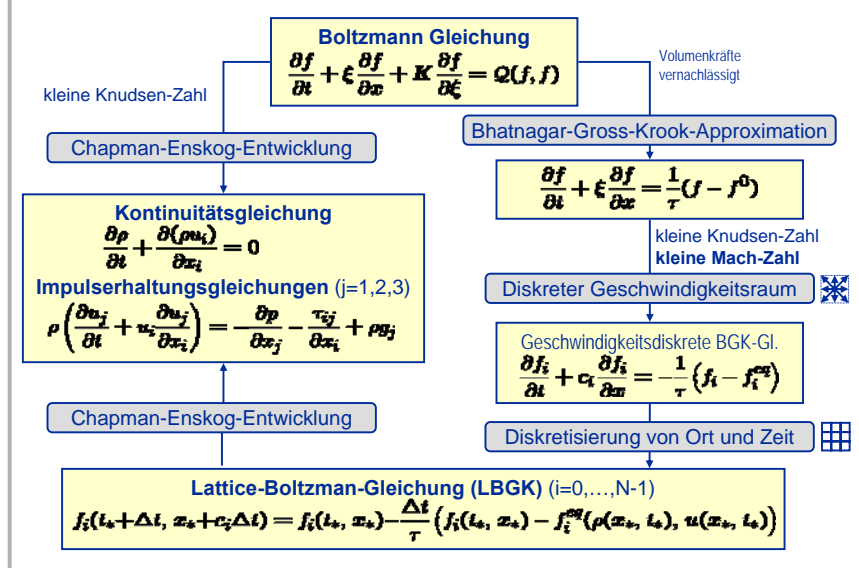
1. Motivation und Ziel der Arbeit Warum Nutzung von und Optimierung für Hochleistungsrechner?

- Moore's Law („Verdoppelung der Transistoren in 18 Monaten“) wird noch einige Zeit gelten — der einzelne Rechenkern wird aber nicht mehr signifikant schneller werden.
 - Die Spitzenleistung von Hochleistungsrechnern wird weiter steigen — aber nur durch erhöhte Parallelität („Masse statt Klasse“).
 - Die Speicherbandbreite wird nur langsam ansteigen (bzw. pro Rechenkern sogar deutlich sinken) — die Zahl der Speicherhierarchien nimmt zu.
- spezifische Optimierungen (weiterhin) nötig!
→ Einzelprozesseroptimierung bleibt der erste Schritt.

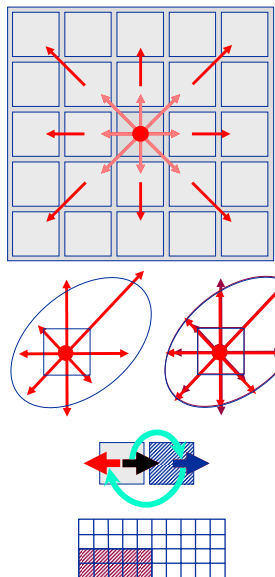


Abbildungen: intel bzw. top500.org

2. Grundlagen der Lattice-Boltzmann-Verfahren: Von „Boltzmann“ zu „Lattice-Boltzmann“ und „Navier-Stokes“



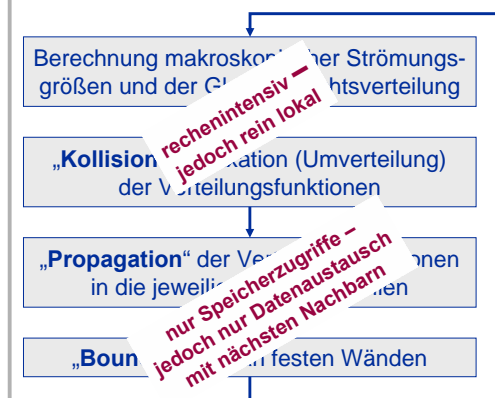
2. Grundlagen der Lattice-Boltzmann-Verfahren: Umsetzung des Lattice-Boltzmann-Ansatzes



- Das Gebiet wird in „Voxel“ zerlegt.
- In jedem Voxel werden N Verteilungsfunktionen f_i gespeichert.
- In jedem Zeitschritt „springen“ diese Verteilungsfunktionen entsprechend ihrer mikroskopischen Geschwindigkeit in die jeweils nächste Zelle („Propagation“).
- Betrag und Richtung der mikroskopischen Geschwindigkeiten ist durch die Gitterwahl bestimmt.
- Die Verteilungsdichten sind dagegen orts- und zeitabhängig.
- Durch Kollisionen wird der Gleichgewichtszustand angenähert („Relaxation“).
- Durch „Bounce-back“ an Hindernissen wird die Haft-Wandrandbedingung realisiert.
- Beliebig komplizierte Strömungsgeometrien lassen sich durch den Marker-and-Cell-Ansatz darstellen.

2. Grundlagen der Lattice-Boltzmann-Verfahren: Umsetzung des Lattice-Boltzmann-Ansatzes (II)

Örtliche und zeitliche Entwicklung der Verteilungsfunktion f_i jeder Zelle in den diskreten Zeitschritten:



$$\rho = \sum_i f_i$$

$$\rho u = \sum_i c_i f_i$$

$$f_i^{eq}(x_*, t_*) = t_p \rho \left\{ 1 + \frac{c_i u}{c_s^2} + \frac{(c_i u)^2}{2c_s^4} - \frac{|u|^2}{c_s^2} \right\}$$

$$\frac{1}{\tau} (f_i(x_*, t_*) - f_i^{eq}(x_*, t_*)) = Q$$

mit $\nu = \frac{2\tau-1}{6}$

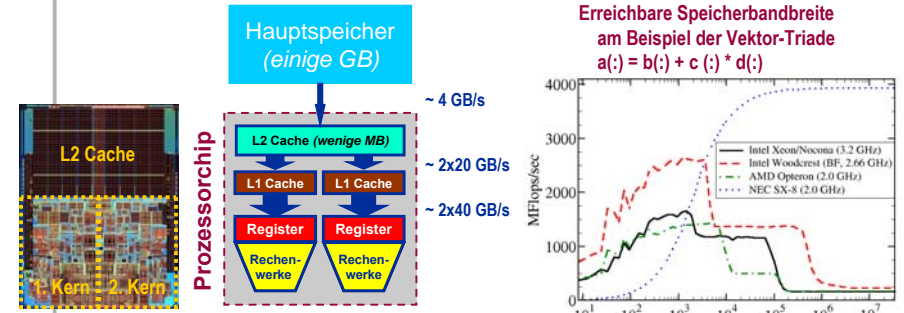
$$f_i(x_* + c_i, t_* + 1) = f_i(x_*, t_*) - Q$$

3. Effiziente Implementierung von Lattice-Boltzmann-Verfahren

Ausgewählte Optimierungsstrategien:

- „verständlichen“ Code schreiben
→ für Mensch und Compiler hilfreich
 - Eliminierung unnötiger Rechenoperationen
→ insbesondere bei der Gleichgewichtsberechnung
 - geschickte mathematische Umformung
→ Reduktion von Rundungsfehlern durch Subtraktion von konstantem Anteil
 - Speicherzugriffe *minimieren*
→ Kollisions- und Propagationsschritt zusammen in einer Schleife ausführen
 - Speicherzugriff optimieren und Speicherhierarchie berücksichtigen
→ korrektes Datenlayout wählen: $f(x,y,z,i,t) \leftrightarrow f(i,x,y,z,t)$
→ geblockte Schleifen oder raumfüllende Kurven statt lexikographische Reihenfolge
→ ccNUMA, Loop-Splitting, ...
 - (problemspezifisch) angepasste Datenstrukturen
→ nur 1-D Liste der Fluidzellen statt maskierte 3-D Arrays speichern
- erst seriell: Performancemodellierung + Einzelprozessoroptimierung!
→ dann parallel: Speicherzugriff + Lastverteilung bei LB entscheidend!

3. Effiziente Implementierung von Lattice-Boltzmann-Verfahren Performance-Modellierung



Für den Lattice-Boltzmann-Algorithmus gilt mit dem Performance-Maß „MLUP/s“ = million lattice updates per second

- Limitierung durch Rechenleistung (Daten sind in unendlich schnellem Cache)
 - max. MLUP/s = Peakperformance in MFlop/s / 200 Flops
 - aktuelle Intel CPUs: theoretisch bis zu ~60 MLUP/s
- Limitierung durch die Speicherbandbreite (... d.h. im Normalfall)
 - max. MLUP/s = (effektive) Speicherbandbreite in MB/s / 456 Bytes
 - aktuelle Systeme: bis zu ~8 MLUP/s

3. Effiziente Implementierung von Lattice-Boltzmann-Verfahren Einfluss des Datenlayouts auf die Performance

```

real*8 f(0:18,0:Nx+1,0:Ny+1,0:Nz+1,0:1) f(0:Nx+1,0:Ny+1,0:Nz+1,0:18,0:1)
do z=1,Nz; do y=1,Ny; do x=1,Nx
  if( fluidcell(x,y,z) ) then
    LOAD f( 0,x,y,z,t)
    LOAD f( 1,x,y,z,t)
    ...
    LOAD f(18,x,y,z,t)
  Relaxation (complex computations)
  SAVE f( 0,x,y,z,t+1)
  SAVE f( 1,x+1,y,z,t+1)
  SAVE f( 2,x,y+1,z,t+1)
  SAVE f( 3,x-1,y+1,z,t+1)
  ...
  SAVE f(18,x,y-1,z-1,t+1)
endif
enddo; enddo; enddo
    
```

„Kollision“: LOAD f(0,x,y,z,t) ... LOAD f(18,x,y,z,t) im Speicher aufeinander folgende Daten

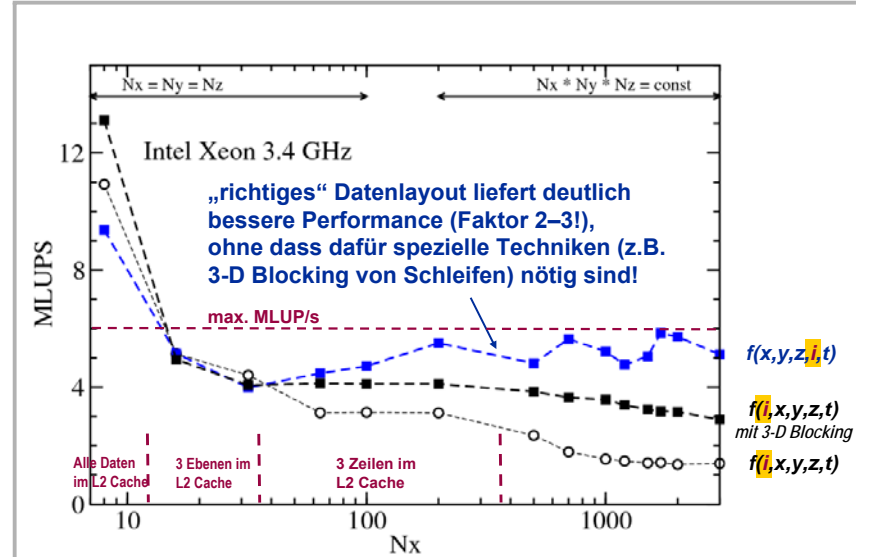
„Propagation“: SAVE f(0,x,y,z,t+1) ... SAVE f(18,x,y-1,z-1,t+1) im Speicher weit auseinander; Cachezeilen kaum wieder verwendbar

19 verschiedene Cachezeilen, aber weiter verwendbar

19 Cachezeilen; weiter verwendbar

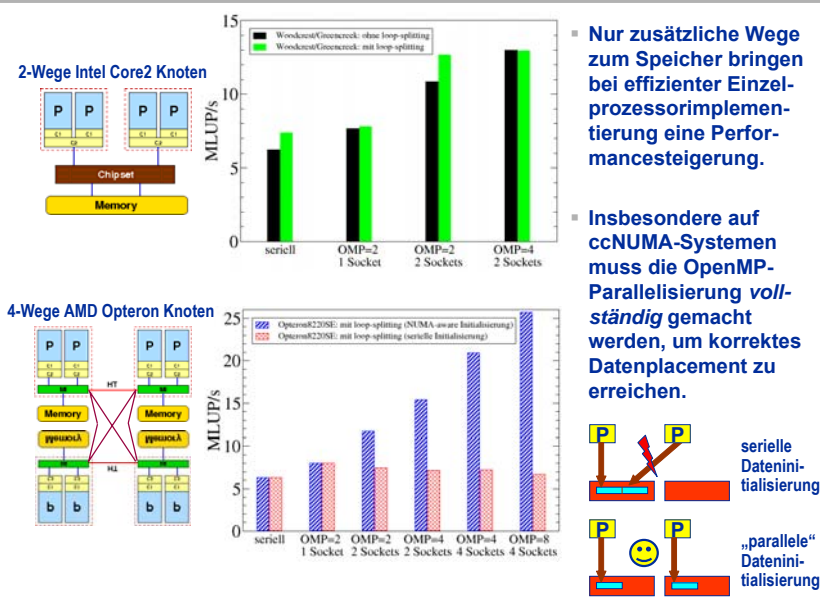
- Fortran-Arrays: 1. Index adressiert fortlaufende Speicherbereiche
- Cache-Organisation
 - es müssen immer ganze Cachezeilen (= 128 Bytes = 16 Gleitkommazahlen) mit dem Hauptspeicher ausgetauscht werden
 - es können nur Daten modifiziert werden, die im Cache vorhanden sind, d.h. „modifizieren“ = „read for ownership“ + „schreiben“ (Daten gehen 2x über den Speicherbus)

3. Effiziente Implementierung von Lattice-Boltzmann-Verfahren Einfluss des Datenlayouts auf die Performance



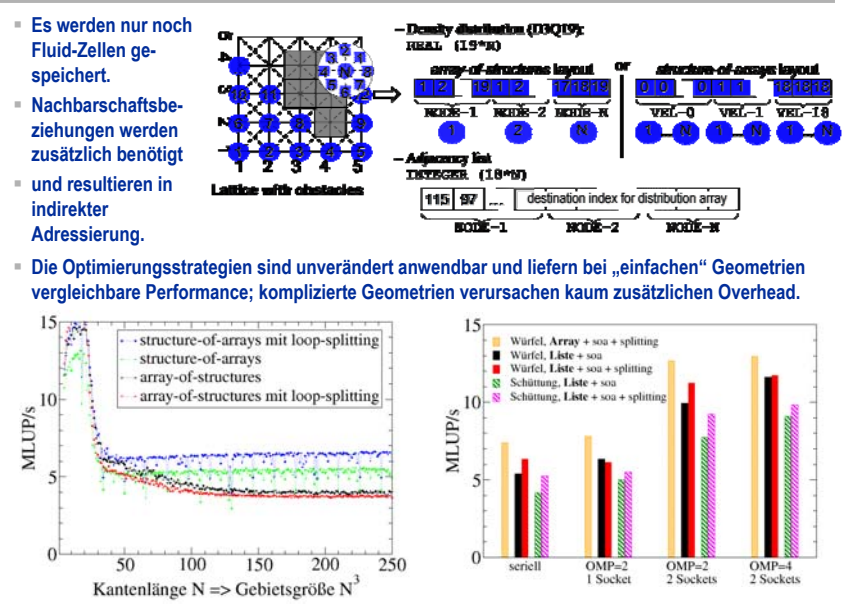
3. Effiziente Implementierung von Lattice-Boltzmann-Verfahren Skalierbarkeit innerhalb eines Rechenknotens

Schematische Darstellungen: G. Hager

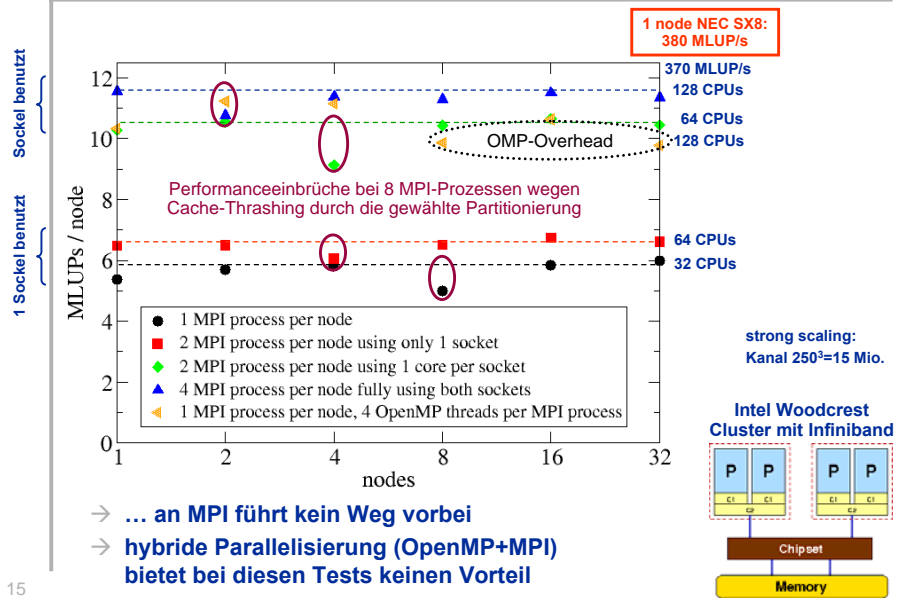


3. Effiziente Implementierung von Lattice-Boltzmann-Verfahren „Sparse-LBM“: Datenstruktur für poröse Medien

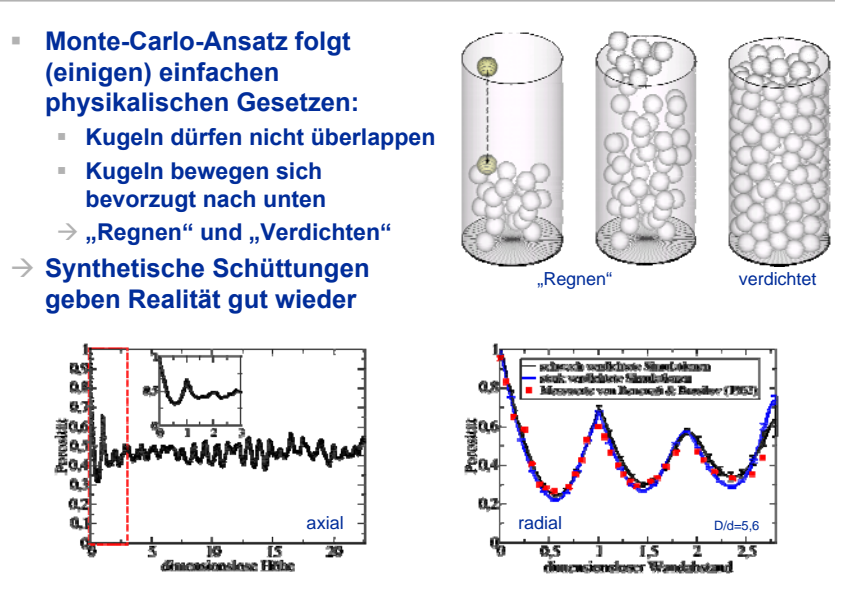
Ursprüngliche Skizze der Konzeptdarstellung: J. Bernsdorf
Grundidee für die Datenstruktur: C. Pan (2004), Karliczyk et al. (2000,2001,2002)



3. Effiziente Implementierung von Lattice-Boltzmann-Verfahren Parallele Skalierbarkeit über Knotengrenzen hinweg



4. Gewinnung, Aufbereitung und Analyse der Geometrie Synthetische Schüttungsgenerierung



4. Gewinnung, Aufbereitung und Analyse der Geometrie Messung mit bildgebenden Verfahren: NMR/MRI

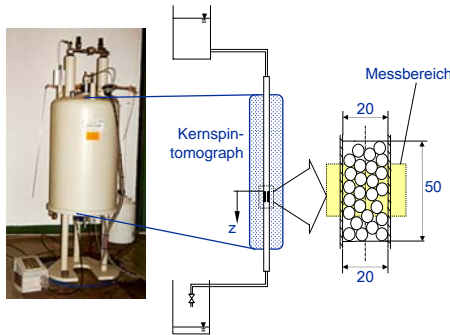
Schematische Darstellung des NMR-Messaufbaus der For338/Uni-Karlsruhe: Schwerkraft getriebene Durchströmung von Schüttungen

Bruker Advance 200 NMR-Tomograph

- Messprinzip: Wechselwirkung von Protonen mit elektro-magnetischen Feldern
- Statisches Magnetfeld: 4,7 Tesla
- Resonanzfrequenz 200 MHz

Ausgewählte Schüttung

- Durchmesser Verhältnis Rohr/Kugel=5,0
- Kugeldurchmesser 4mm
- Messung 3-D Geometrie:**
Erreichte Auflösung: 200 bzw. 100 μm
 128^3 (20 Voxel/Kugeldurchmesser)
 256^3 (40 Voxel/Kugeldurchmesser)
Messzeit für 3-D Geometrie bei 256^3 Voxel: 20-60h
- Geschwindigkeitsmessung** nur in 2-D Ebenen mit 128^2 Voxel und 64 Geschwindigkeitsklassen für FoF;
Messdaten für 3 horizontale Ebenen aus der Mitte der Schüttung verfügbar

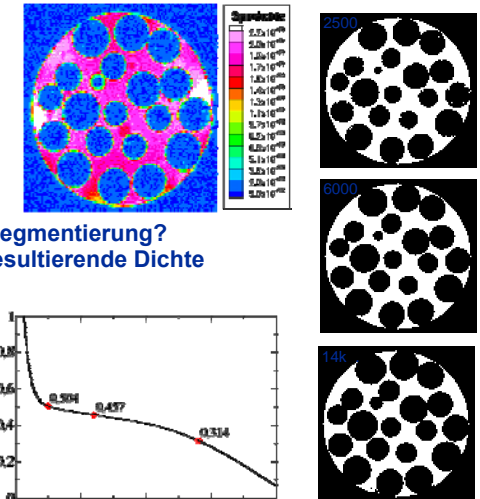


→ Vergleich von Simulationsergebnissen und Messdaten für ein und dieselbe Zufallsschüttung möglich

Abbildungen: C. Heinen & J. Tillich, Uni-Karlsruhe, 2003

4. Gewinnung, Aufbereitung und Analyse der Geometrie Aufbereitung der NMR-Bilddaten: Segmentierung

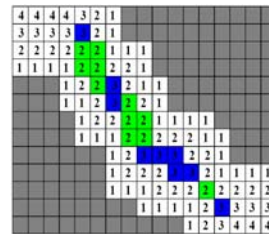
- Die NMR-Tomographie liefert „Spindichten“ für die einzelnen Voxel.
 - Für die weitere Verarbeitung müssen aus diesen „Graustufen“ „Schwarz-Weiss“-Bilder erzeugt werden.
- Korrekter Schwellwert der Segmentierung? Häufigkeitsverteilung und resultierende Dichte sind nur Anhaltspunkte!



Exemplarisch für den 128^3 -Datensatz bzw. 2-D Ebene daraus

4. Gewinnung, Aufbereitung und Analyse der Geometrie Einfache morphologische Analyse von (3-D) Pixelbildern

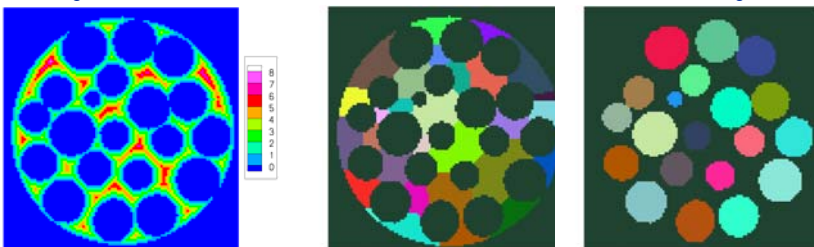
- Skeletierung/Mittelpunktsbestimmung im Gegensatz zur Porosität weitgehend unempfindlich gegenüber dem Segmentierungsschwellwert**
- Durch Umkehrung: automatische Identifikation von einzelnen Poren/Kugeln**
Dullien: Grenze zwischen zwei Poren = Stelle mit der geringsten Querschnittsfläche



→ Porengröße/Wandabstand

→ einzelne Poren

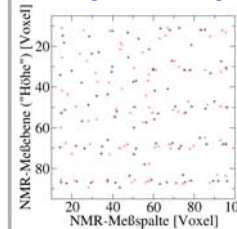
→ einzelne Kugeln



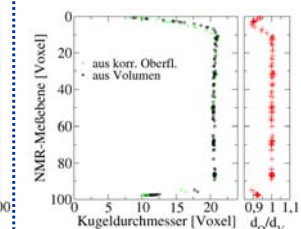
Bildanalyse exemplarisch für eine 2-D Ebene aus dem 128^3 -Datensatz

4. Gewinnung, Aufbereitung und Analyse der Geometrie Analyse der NMR-Geometriedaten

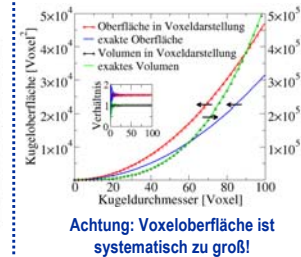
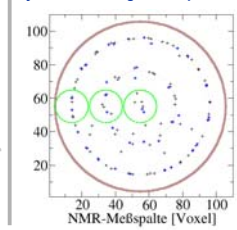
Wie zufällig ist die Kugelanordnung?



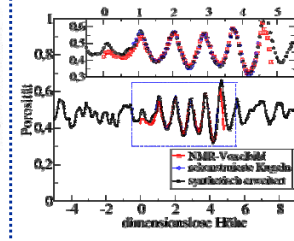
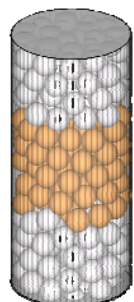
Wurden Kugeln erkannt?



Vertikale bzw. horizontale Projektion der Kugelmittelpunkte



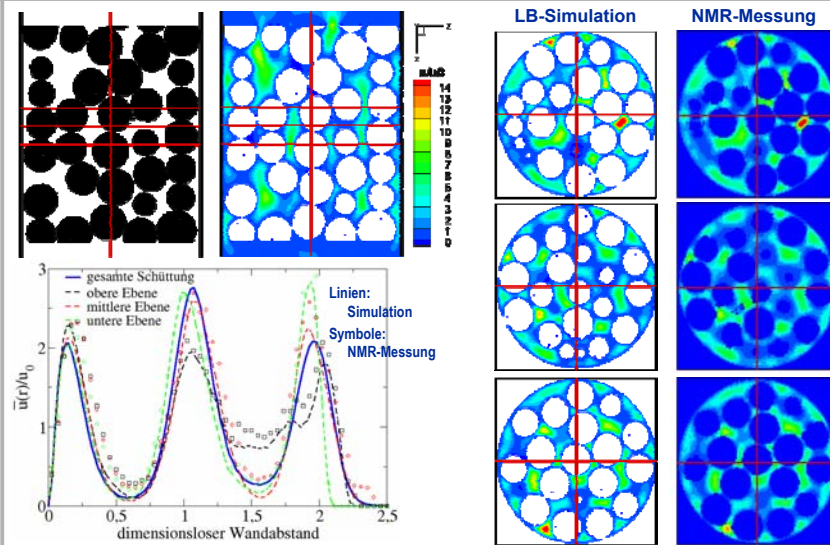
Synthetische Verlängerung



Bildanalyse mit 128^3 -Datensatz

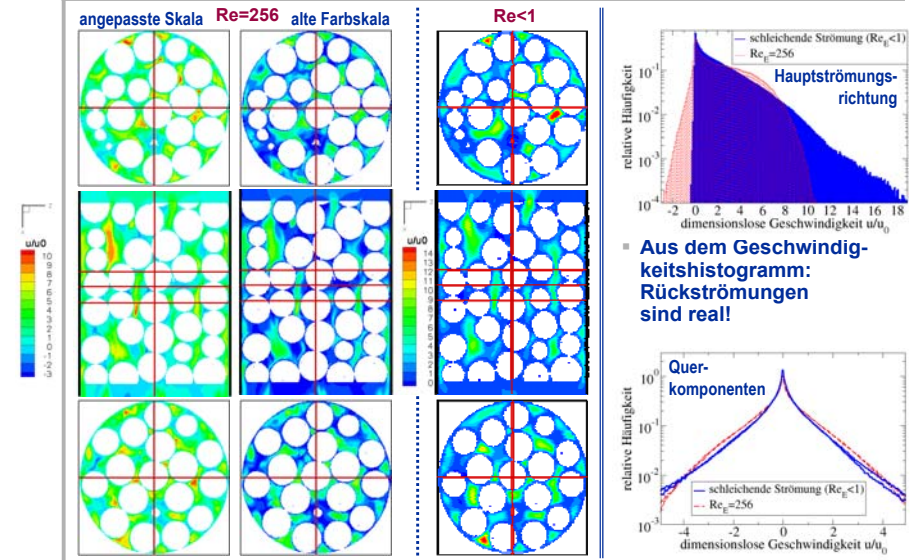
5. Anwendungsbeispiel: Vergleich von Simulation und Messung – Geschwindigkeitsfeld bei schleichender Strömung (Re<1)

Simulation auf Basis des 256³-Datensatzes; NMR-Geschwindigkeit auf 128³



- Gute Übereinstimmung zwischen Simulation und Messung
- Starke lokale Geschwindigkeitsschwankungen; $u_{max}/u_0 > 15$

5. Anwendungsbeispiel: Erweiterter Parameterbereich – Simuliertes Geschwindigkeitsfeld bei höherer Reynoldszahl, Re=256



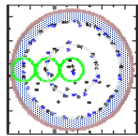
Re ↑: weniger stark ausgeprägte lokale Geschwindigkeitsspitzen

▪ Aus dem Geschwindigkeitshistogramm: Rückströmungen sind real!

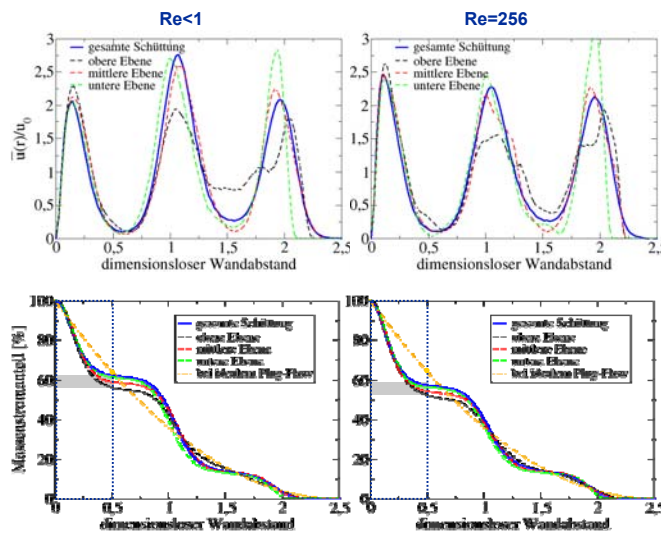
5. Anwendungsbeispiel Quantifizierung der Randgängigkeit

Definition der Randgängigkeit nach Vortmeyer&Schuster, CES, 1983

- Mit steigender Re-Zahl findet eine Massenstromverlagerung in den Wandbereich statt.



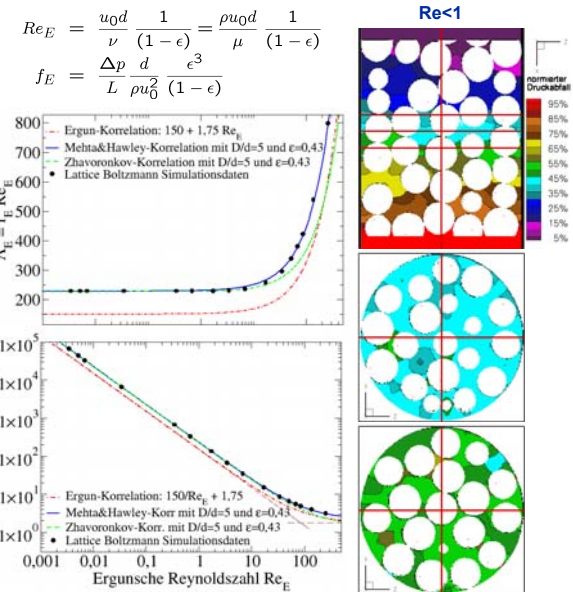
- Der Massenstromanteil zwischen Wand und äußerstem Kugerring beträgt >40% und ist höher als der entsprechende Flächenanteil



5. Anwendungsbeispiel Druckabfall und Druckverlust dieser Schüttung

- Ergun-Gleichung unterschätzt den Druckverlust (wie erwartet) signifikant
- gute Übereinstimmung mit Korrelation von Mehta&Hawley im gesamten Untersuchungsbereich

- Druckverteilung im Querschnitt nicht exakt gleich
- ausreichende Schüttungslänge bei Druckverlustmessungen über Wandbohrungen nötig



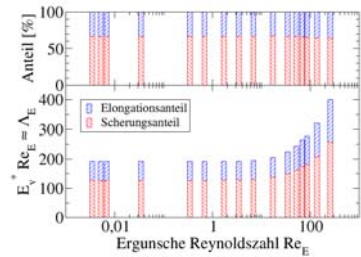
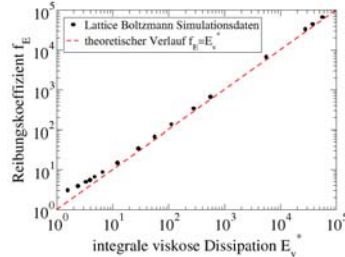
5. Anwendungsbeispiel Zusammensetzung des Druckverlusts: Dehnung + Scherung

Ausgangspunkt: Mechanische Energiegleichung in Differentialform

$$\frac{\partial(\rho u_j^2)}{\partial t} + \frac{\partial(\rho u_j u_k)}{\partial x_k} = \frac{\partial(\rho u_j)}{\partial x_j} + \frac{\partial(\tau_{ij})}{\partial x_i} + \tau_{ij} \frac{\partial u_j}{\partial x_i}$$

nach Integration und Umformung: $f \approx \mu \int_V \phi dV \frac{l_c}{AL\rho u_0^3}$

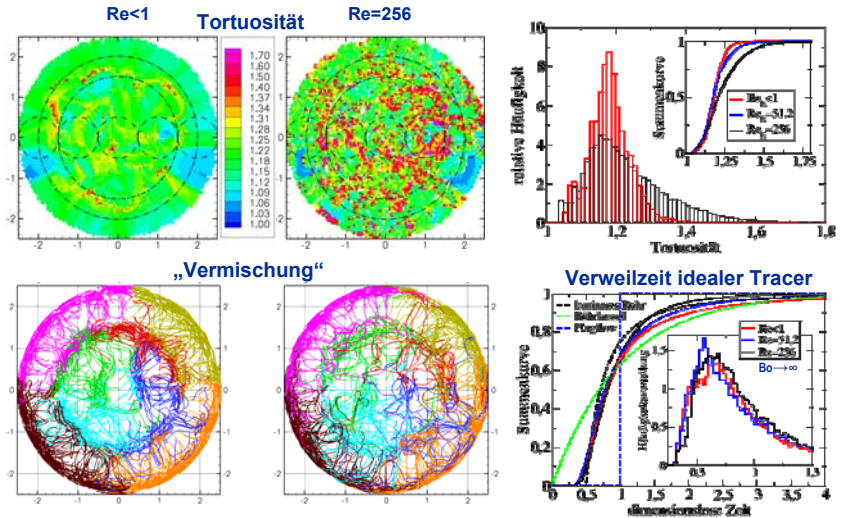
$$\text{mit: } \phi = 2 \left[\underbrace{\left(\frac{\partial u_1}{\partial x_1}\right)^2 + \left(\frac{\partial u_2}{\partial x_2}\right)^2 + \left(\frac{\partial u_3}{\partial x_3}\right)^2}_{\text{Dehnung}} + \underbrace{\left(\frac{\partial u_2}{\partial x_1} + \frac{\partial u_1}{\partial x_2}\right)^2 + \left(\frac{\partial u_3}{\partial x_2} + \frac{\partial u_2}{\partial x_3}\right)^2 + \left(\frac{\partial u_1}{\partial x_3} + \frac{\partial u_3}{\partial x_1}\right)^2}_{\text{Scherung}} \right]$$



Geometrie	NMR-Kugelschüttung	2. Kugelschüttung	SIC-Schaum	Knochen ↓	Knochen ↔
Scherungsanteil	66%	57%	63%	70%	55%

Details: siehe Bird, Steward, Lightfoot (2002); Durst et al. (1987)

5. Anwendungsbeispiel Tortuosität, Vermischung und Verweilzeit idealer Tracer



Re ↑: breitere Tortuositätsverteilung, stärkere Vermischung, jedoch gleichmäßigere Verweilzeitverteilung

6. Zusammenfassung und Ausblick

- Leistungsfähige Tools für die gesamte Simulationskette (Geometrieaufbereitung & Strömungssimulation) wurden entwickelt.
- Die Gesamtleistung von Hochleistungsrechnern wird weiterhin ansteigen. Der Programmieraufwand, um die hohe Leistung auch wirklich nutzen zu können, steigt jedoch auch deutlich.
- Exemplarische Anwendung der Tools auf eine Kugelschüttung, die in Karlsruhe auch experimentell untersucht wurde.
- Simulationen auf Porenebene sind geeignet für „Detailuntersuchungen“ und auch zum Reaktordesign,
- jedoch ungeeignet, um beispielsweise Druckverlustkurven systematisch als Funktion der Reynoldszahl aufzunehmen.

Ausblick:

<p><u>medizinische Anwendung</u></p>	<p><u>chemische Reaktorentwicklung</u></p>	<p><u>methodische Erweiterung</u></p> <ul style="list-style-type: none"> Stofftransport nicht-isotherm ... <p><u>algorithm. Erweiterung</u></p>
--------------------------------------	--	--

Danksagung