Prof. Dr. rer. nat. habil. Martin O. Steinhauser

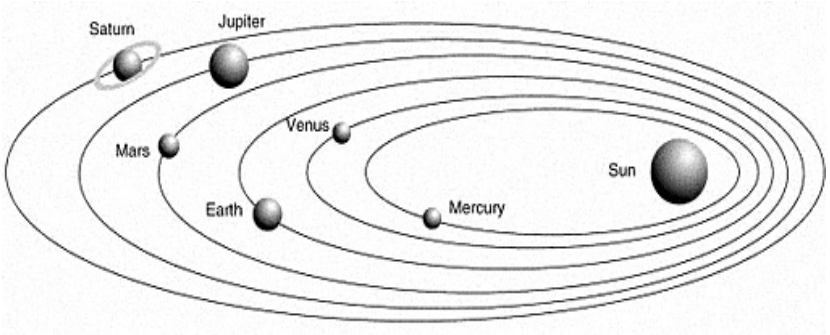
Frankfurt University of Applied Sciences, Germany

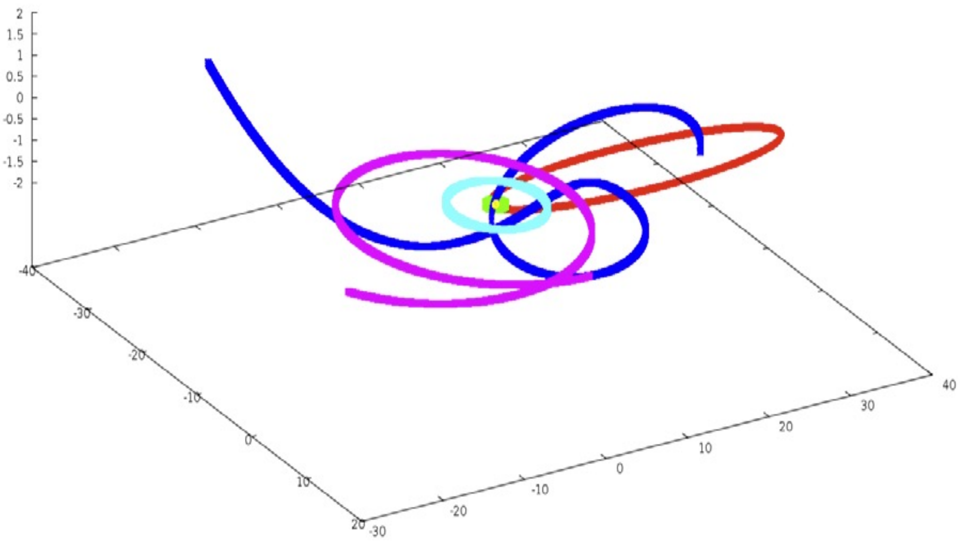
Faculty of Computer Science and Engineering

Short Lecture Course:

Introduction to Computational Science with Applications in Molecular Dynamics

Session 3: The Molecular Dynamics Method





Overview of this short course

Topics Covered (subject to change)

- 1st Session: Lec. 1-2 — Introduction & Bits and Bytes
- 2nd Session: Lec 3 (2x) — Bits and Bytes continued
- 3rd Session: Lec 4-6 — Molecular Dynamics (MD)
- 4th Session: Lec 7-8 — MD continued / Algorithms
- 5th Session: Lec 9 (2x) — Algorithms/ Problem of Sorting
- 6th Session: Lec 10-11 — Problem of Sorting / Monte Carlo
- 8th Session: Lec 12-13 — Monte Carlo/Random Numbers

Session 3: Overview

OUTLINE OF LECTURE

- What is the MD method?
- Newtonian/Lagrangian/Hamiltonian Dynamics
- A Molecular Dynamics Program: Planetary Motion
- **Handout 4:** Introduction to Molecular Dynamics Simulations

  (Original Publication by M. O. Steinhauser)
- **Handout 5:** C-Code: PMC.zip

To download lecture material, please go to Github:

https://github.com/Kosmokrat/JapanLecture2024

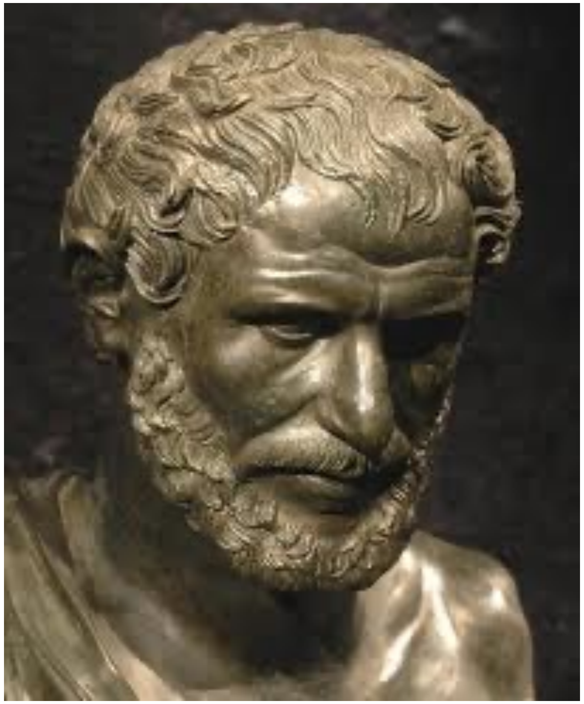
Session 3

1	The Molecular Dynamics Method

What is Molecular Dynamics?

You cannot step twice in the same river

Heraclitus



What is Molecular Dynamics?

PHYSICAL REVIEW VOLUME 136, NUMBER 2A 19 OCTOBER 1964

Correlations in the Motion of Atoms in Liquid Argon*

A. Rahman
Argonne National Laboratory, Argonne, Illinois
(Received 6 May 1964)

A system of 864 particles interacting with a Lennard-Jones potential and obeying classical equations of motion has been studied on a digital computer (CDC 3600) to simulate molecular dynamics in liquid argon at 94.4°K and a density of 1.374 g cm^{-3}. The pair-correlation function and the constant of self-diffusion are found to agree well with experiment; the latter is 15% lower than the experimental value. The spectrum of the velocity autocorrelation function shows a broad maximum in the frequency region $\omega=0.25(k_BT/\hbar)$. The shape of the Van Hove function $G_s(r,t)$ attains a maximum departure from a Gaussian at about $t=3.0\times10^{-12}$ sec and becomes a Gaussian again at about 10^{-11} sec. The Van Hove function $G_d(r,t)$ has been compared with the convolution approximation of Vineyard, showing that this approximation gives a too rapid decay of $G_d(r,t)$ with time. A delayed-convolution approximation has been suggested which gives a better fit with $G_d(r,t)$; this delayed convolution makes $G_d(r,t)$ decay as t^4 at short times and as t at long times.

- MD was *first introduced* for the study of liquids for which an analytical theory is extremely difficult to be formulated
  (1964 landmark paper by Rahman)

Remark:

- Solid states are *periodic* and thus simpler in the theoretical treatment than liquids. For liquids, long-range disorder is an essential part of the system.

Some History of MD

- Rahman (Argonne 1964): liquid Argon with 864 particles

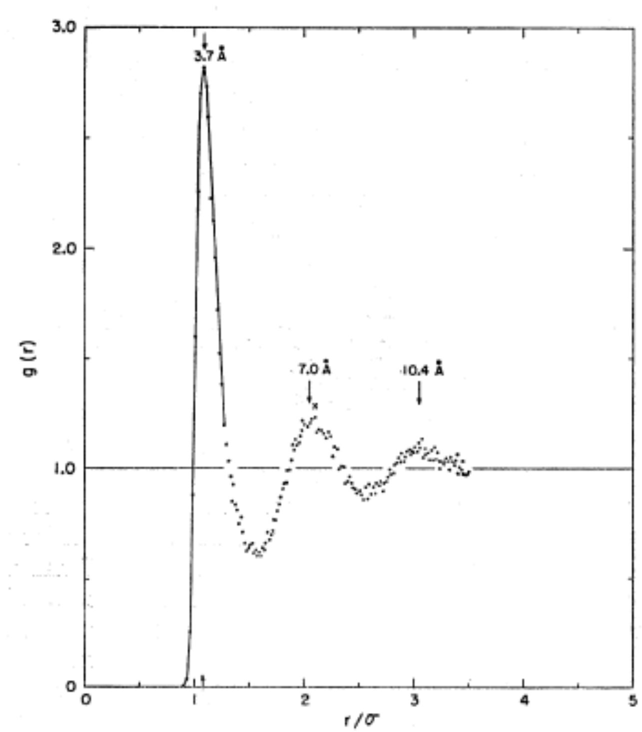


FIG. 2. Pair-correlation function obtained in this calculation at 94.4°K and 1.374 gcm^{-3}. The Fourier transform of this function has peaks at $\kappa\sigma = 6.8$, 12.5, 18.5, 24.8.

Figure 2 in A. Rahman, „Correlations in the Motion of Atoms in Liquid Argon“, *Physical Review* **136**, A405-A411, 1964

Some History of MD

- Alder and Wainwright (@Livermore, USA, 1956)
  - Dynamics of hard spheres

Phase Transition for a Hard Sphere System

B. J. ALDER AND T. E. WAINWRIGHT
University of California Radiation Laboratory, Livermore, California
(Received August 12, 1957)

A CALCULATION of molecular dynamic motion has been designed principally to study the relaxations accompanying various nonequilibrium phenomena. The method consists of solving exactly (to the number of significant figures carried) the simultaneous classical equations of motion of several hundred particles by means of fast electronic computors. Some of the details as they relate to hard spheres and to particles having square well potentials of attraction have been described.[1,2] The method has been used also to calculate equilibrium properties, particularly the equation of state of hard spheres where differences with previous Monte Carlo[3] results appeared.

Very FIRST MD publication:

J. Chem. Phys. 27, 1208 (1957)

A 32 particle system calculated on a UNIVAC machine

Some History of MD

- Vineyard et al. (Brookhaven 1959-1960): Radiation damage in copper

First example of atomistic modeling of materials

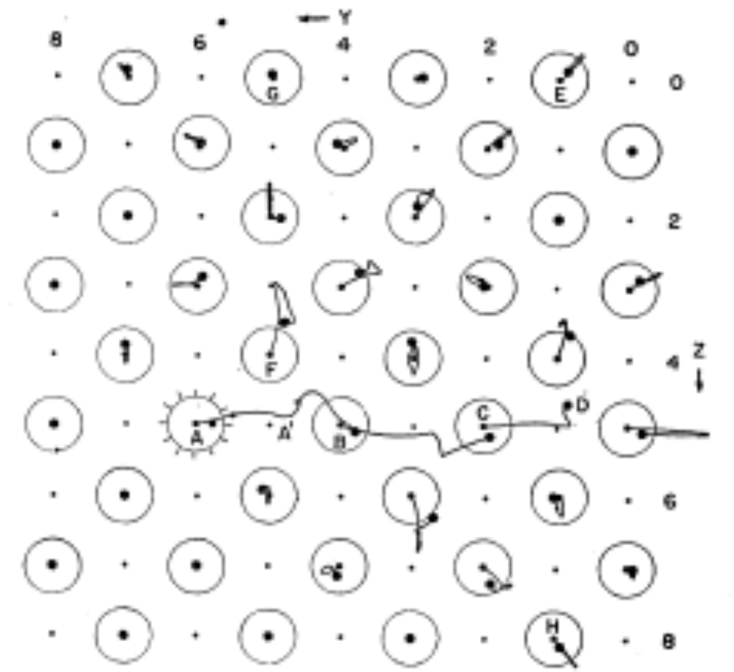


Fig. 6. Atomic orbits produced by shot in (100) plane at 40 ev. Knock-on was at *A* and was directed 15° above $-y$ axis. Large circles give initial positions of atoms in plane; small dots are initial positions in plane below. Vacancy is created at *A*, split interstitial at *D*. Run to time 99. (Run No. 12).

Figure 6 in J.B. Gibson, A.N. Goland and G.H. Vineyard, „Dynamics of Radiation Damage“, *Physical Review* **120**, 1229-1253, 1960

A Historical Perspective of MD

- **1500-1600s**: L. da Vinci, Galileo Galilei

  1700-1800: Euler, Bernoulli
  Beam theories, rods (partial differential equations, continuum theories)
- **Continuum mechanics** theories
- Development of theories of fracture mechanics, **theory of dislocations** (**1930s**)
- **1960..70s**: Development of **FE theories** and methods (engineers)
- **1990s**: Marriage of MD and FE via **Quasicontinuum Method** (Ortiz, Tadmor, Phillips) and others

Continuum

Atomistic

- **20th century:** Atoms discovered (Jean Perrin)
- **MD:** First introduced by Alder and Wainwright in the late 1950's (interactions of hard spheres). Many important insights concerning the behavior of simple liquids emerged from their studies.
- **1964**, when Rahman carried out the first simulation using a realistic potential for liquid argon (Rahman, 1964).
- Numerical methods like DFT (Kohn-Sham, **1960s-80s**)
- First molecular dynamics simulation of a realistic system was done by Rahman and Stillinger in their simulation of liquid water in **1974** (Stillinger and Rahman, 1974).

- **Now:** MD simulations of biophysics problems, fracture, deformation are routine
- The number of simulation techniques has greatly expanded: Many specialized techniques for particular problems, including mixed quantum mechanical - classical simulations, that are being employed to study enzymatic reactions (“QM-MM”) or fracture simulations

Some History of MD

- Car and Parinello (1985): ab-initio MD
  - Extends the Lagrangian of a particle system by introducing the explicit degrees of freedom of the electrons as dynamic variables, which leads to a system of coupled EOM for the electrons and the nuclei.

VOLUME 55, NUMBER 22 PHYSICAL REVIEW LETTERS 25 NOVEMBER 1985

Unified Approach for Molecular Dynamics and Density-Functional Theory

R. Car

International School for Advanced Studies, Trieste, Italy

and

M. Parrinello

Dipartimento di Fisica Teorica, Università di Trieste, Trieste, Italy, and
International School for Advanced Studies, Trieste, Italy

(Received 5 August 1985)

We present a unified scheme that, by combining molecular dynamics and density-functional theory, profoundly extends the range of both concepts. Our approach extends molecular dynamics beyond the usual pair-potential approximation, thereby making possible the simulation of both covalently bonded and metallic systems. In addition it permits the application of density-functional theory to much larger systems than previously feasible. The new technique is demonstrated by the calculation of some static and dynamic properties of crystalline silicon within a self-consistent pseudopotential framework.

PACS numbers: 71.10.+x, 65.50.+m, 71.45.Gm

R. Car and M. Parinello, *Phys Rev. Lett.*, **55**, 2471 (1995)

An Operational Definition of MD

- We follow the evolution of a system composed of many classical particles
- Each particle interacts simultaneously with every other particle (but can also have „hard spheres“ interaction), and can experience additional external potential.
- It‘s a *many-body* problem

Not solvable analytically for $N > 2$ (proved by Poincaré)

Newton‘s Equations and Laplace‘s Demon

- Follow the dynamics (the motion) of all the atoms in your material
- Numerically solve classical equations of motion (Newton)

$$m_i \frac{d^2\vec{r}}{dt^2} = \vec{F}_i\left(\vec{r}_1,\cdots,\vec{r}_N\right)$$

LAPLACE:

Nous devons donc envisager l'état présent de l'universe comme l'effet de son état antérieur et comme la cause de delui qui va suivre. Une intelligence qui, pour un instant donné, connaîtrait toutes les forces dont la nature est animée et la situation respective des êtres qui las composent, si d'ailleurs elle était assez vaste pour soumettre ces données à l'Analyse, embrasserait dans la même formule les mouvements des plus grands corps de l'univers et ceux du plus lèger atome : rien ne serait incertain pour elle, et l'avenir, comme le passé, serait présent à ses yeux.

Calculating the Forces of Particles (often: Atoms)

- Forces on the atoms come from the interaction with other atoms:

Total potential energy
(from QM or interatomic potentials)

Approximated
(in almost all cases)

$$\vec{F}\left(\vec{r}_1, \cdots \vec{r}_N\right) = -\vec{\nabla}_{\vec{r}_i} \Phi\left(\{\vec{r}_j\}\right)$$

Solve Differential Equations:

- One needs *initial conditions* and the EOMs
- An absolute *deterministic view* of the physical world

Review of Classical Mechanics: Newton

- Different formulations of classical mechanics:
- **Newton:** Direct description of a mechanical system in position space
- **Equations of Motion:** $\vec{F}_i = \sum_{i=1}^{N} m_i \ddot{\vec{r}}_i = \sum_{i=1}^{N} \dot{\vec{p}}_i = -\vec{\nabla}_{\vec{r}_i} \phi\left(\left\{\vec{r}_i\right\}\right)$

Remark:

- If a potential exists, the system is called *conservative:* $\oint_{curve} \vec{F}\, \mathrm{d}\vec{s} = 0$
- Important, because here, the total energy is *conserved:*

$$E = \sum_{i=1}^{N} 1/2\; m_i \vec{v}_i^{\,2} + \phi\left(\vec{r}_i\right) \Rightarrow \frac{dE}{dt} = \sum_{i=1}^{N} m_i \vec{v}_i \dot{\vec{v}}_i + \frac{d\phi}{dt}$$

$$= \sum_{i=1}^{N} m_i \dot{\vec{v}}_i \vec{v}_i - \vec{F}_i \vec{v}_i = 0, \text{ because } \frac{d\phi}{dt} = \frac{d\phi}{d\vec{r}_i} \frac{d\vec{r}_i}{dt} = F_i \vec{v}_i$$

Review of Classical Mechanics: Lagrange

- **Lagrange:** There exists a function $L = L(\dot{q}_i, q_i, t)$

  for which the following *variational principle* holds:

$$I = \int_{t_0}^{t} L(\dot{q}_i, q_i, t)\, dt = 0$$

- **Equations of Motion: Lagrange Equations of the 2nd kind:**

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{q}_i} - \frac{\partial L}{\partial q_i} = 0$$

Remark: $L = L(\dot{q}_i, q_i, t) = E_{kin} - E_{pot} = K(\dot{q}_i, q_i, t) - \phi(q_i)$

- Advantage of Lagrange formulation: L can be formulated in *any system* using generalized velocities $\dot{q}_i$ and coordinates q_i .

Review of Classical Mechanics: Hamilton

- Reformulation of classical mechanics (1800‘s):

- **Hamiltonian:** Description of a mechanical system in 6N-dim. phase space

  There exists a function $H(p_i, q_i, t) = \sum_{i=1}^{3N} p_i q_i - L(\dot{q}_i, q_i, t)$

  where $p_i = \dfrac{\partial L}{\partial \dot{q}_i}$ is the *generalized momentum*

- **Equations of Motion** are the *canonical* EOM:

$$\dot{q}_i = \frac{\partial H}{\partial p_i} \qquad \dot{p}_i = -\frac{\partial H}{\partial q_i} \qquad i = 1, \ldots, 3N$$

Remark:

- $H(p_i, q_i, t)$ and $L(\dot{q}_i, q_i, t)$ are connected via a

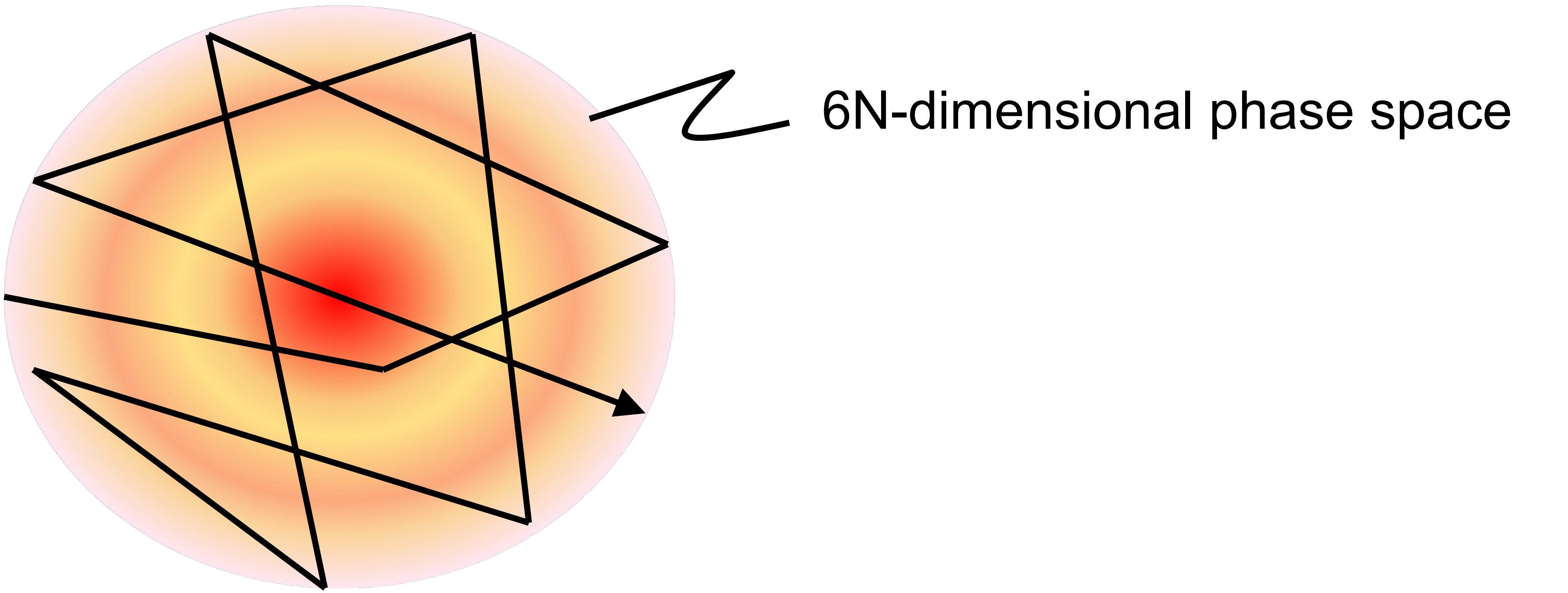
  Legendre-Transformation

Description of Classical Systems: Phase Space

- When we have N particles, we need to specify positions and velocities for all of them (6N variables) to uniquely identify the dynamical system.

- **One point** in a **6N-dimensional space** (the phase space) represents our complete dynamical system.

Ergodicity Hypothesis



All parts of phase space are eventually touched:
Time-average = Ensemble Average

- There is **no general proof** for ergodicity!

Thermodynamic Averages

$$\left\langle A \right\rangle = \frac{\int A \exp\left(-\beta E\right) dr^{3N} dp^{3N}}{\int \exp\left(-\beta E\right) dr^{3N} dp^{3N}} \Leftrightarrow \overline{A} = \frac{1}{T}\int_0^T A\left(t\right)$$

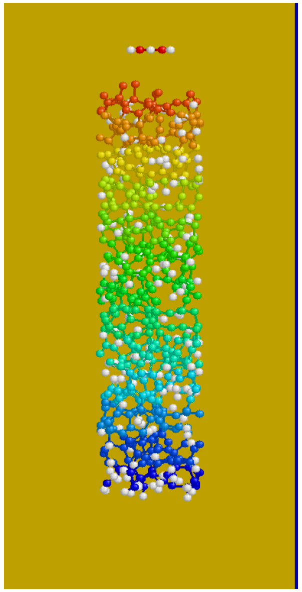
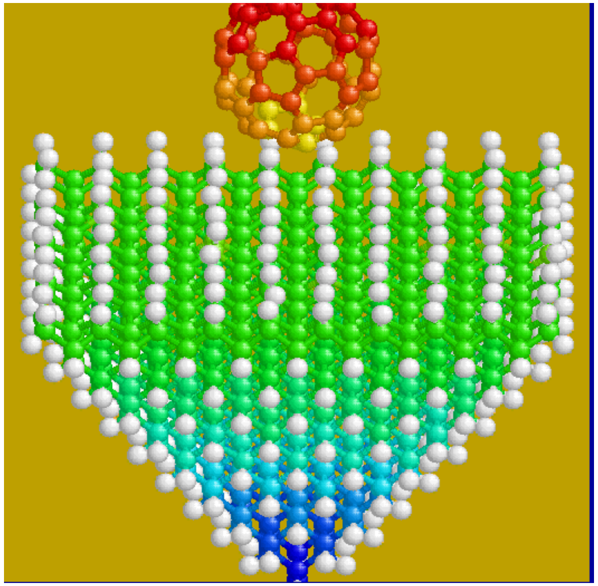
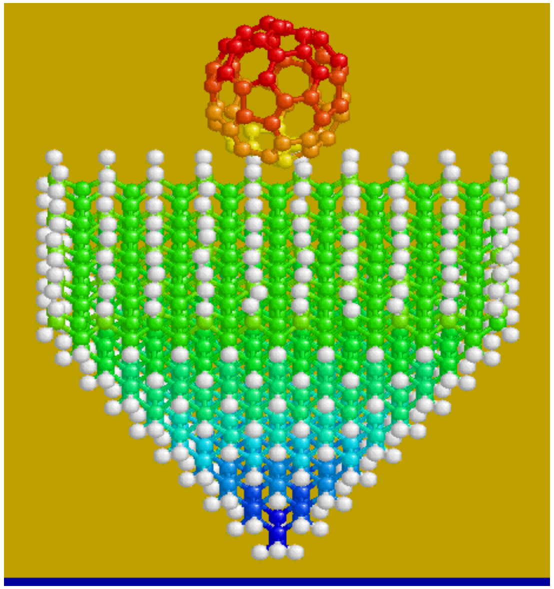
$$\overline{A} = \frac{1}{T}\int_0^T A\left(t\right)$$

Ergodic Hypothesis

Three Main Goals of MD

- Calculate Ensemble Averages (thermodynamics)
  - For example a Microcanonical Molecular Fluid

- Study real-time evolution (e.g. chemistry)

Three Main Goals of MD

- Calculate Ensemble Averages (thermodynamics)
  - For example a Microcanonical Molecular Fluid
- Study real-time evolution (e.g. chemistry)
  - Classical Example: Planetary Motion
- Groud-state optimization of complex structures
  - Usually in atomistic QM-Simulations of VERY SIMPLE systems

Limitations of MD Simulations

- Time Scales
- Length Scales (PBC can help)
- Accuracy of forces (neglection of electron movement)
- Nuclei modeled as classical particles

Basic MD Integration Algorithm

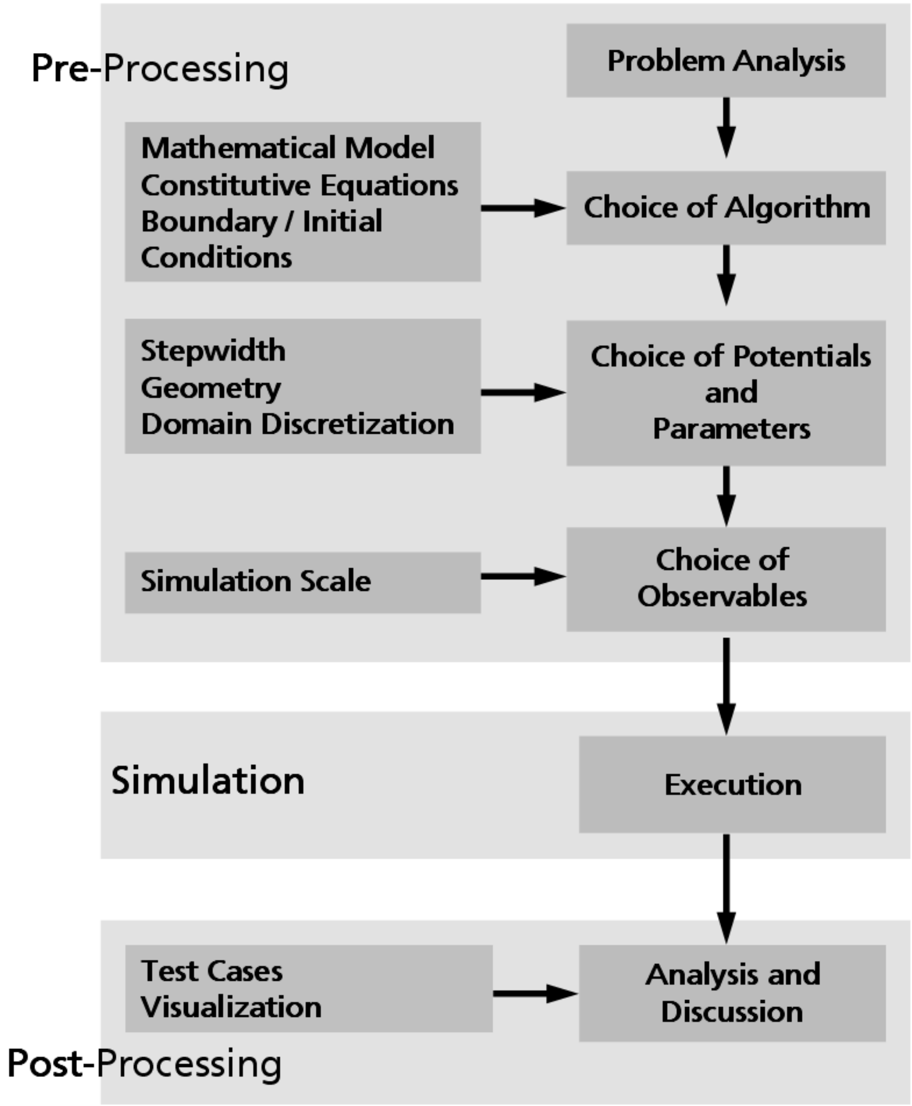
Algorithm 1.1 Basic Algorithm

```
real t = t_start;
for i = 1, ..., N
  set initial conditions x_i (positions) and v_i (velocities);
while (t < t_end) {
  compute for i = 1, ..., N the new positions x_i and velocities v_i
    at time t + delta_t by an integration procedure from the
    positions x_i, velocities v_i and forces F_i on the particle at
    earlier times;
  t = t + delta_t;
}
```

The Computational MD Experiment

- **Initialize:** select positions and velocities
- **Integrate:** compute all forces, and determine new positions
- **Equilibrate:** let the system reach equilibrium
  (i.e. lose memory of initial conditions)
- **Average:** accumulate quantities of interest

Scheme of any Computer Simulation



Taken from:
M. O. Steinhauser: Multiscale Modeling of Fluids and Solids – Theory and Applications, Springer, 2nd edition, Berlin, Heidelberg, Boston, 2008

Integration: Many Variants of MD According to the Ensemble

- Use an integrator (Verlet, leapfrog, velocity verlet, Gear-predictor-corrector…)

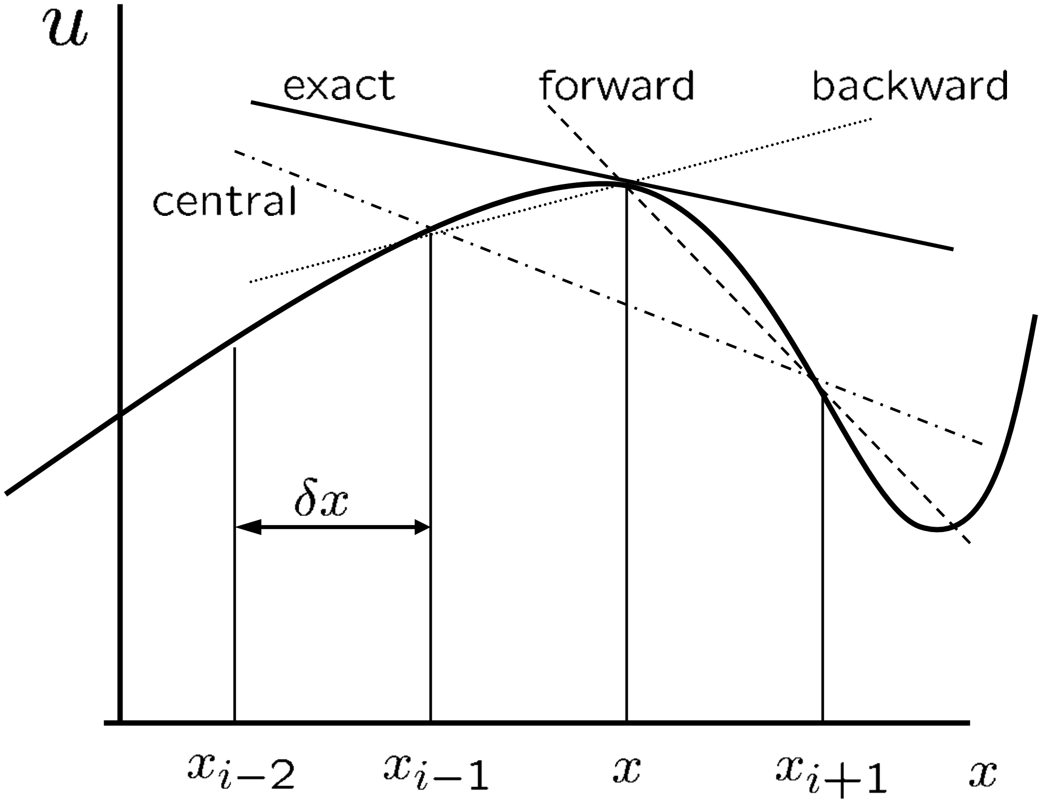
- **Robust**, long-term **conservation of the constants of motions**, **time-reversible**, constant volume in phase space

- Choose the desired **thermodynamic ensemble** (microcanonical NVE, or canonical NVT using a thermostat, isobaric-isothermic NOT with a barostat,…)

- **Stochastic** (Langevin), **constrained** (velocity re-scaling,…), **extended system** (Nosé-Hover)

Spatial and Temporal Discretization

- Numerical integration
  - Forward difference
  - Backward difference
  - Central difference



Molecular Dynamics Solves the N-Body Problem
Naive Approach: Taylor Expansion

Classical N-body initial value problem:

Can only be solved numerically (except in very special cases)

How?

$$X(t+\Delta t) = X(t) + \dot{X}(t)\Delta t + \frac{1}{2!}\ddot{X}\left(t\right)\Delta t^2 + \frac{1}{3!}\dddot{X}\left(t\right)\Delta t^3 + \cdots$$

Molecular Dynamics Solves the N-Body Problem
Naive Approach: Taylor Expansion

Classical N-body initial value problem:

Can only be solved numerically (except in very special cases)

How?

$$X(t+\Delta t) = X(t) + \dot{X}(t)\Delta t + \frac{1}{2!}\ddot{X}\left(t\right)\Delta t^2 + \frac{1}{3!}\dddot{X}\left(t\right)\Delta t^3 + \cdots$$

Solving the N-Body Problem
Naive Approach: Taylor Expansion

Classical N-body initial value problem:

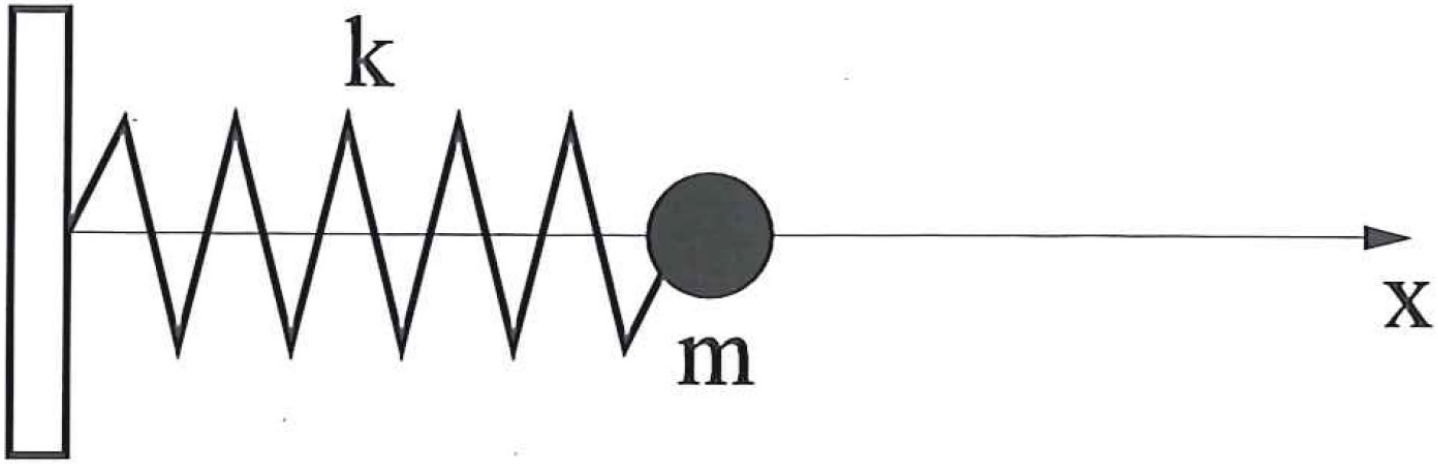
Can only be solved numerically (except in very special cases)

How? Truncate the Taylor Expansion

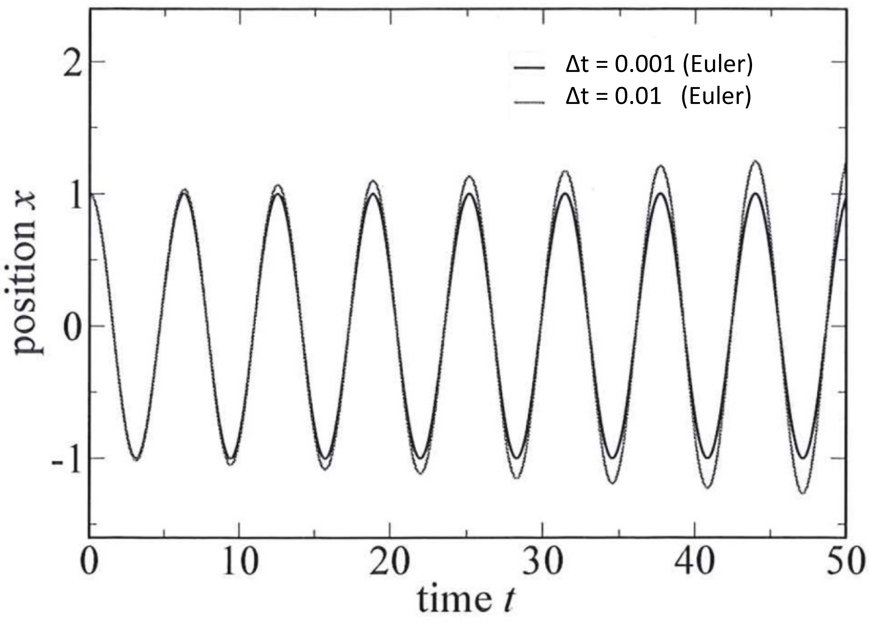
$$\xcancel{X(t+\Delta t) = X(t) + \dot{X}(t)\Delta t + \frac{1}{2!}\ddot{X}(t)\Delta t^2}$$

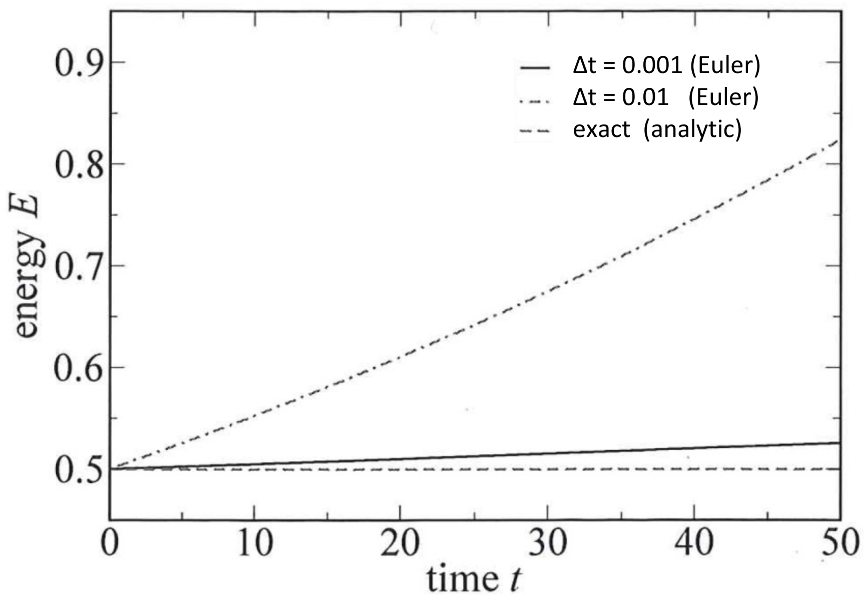
Absolutely Forbidden!

Simple Example: 1D Harmonic Oscillator



Simple Example: 1D Harmonic Oscillator





Forward Euler Method:

- Is not time reversible
- Does not conserve volume in phase space
- Suffers from energy drift

Solving the N-Body Problem
Naive Approach: Taylor Expansion (Forward Euler Method)

Forward Euler Method:

- Is not time reversible
- Does not conserve volume in phase space
- Suffers from energy drift

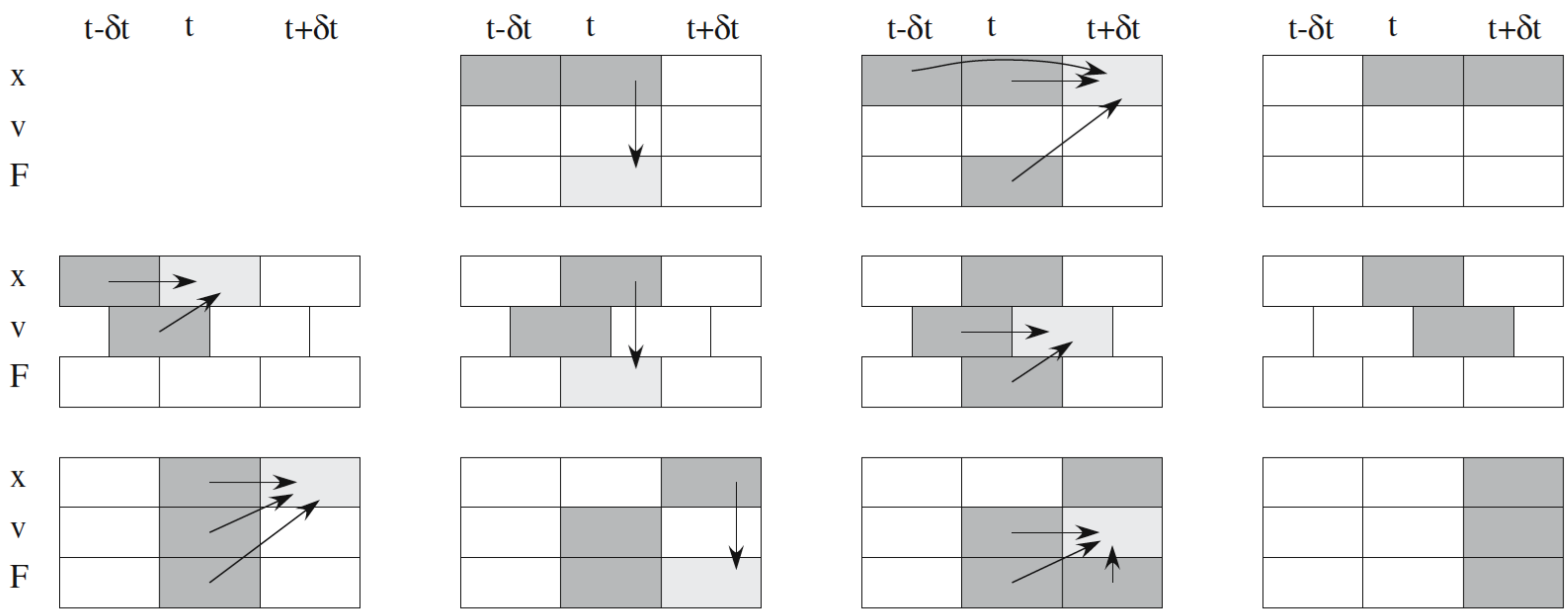
Absolutely NOT !

The Standard Velocity Verlet Algorithm

Algorithm 1.2 Velocity-Störmer-Verlet Method

```
// start with initial data x, v, t
// auxiliary vector F^old;
compute forces F;
while (t < t_end) {
  t = t + delta_t;
  loop over all i {                                    // update x
    x_i = x_i + delta_t * (v_i + .5 / m_i * F_i * delta_t);  // using (6*)
    F_i^old = F_i;
  }
  compute forces F;
  loop over all i                                      // update v
    v_i = v_i + delta_t * .5 / m_i * (F_i + F_i^old);  // using (7*)
  compute derived quantities as for example kinetic or potential energy;
  print values of t, x, v as well as derived quantities;
}
```

Different Schemes of The Verlet Algorithm



- Top: Standard Verlet Scheme
- Middle: Leapfrog Scheme
- Bottom: Velocity Verlet

Example: Planetary Motion Code (PMC)

Data structure Particle

```
typedef struct {
  real m;             // mass
  real x[DIM];        // position
  real v[DIM];        // velocity
  real F[DIM];        // force
} Particle;
```

Algorithm Velocity-Störmer-Verlet Method

```
void timeIntegration_basis(real t, real delta_t, real t_end,
                           Particle *p, int N) {
  compF_basis(p, N);
  while (t < t_end) {
    t += delta_t;
    compX_basis(p, N, delta_t);
    compF_basis(p, N);
    compV_basis(p, N, delta_t);
    compoutStatistic_basis(p, N, t);
    outputResults_basis(p, N, t);
  }
}
```

Example: Planetary Motion Code (PMC)

Algorithm _._ Routines for the Velocity-Störmer-Verlet Time Step for a Vector of Particles

```
void compX_basis(Particle *p, int N, real delta_t) {
  for (int i=0; i<N; i++)
    updateX(&p[i], delta_t);
}
void compV_basis(Particle *p, int N, real delta_t) {
  for (int i=0; i<N; i++)
    updateV(&p[i], delta_t);
}
```

Algorithm Computation of the Force with $\mathcal{O}(N^2)$ Operations

```
void compF_basis(Particle *p, int N) {
  for (int i=0; i<N; i++)
    for (int d=0; d<DIM; d++)
      p[i].F[d] = 0;                 // set F for all particles to zero
  for (int i=0; i<N; i++)
    for (int j=0; j<N; j++)
      if (i != j) force(&p[i], &p[j]);     // add the forces F_ij to F_i
}
```

Example: Planetary Motion Code (PMC)

Algorithm Gravitational Force between two Particles

```
void force(Particle *i, Particle *j) {
  real r = 0;
  for (int d=0; d<DIM; d++)
    r += sqr(j->x[d] - i->x[d]);                  // squared distance r=r_ij^2
  real f = i->m * j->m /(sqrt(r) * r);
  for (int d=0; d<DIM; d++)
    i->F[d] += f * (j->x[d] - i->x[d]);
}
```

Code fragment Allocate and Free Memory Dynamically

```
Particle *p = (Particle*)malloc(N * sizeof(*p));     // reserve
free(p);                                             // and release memory
```

Example: Planetary Motion Code (PMC)

Algorithm Main Program

```
int main() {
  int N;
  real delta_t, t_end;
  inputParameters_basis(&delta_t, &t_end, &N);
  Particle *p = (Particle*)malloc(N * sizeof(*p));
  initData_basis(p, N);
  timeIntegration_basis(0, delta_t, t_end, p, N);
  free(p);
  return 0;
}
```

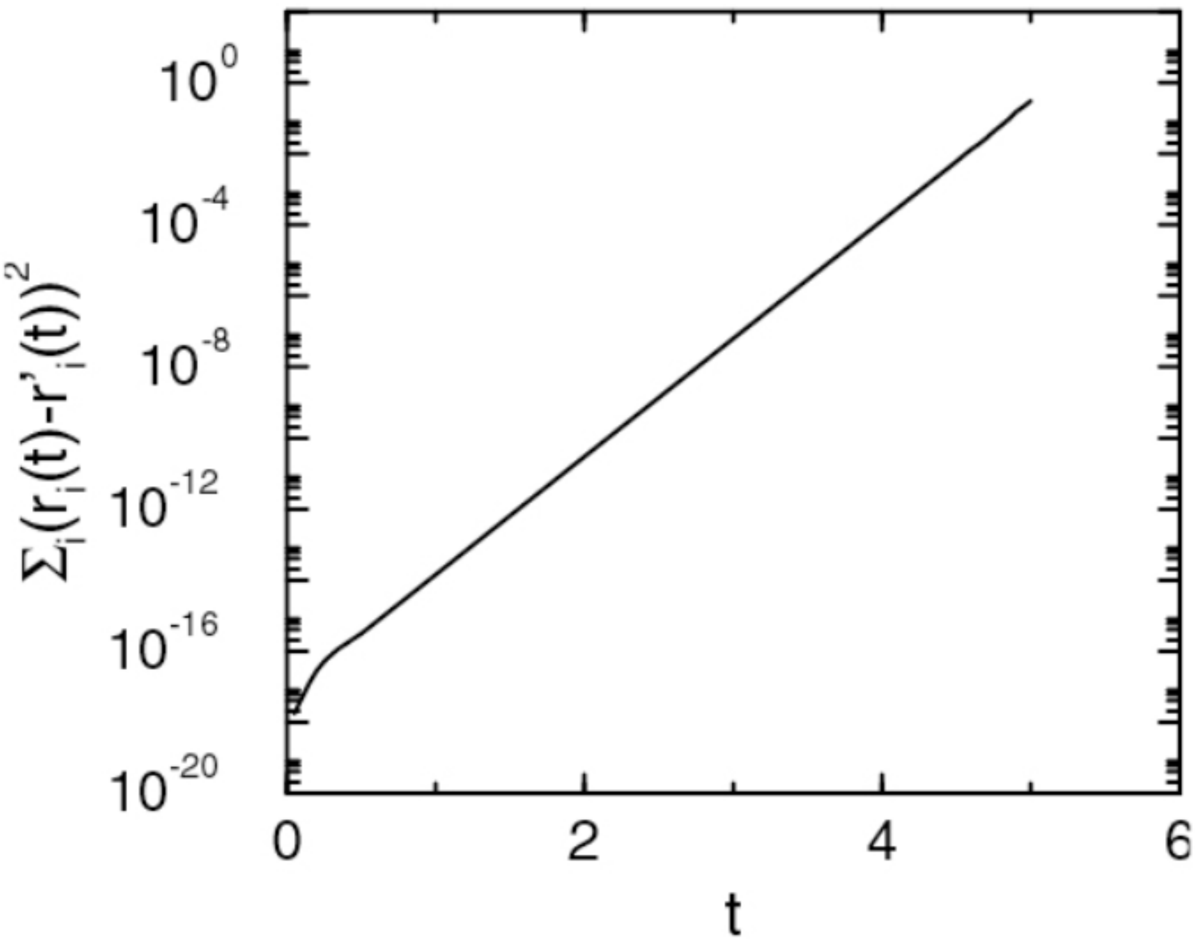
Lyapunov Instabilities

- The dynamics of a well-behaved classical many body (N particle) system is **chaotic!**

- Consequence: Trajectories of particles that differ very slightly in their initial conditions, **diverge exponentially ! (Lyapunov Instability)**

Lyapunov Instabilities

- The Lyapunov desaster in action…



Lyapunov Instabilities

- Any small error in the numerical integration of the equations of motion **will blow up exponentially…**

  always…

  and… for *any* algorithm !

So…

Why should anyone believe in Molecular Dynamics Simulation?

What is the point of simulating dynamics if we cannot trust the resulting time-evolution?

Answer: We‘re interested in *Statistical* Properties

Here, everything works out fine!

Analysis and Interpretation of MD

Relate **microscopic** phenomena simulated with the MD method and **macroscopic** properties:

Given a thermodynamic state of a material, what are the probabilities of finding the system in the various possible microscopic states?

Or: Given a series of microscopic states, what is the corresponding macroscopic state?

→ To answer this question, we need Statistical Mechanics !

Live Demo

My University Research Page: https://www.frankfurt-university.de/steinhauser

Contact Me: martin.steinhauser@fb2.fra-uas.de

Research Gate: https://www.researchgate.net/profile/Martin-Steinhauser