

# Molecular Computing

## Vorlesung Bioinformatik und Informatik

### Sommersemester 2010

Thomas Hinze

LS Bioinformatik  
Friedrich-Schiller-Universität Jena

thomas.hinze@uni-jena.de  
<http://users.minet.uni-jena.de/~hinze>

---

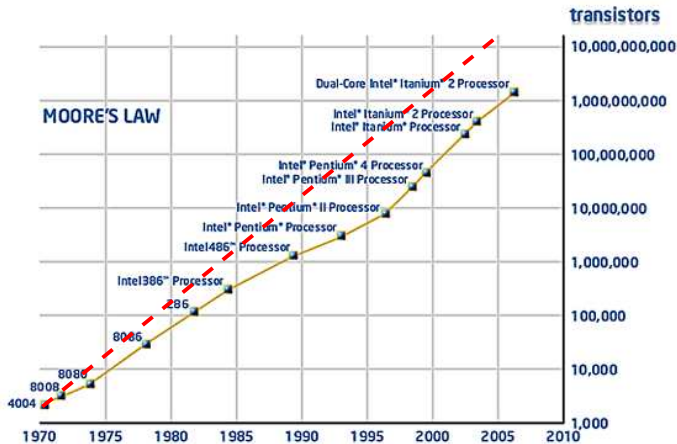
1. Einführung

---



# Moore's Law

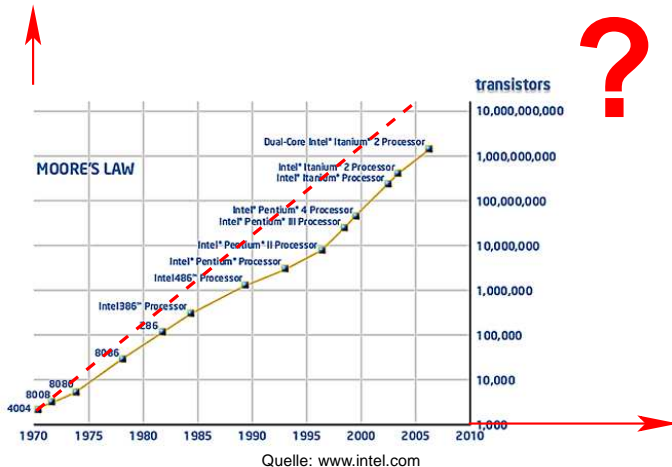
Verdopplung der Rechenleistung alle 18 Monate



Quelle: www.intel.com

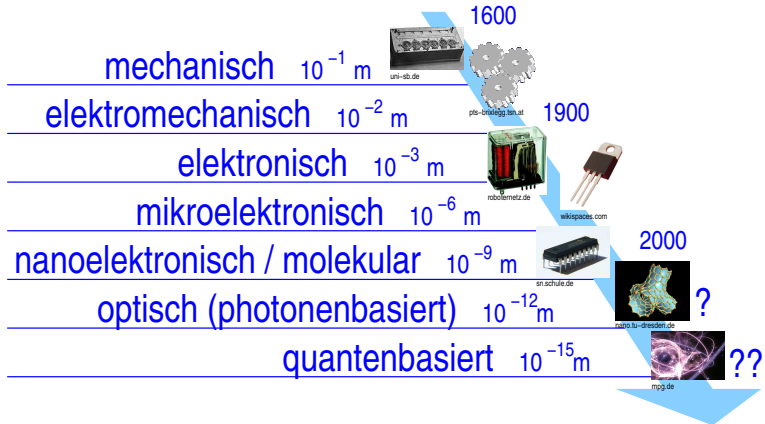
# Moore's Law

Verdopplung der Rechenleistung alle 18 Monate



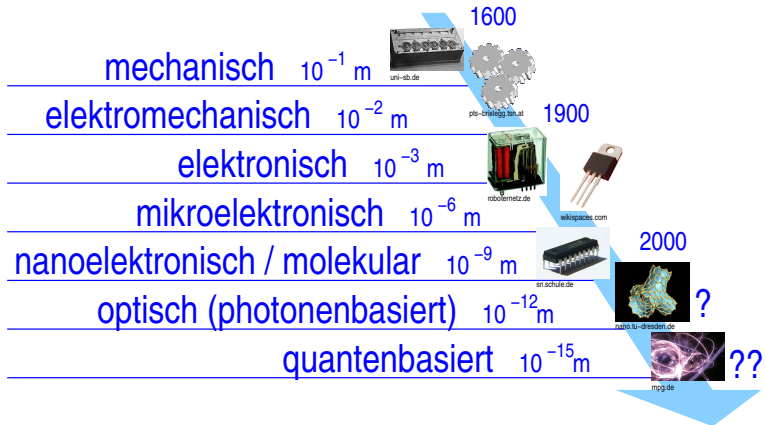
Prognostizierte Verdopplung erst alle 12, dann 18, jetzt 24 Monate ...

# Arbeitsgrundlage für Rechnerarchitekturen



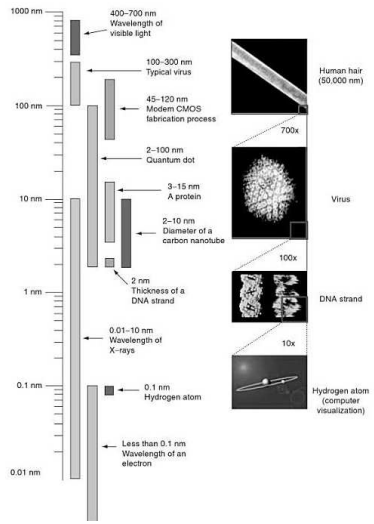
⇒ ingenieurtechnische Evolution

# Arbeitsgrundlage für Rechnerarchitekturen



⇒ **ingenieurtechnische Evolution**

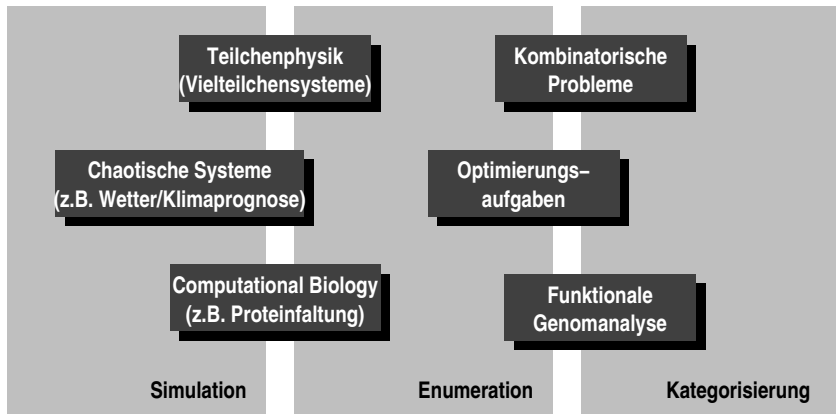
# Willkommen in der Nanowelt: ein Größenvergleich



M. Eshaghian-Wilner. Bio-Inspired and Nanoscale Integrated Computing. John Wiley & Sons, 2009

# Einige rechenintensive Aufgaben

Einsatzgebiete für Supercomputer



Hohe Zeit- bzw. Speicherplatzkomplexität

# Ideen für neuartige Computingkonzepte

## Engineered Computing

- Grid Computing
- Organic Computing
- Pervasive Computing

## Natural Computing

- Amorphous Computing
- Evolutionary Computing
- **Molecular Computing**
- Neural Computing
- Optical Computing
- Quantum Computing

⇒ riesige Speicher, Dezentralisierung, lokale Kommunikation

⇒ nicht zwingend „lebend“



# Ideen für neuartige Computingkonzepte

## Engineered Computing

- Grid Computing
- Organic Computing
- Pervasive Computing

## Natural Computing

- Amorphous Computing
- Evolutionary Computing
- **Molecular Computing**
- Neural Computing
- Optical Computing
- Quantum Computing

⇒ riesige Speicher, Dezentralisierung, lokale Kommunikation

⇒ nicht zwingend „lebend“

# Ideen für neuartige Computingkonzepte

## Engineered Computing

- Grid Computing
- Organic Computing
- Pervasive Computing

## Natural Computing

- Amorphous Computing
- Evolutionary Computing
- **Molecular Computing**
- Neural Computing
- Optical Computing
- Quantum Computing

⇒ riesige Speicher, Dezentralisierung, lokale Kommunikation

⇒ nicht zwingend „lebend“

# Molecular Computing

## Versuch einer begrifflichen Abgrenzung

**Molecular Computing** beschreibt ein Konzept zur Ausführung von Berechnungen, bei dem

- Moleküle als Datenträger/Speichermedium eingesetzt und
- Wechselwirkungen zwischen Molekülen sowie Einwirkungen aus ihrer Umgebung als Rechenschritte aufgefasst werden.

⇒ **Schrittweise Modifikation von Molekülen in einem Pool widerspiegelt Rechengang**

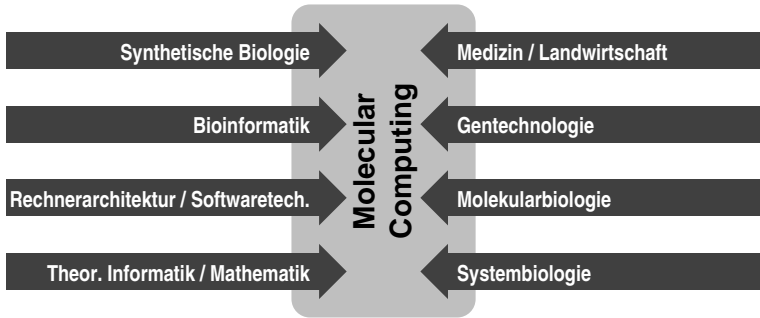
# Was verspricht man sich von Molecular Computing?

## Zukunftsvisionen eines neuen Wissensgebietes

- elementar kleine Speichereinheiten hoher Dichte
- wahlweise assoziative oder adressierbare Datenspeicherung
- massiv parallele Verarbeitung
- hohe Energieeffizienz
- freie Programmierbarkeit mit effektiven Mitteln
- Dezentralität
- Selbstorganisations- und Reparaturfähigkeiten
- Biokompatibilität
- Recyclingfähigkeit von Hardwarekomponenten
- technisches Systemdesign ab initio

# Interdisziplinarität

Enge, vielschichtige Vernetzung mit umgebenden Fachgebieten



# Ausprägungen des Molecular Computing

- Chemical Computing  
(allg. Reaktionssysteme, künstliche Chemien, ...)
- DNA-Computing
- RNA-Computing
- Protein-Computing
- Nanomaschinen

⇒ Grenzen zwischen den Ausprägungen verschwimmen zunehmend, z.B. durch hybride Moleküle (wie PNA) oder in-vivo-Techniken

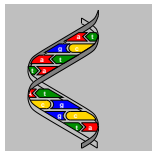
# Ausprägungen des Molecular Computing

- Chemical Computing  
(allg. Reaktionssysteme, künstliche Chemien, ...)
- DNA-Computing
- RNA-Computing
- Protein-Computing
- Nanomaschinen

⇒ Grenzen zwischen den Ausprägungen verschwimmen zunehmend, z.B. durch hybride Moleküle (wie PNA) oder in-vivo-Techniken

## Warum DNA?

- hohe Speicherkapazität und -dichte (bis  $10^{21} \frac{\text{bp}}{\text{l}} \approx 1 \frac{\text{bit}}{\text{nm}^3}$ )
- Langlebigkeit, Persistenz, Duplizierbarkeit
- redundante, dezentrale, verlustsichere Speicherung
- richtungsbehaftet → unterstützt Datenkodierung
- breites Spektrum von in-vitro-Techniken (Synthese, Rekombination, Separation, Analyse)
- Energieeffizienz ( $\approx 2 \cdot 10^9 \frac{\text{ops}}{\text{J}}$ )
- regelmäßige Raumstruktur (drei Konformationen), dadurch leicht analysierbar
- „wohluntersucht“



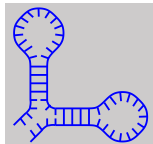


## Warum RNA?

- strukturell sehr altes Biopolymer (vermutlich älter als DNA)
- faltungsfreudiger als DNA (Faltung und Raumstruktur ebenfalls für Datenkodierung nutzbar)
- flexibler als DNA für strukturelle Veränderungen (z.B. Hybridbildung)
- selbstkatalysierend (Ribozyme), dadurch kleinere bzw. selbsterhaltende Reaktionssysteme
- gemeinsame Kodierung von Daten und Funktionen in ein und demselben Molekül
- noch geringfügig energieeffizienter als DNA
- leichter aus präbiotischen Bestandteilen synthetisierbar als DNA

### aber:

kurzlebig (Zerfall nach wenigen Stunden, temperaturempfindlich)

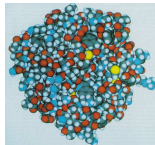


## Warum Proteine?

- großes Repertoire vorhandener und „erprobter“ Proteine
- hohe Spezifität
- kompakte Datenkodierung über Primärstruktur hinaus
- Protein als einsatzfähige molekulare Maschine (Multifunktionsmoleküle)
- Biokompatibilität ohne genetische Veränderungen
- Funktionsvielfalt
- Interagierbarkeit mit Fremdstoffen

### **aber:**

sehr empfindlich, leicht zerstörbar,  
äußerst komplex, Aspekte der Proteomik  
noch nicht vollständig verstanden



# Themenkomplexe

## Gliederung der Lehrveranstaltung

### 1. Biomolekulare Algorithmen in vitro

- **Ausgewählte problemspezifische Biohardware:**  
Adleman-Experiment – SAT – Knapsack – Shapiro
- **Spezielle algorithmische Baustein-Techniken:**  
Hairpin Formation – Whiplash PCR – Tiling – Counting – Immobilisation – Sequenzdesign
- **Vorstufen universeller Biohardware:**  
Rothemunds Turingmaschine – DNA-Splicing – Mikroflussreaktoren – self-assembly – molekulare Maschinen

# Themenkomplexe

## Gliederung der Lehrveranstaltung

### **2. Modelle und Programmiersprachen molekul. Computer**

- Motivation, Einordnung, Grundlegendes
- Filteringmodelle (Adleman, Lipton, Amos)
- DNA-Pascal
- Parallel Associative Memory (PAM)
- Insertion-Deletion-Systeme
- Splicing-Systeme (H-, EH- und Mehrtubesysteme)
- Sticker-Systeme
- künstliche Chemien und P-Systeme

# Themenkomplexe

## Gliederung der Lehrveranstaltung

### 3. Labornahe Simulation molekularer Computer

- Motivation, Einordnung, Simulationsmethoden
- Reaktionsnetzwerke (dynamisches Verhalten)
- Kollisionsmodelle (dynamisches und spatiales Verhalten)
- thermodynamische Ansätze (stochastische Verfahren)
- regelbasierte Ansätze (deterministisch / nichtdeterministisch)
- Beispiele und Softwaredemos:  
Hellics, saces, Sisyphus, SRSim, ...

## Kurzer geschichtlicher Abriss

- Auseinandersetzung mit Komplexität und Grenzen der Berechenbarkeit seit den 1920er Jahren begleitend mit ingenieurtechnischer Entwicklung von Rechentechnik
- „Thermodynamic limits of computing“ (Bennett 1973, Landauer 1982)
- „The fundamental physical limits of computation“ (Bennett, Landauer 1985)
- erste Ideen in der Theorie zu Splicing-Systemen (Head 1987)
- erste molekulare Lösung einer Instanz des Hamiltonkreisproblems (Adleman 1994)
- Blütezeit experimenteller wie theoretischer Arbeiten (etwa 1995 – 2001)
- Fokus auf Nanotechnologie, molekulare Maschinen und Agenten (ab etwa 2002)

## Kurzer geschichtlicher Abriss

- Auseinandersetzung mit Komplexität und Grenzen der Berechenbarkeit seit den 1920er Jahren begleitend mit ingenieurtechnischer Entwicklung von Rechentechnik
- „Thermodynamic limits of computing“ (Bennett 1973, Landauer 1982)
- „The fundamental physical limits of computation“ (Bennett, Landauer 1985)
- erste Ideen in der Theorie zu Splicing-Systemen (Head 1987)
- erste molekulare Lösung einer Instanz des Hamiltonkreisproblems (Adleman 1994)
- Blütezeit experimenteller wie theoretischer Arbeiten (etwa 1995 – 2001)
- Fokus auf Nanotechnologie, molekulare Maschinen und Agenten (ab etwa 2002)

## Kurzer geschichtlicher Abriss

- Auseinandersetzung mit Komplexität und Grenzen der Berechenbarkeit seit den 1920er Jahren begleitend mit ingenieurtechnischer Entwicklung von Rechentechnik
- „Thermodynamic limits of computing“ (Bennett 1973, Landauer 1982)
- „The fundamental physical limits of computation“ (Bennett, Landauer 1985)
- erste Ideen in der Theorie zu Splicing-Systemen (Head 1987)
- erste molekulare Lösung einer Instanz des Hamiltonkreisproblems (Adleman 1994)
- Blütezeit experimenteller wie theoretischer Arbeiten (etwa 1995 – 2001)
- Fokus auf Nanotechnologie, molekulare Maschinen und Agenten (ab etwa 2002)



## Kurzer geschichtlicher Abriss

- Auseinandersetzung mit Komplexität und Grenzen der Berechenbarkeit seit den 1920er Jahren begleitend mit ingenieurtechnischer Entwicklung von Rechentechnik
- „Thermodynamic limits of computing“ (Bennett 1973, Landauer 1982)
- „The fundamental physical limits of computation“ (Bennett, Landauer 1985)
- erste Ideen in der Theorie zu Splicing-Systemen (Head 1987)
- erste molekulare Lösung einer Instanz des Hamiltonkreisproblems (Adleman 1994)
- Blütezeit experimenteller wie theoretischer Arbeiten (etwa 1995 – 2001)
- Fokus auf Nanotechnologie, molekulare Maschinen und Agenten (ab etwa 2002)

## Kurzer geschichtlicher Abriss

- Auseinandersetzung mit Komplexität und Grenzen der Berechenbarkeit seit den 1920er Jahren begleitend mit ingenieurtechnischer Entwicklung von Rechentechnik
- „Thermodynamic limits of computing“ (Bennett 1973, Landauer 1982)
- „The fundamental physical limits of computation“ (Bennett, Landauer 1985)
- erste Ideen in der Theorie zu Splicing-Systemen (Head 1987)
- erste molekulare Lösung einer Instanz des Hamiltonkreisproblems (Adleman 1994)
- Blütezeit experimenteller wie theoretischer Arbeiten (etwa 1995 – 2001)
- Fokus auf Nanotechnologie, molekulare Maschinen und Agenten (ab etwa 2002)

## Kurzer geschichtlicher Abriss

- Auseinandersetzung mit Komplexität und Grenzen der Berechenbarkeit seit den 1920er Jahren begleitend mit ingenieurtechnischer Entwicklung von Rechentechnik
- „Thermodynamic limits of computing“ (Bennett 1973, Landauer 1982)
- „The fundamental physical limits of computation“ (Bennett, Landauer 1985)
- erste Ideen in der Theorie zu Splicing-Systemen (Head 1987)
- erste molekulare Lösung einer Instanz des Hamiltonkreisproblems (Adleman 1994)
- Blütezeit experimenteller wie theoretischer Arbeiten (etwa 1995 – 2001)
- Fokus auf Nanotechnologie, molekulare Maschinen und Agenten (ab etwa 2002)

## Kurzer geschichtlicher Abriss

- Auseinandersetzung mit Komplexität und Grenzen der Berechenbarkeit seit den 1920er Jahren begleitend mit ingenieurtechnischer Entwicklung von Rechentechnik
- „Thermodynamic limits of computing“ (Bennett 1973, Landauer 1982)
- „The fundamental physical limits of computation“ (Bennett, Landauer 1985)
- erste Ideen in der Theorie zu Splicing-Systemen (Head 1987)
- erste molekulare Lösung einer Instanz des Hamiltonkreisproblems (Adleman 1994)
- Blütezeit experimenteller wie theoretischer Arbeiten (etwa 1995 – 2001)
- Fokus auf Nanotechnologie, molekulare Maschinen und Agenten (ab etwa 2002)

## Organisation der Lehrveranstaltung

- insgesamt 14 Veranstaltungen (7V, 4Ü, 3S) für drei Themenkomplexe
- Vorlesung und Übung im Wechsel (flexibel)
- Vorlesungsmaterial, Übungsaufgaben und Zusatzinfos regelmäßig im CAJ bereitgestellt
- für Themenkomplexe 2 und 3 zusätzliches Skript in Papierform

### **für Modulprüfung im Modul D1**

#### **(3LP+Note bei Wahlpflichtfach, 3LP bei Wahlfach):**

1. aktive Mitarbeit in Vorlesung und Übung
  2. Präsentation eines Fachartikels zur Thematik
- etwa 20 min pro Vortrag, dafür die letzten drei Veranstaltungen reserviert (22.06., 29.06. und 06.07.10)
  - Auswahl geeigneter Fachartikel (Vorschläge) bereitgestellt

## Weiterführende Literatur (Auswahl)

- ausgewählte Fachartikel
- **M. Amos.** *Theoretical and Experimental DNA Computation.* Springer, 2005
- **C. Calude, G. Păun.** *Computing with Cells and Atoms.* Taylor and Francis, 2001
- **L.N. de Castro.** *Fundamentals of Natural Computing: Basic Concepts, Algorithms, and Applications.* Taylor and Francis, 2006
- **M. Gheorghe.** *Molecular Computational Models: Unconventional Approaches.* IGI Global, 2005
- **T. Hinze, M. Sturm.** *Rechnen mit DNA - Eine Einführung in Theorie und Praxis.* Oldenbourg, 2004
- **Z. Ignatova, I. Martinez-Perez, K.H. Zimmermann.** *DNA Computing Models.* Springer, 2008
- **N. Jonoska, G. Păun, G. Rozenberg (Eds.).** *Aspects of Molecular Computing.* Springer, 2004
- **N. Krasnogor, S. Gustafson, D.A. Pelta, J.L. Verdegay (Eds.).** *Systems Self-Assembly: Multidisciplinary Snapshots.* Elsevier, 2008
- **G. Păun.** *Computing with Bio-Molecules: Theory and Experiments.* Springer, 1998
- **G. Păun, G. Rozenberg, A. Salomaa.** *DNA Computing.* Springer, 1998
- **T. Sienko, A. Adamatzky, N. Rambidi, M. Conrad.** *Molecular Computing.* MIT Press, 2003