

1.) Einführung

Marten Richter, EW740

Tel: 33 314-24858

Email: mrichter@itp.fu-berlin.de

Sprechstunde: festlegen oder n.V

Thema: Was es nicht ist:

Kein Programm / C-Kurs (Algorithmen)

Kein Quantencomputing (Wiederherstellen eines Systems mit klassischem Computer)

Kein typisches Quanten Feld

Was es ist:

Experiment: Verbindung von Praxis ^{Computerorientierte} Theorie der Physik in der Datenphysik

Illustration an Beispielen aus der Großskalendatenscience
der Theorie der Physik, viel Quantenmechanische Beispiele

"Bunte" Mischung an angewandter Informatik, Mathematik
und Beispielen aus verschiedenen Gebieten

Gute Vorbereitung: für Bachelor, Master-, Doktor-,

Post Doc ... in computerorientierter Theoretischer Physik

Vorbereitung versucht fortgeschrittenen Niveau zu bieten
angelehnt an Wissen des Physikstudiums

Gute Parallelisierbare Codes, auch moderne Tensoralgorithmen

⇒ Ready for big Problems from Cluster

Literatur: Wird Abschlusstsweise bekannt gegeben

Computer assisted Quantenmechanik (Vorlesung)

Das Programm (vertikal)

1.) Einführung

Informelle
I.

2.) Rechnerarchitektur

(Was muss man wissen bei Programmieren)

3.) (Nicht-) Lineare Algebra für parallele Programmierung

~~Informelle~~

Physik
Anwesen

4.) Lösung der Schrödingergl. über finite Differenzen für
Exzimer und Trimer

Informelle
II.

5.) Iterationsverfahren: Lanczos Verfahren für Eigenwerte
Krylov-Subraum Verfahren für Eigen

6.) Preconditioning von Matrizen

Physik / Chemie
III.

7.) Basis systeme, Bandstruktur (Moleküle versus
8.) Confined systems, Festkörper)
9.) Dichte funktionelle Theorie

Informelle
=>

10.) Born-Oppenheimer => Optimalitätsprobleme

Physik

11.) Zustände Dynamik: Dichtematrix Theorie,
Blochgleichungen, Gitter

Informelle
Physik
IV.

12.) Beispiele für Zustände: Fermi (adaptive und Adaptive)
13.) Vielteilchen dynamik, Kondensatoren, Anleitschaltung

Informelle
V.

14.) Stufenweise Probleme: Mittelwert Methode, SWFS

Informelle
VI.

15.) Formeln Transformieren DFT / FFT

16.) Herunterladen

|| Weihnachtsaktion auf der halbe des Kanons ||
Arbeiten mit Profiler, Debugger

Computerorientierte Theo. Physik (Vorlesung)

- Physik / Mathematik VII
- 17.) Partielle Dgl der Elektrodynamik: FDTD für Maxwell dynamik
 - 18.) Finite Element Methoden (z.B. Elektrodynamik aber auch Mechanik)
- Mathematik fortgeschritten Quantenrechnen VIII
- 19.) Matrizen, Tensor; Reduktion, Normen, Diagonale
 - 20.) Singulärwertzerlegung, Schmidt-Zerlegung, Verschiebung, H-Tode und MPS
 - 21.) MPS und MPO: Normierung, Kanonische Form und Reduktion
 - 22.) DMRG (Density Matrix Renormalization group)
 - 23.) QTT Quanten Tensor Train Anwendung auf PDE
 - 24.) zeitliche Dynamik von MPS
 - 25.) Vertikale Tensornetzwerke

Am Ende Vorstellung Studierende und Vorstellung weiteren Memento Institut.

[Erklärung Übungsverfahren]

Computer orientierte Quantenmechanik (Vorlesung)

I.2) Rechnerarchitektur

Für eine effektive Programmierung muss man wissen wie Computer gebaut sind. (V. Sehr vereinfachte Darstellung, Fokus auf das für Mensch relevante)

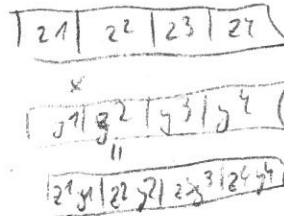
Die CPU

Besteht aus mehreren Kernen (Core).



Jeder Kern besteht aus mehreren Rechnerwerken:

- Fließkammerarchitektur
- SIMD (single ^{stage} input multiple data)
 - (mehrere gleichzeitig Instanzen auf einmal, z.B. bei Vektoroperationen z.B. MMX, SSE, SSE2, ... avx)
 - z.B. 4 Multiplikatoren von denen in Spalten



Optimiert Code für Vektoroperationen (Schwierig, gute Compiler versuchen, oder optimiert BLAS z.B. Lin Algebra)

Ben: Mehrere Einheiten teilen sich CPU Einheiten
 Meist gibt es eine Einheit, die die Assembler Instruktionen decodiert und auf die Einheiten verteilt. (Teilerweiterung)
 (Bei Hyperthreading werden zwei cores simuliert in der die für Einheiten angesetzt sind). Sidenote Sidenote

Wichtige Regel: Zum Programmieren: Finde heraus was in dem Programm am längsten dauert (z.B. mit ein Prof. Tool).

Finde heraus, ob es die Rechenzeit ist. Dann optimiere nur diesen Teil!

Aber meist ist es Speicherzugriff, denn

Speicherzugriff ist langsam.