

Computational Science

Molecule Dynamic simulation

Task:

Implementation of an MD-simulation for Lennard-Jones particles/ ions/ water in 2D and 3D. →interaction between the particles an thereby causes spatial motion by iterativ calculations

Computational Science	1
Task:	1
Concept:	3
Workflow	4
So, 13.01.19	4
Selbstständige Arbeitsphase	4
Sa, 19.01.19	4
Arbeitsphase	4
So, 27.01.19	5
Overview:	6
Information about modules	8
Data storage module	8
Sampling module	8
Distance module	9
Potential module	10
Integrator module	10
Minimisation	10

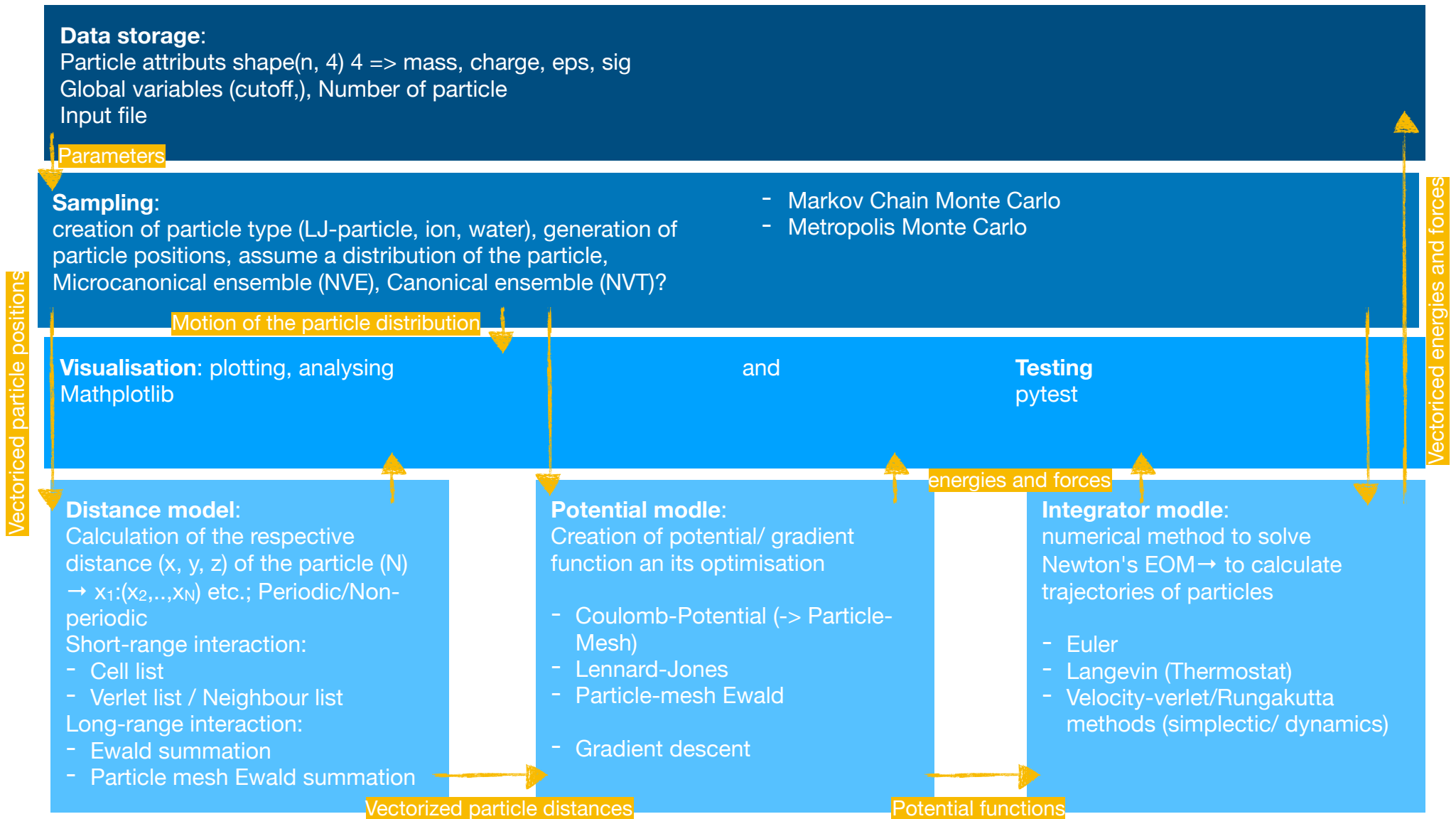
Lecture notes

11

Präsentation

12

Concept:



Workflow

So, 13.01.19

- ✓ finale aufgaben Verteilung
- ✓ modules diskutieren
- Labeling festlegen (formate, Bezeichnungen, Abkürzungen, Parameter)
- ✓ Input und Output formate und Benennungen definieren
- ✓ Ziele festlegen für Fortschritt bis zum nächsten Treffen
- Parallelization strategies ? Domain decomposition method
- (Distribution of system data for parallel computing)

Selbstständige Arbeitsphase

- Module fertig implementieren

Sa, 19.01.19

- ✓ Fortschritte präsentieren
- ✓ Fehler Probleme diskutieren
- testing diskutieren
- Input Beispiele
- Labeling festlegen (formate, Bezeichnungen, Abkürzungen, Parameter)
- ✓ Erste Ideen Wie schreiben wir alles zusammen
- Parallelization strategies ? Domain decomposition method
- ✓ Erster primitiver Testlauf

Arbeitsphase

- Visualisierung, testing
- Sampling optimieren (different properties for particles?)
- Integrator timing , visualisation closed-, pbc- system, grid ?
- Minimization
- Ewald- summation

- Create classes and Build the structure of the whole code
- Implementing grid ?

So, 27.01.19

- input examples
- Combine all modules
- Discuss extra features
 - Water
 - Parallelisation ?

-

FRAGEN:

- box anders definieren oder potential anpassen brauche jetzt array von box size !
- working with grid —>particle.mesh Ewald
- Proof vv and Langevin (pdf), was meint e_{kin} ?
- Achtung was ist mit hohen Geschwindigkeiten vv und Langevin halten keine hohen start Geschwindigkeiten aus warum ?
- Was mit histogram ?

Overview:

Aufgabe für (Person): Phillip (P), Felix (F), Malte (M), Joana (J)

MODUL	INPUT	TASK	OUTPUT	TESTING	VISUALISATION
DATA STORAGE		Configuration ?			
POTENTIAL (F)	<p>C:</p> <ul style="list-style-type: none"> - Distance shape(nxn), - charge array(n), - ϵ, σ(nx1), forces <p>LJ:</p> <ul style="list-style-type: none"> - dis(nxn), - ϵ, σ(nx1) <p>H:</p> <ul style="list-style-type: none"> - Box dim (x,y,z min and x,y,z max), - Positionen: r (einteilchen or Mehrteilchen potential), r_0= xmax/ 2, k ??? 	<p><u>Particle-particle:</u></p> <ul style="list-style-type: none"> - Coulomb <p>- LJ</p> <p><u>Global potential:</u></p> <ul style="list-style-type: none"> - Harmonic 	<p>C:</p> <ul style="list-style-type: none"> - potential(nx1) or (Gesamtpotential for each particle) - gradient shape(nx3) <p>LJ:</p> <ul style="list-style-type: none"> - potential(nx1) <p>H: Potential</p>	<ul style="list-style-type: none"> - energy consistency - Use simple potential with work and compare 	
SAMPLING (P)	<ul style="list-style-type: none"> - N = number of particle - for each particle charge, mass, ϵ, σ (nx1), k (federkonstnat) <ul style="list-style-type: none"> ▸ Creating a particle ▸ box, pbc, non pbc <p>GLOBAL attributs: ???</p>	<p>configuration: Random then:</p> <ul style="list-style-type: none"> - Markov Chain Monte Carlo - Metropolis Monte Carlo 	<ul style="list-style-type: none"> - Position shape(nxdim) - Velocity shape(nxdim) 	-	<ul style="list-style-type: none"> - distribution of particle plotten -

MODUL	INPUT	TASK	OUTPUT	TESTING	VISUALISATION
DISTANCE (P)	<ul style="list-style-type: none"> - Position shape(nxdim) 	➡ box, pbc, non pbc	<ul style="list-style-type: none"> - dist(nxn) 		
INTEGRATOR (J)	<ul style="list-style-type: none"> - position shape(nxdim), - velocity shape(nxdim), - attribute shape(n, #numb of attributes) - gradient shape(n,dim), - Time step (int) - Total time (int) - Open or close system <ul style="list-style-type: none"> ▸ box, pbc, non pbc - global attributes ? 	Euler Velocity Verlet Langevin	<ul style="list-style-type: none"> - position(size, n, dim) - velocity(size, n, dim) - Accerelation(size, n, dim) 		
Optimization (M)	??	<ul style="list-style-type: none"> - gradient descent, - conjugate gradient, - Newton methods 			

Information about modules

DATA STORAGE MODULE

???

SAMPLING MODULE

Initial configuration (markow chain, Particles equispaced in a box; 2D/ 3D)

MONTE CARLO

INPUT	OUTPUT
initial configuration $\mathbf{r}(0)$, potential, size, beta, step	Position \mathbf{r} , potential(\mathbf{r})

(1) Start with initial configuration $\mathbf{r}^{(0)}$ and set $k = 0$.

(2) Fork=1,...,K:

- (a) Sample random vector $\eta \in \mathbb{R}^N$ with $\eta_i \sim N(0, \sigma^2)$, i.e. an isotropic multivariate Gaussian distribution/ or function with variances σ^2 .
- (b) Propose new configuration $\mathbf{r}' = \mathbf{r}^{(k)} + \eta$.

(c) Accept new configuration with probability $p_{\text{acc}} = \min \left\{ 1, \exp \left(\frac{\phi(\mathbf{r}') - \phi(\mathbf{r}^{(k)})}{k_B T} \right) \right\}$

If accepted, set $\mathbf{r}^{(k+1)} = \mathbf{r}'$, otherwise $\mathbf{r}^{(k+1)} = \mathbf{r}^{(k)}$.

Markov Chain Monte Carlo

- create samples from a possibly multi-dimensional continuous random variable, with probability density proportional to a known function.
- these samples can be used to evaluate an integral over that variable, as its expected value or variance.
- in Markov chain Monte Carlo methods are autocorrelated.
- a equilibrium distribution which is proportional to the function given.

Metropolis Monte Carlo

- Konfigurationen gemäß ihrer Boltzmann-Wahrscheinlichkeit + einfachen arithmetischen Durchschnitt
- zuerst eine zufällige Bewegung ausgeführt
- dann die Boltzmann-Wahrscheinlichkeit einer solchen Bewegung ausgewertet wird und die Wahrscheinlichkeit mit einer Zufallszahl verglichen wird.
- Ist die Boltzmann-Wahrscheinlichkeit der Bewegung größer als die Zufallszahl, wird die Bewegung akzeptiert. Andernfalls wird das System auf seine ursprüngliche Konfiguration zurückgesetzt.

Vor- und Nachteile: ?

- not interested in the statistics of momenta anymore, so we integrate them out
- sampling the probability density.
- simple version of Monte Carlo is Metropolis Monte Carlo

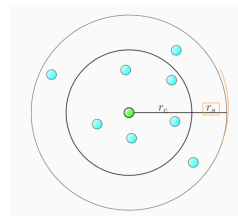
DISTANCE MODULE

INPUT	OUTPUT
Position r , potential(r)	Matrix distances $x_1 \ x_2 \ \dots \ x_N$

- calculation of distances between particles in MD Energy and force Calculation
- Introduce a cutoff and considering only the nearest neighbours within the radius; for LJ.Potential $r_c = 2.5 \sigma$
 - Reduce effort: loop over all pairs of particle $(i,j) = (j,i)$

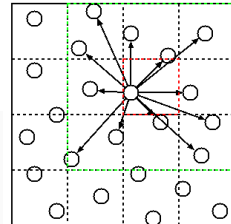
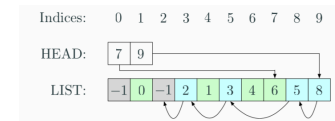
Verlet list

- For each particle i , a Verlet list has lists all other particles j within a smaller distance than $r_c + r_s$
- Update the neighbour list every N_s timesteps
- $r_s > N_s v_{typ} \delta t$ type speed of particles
- Choice between hash table and array



Cell-linked list : get complexity down

- domain into cells with an edge length greater than or equal to the cut-off radius of the interaction to be computed
- particles sorted into cells and the interactions are computed between particles in the same or neighbouring cells
- Needed array HEAD and array LIST
- Think about parallelisation



Cell decomposition approach

- partition the domain into cells and build a cell-linked list
- Verlet list by searching only in the neighbouring cells instead of considering all particles
- Dirty flag ?

POTENTIAL MODULE

description how the particles will interact in the simulation; two main approximations: Born–Oppenheimer approximation (dynamics of electrons), point particles that follow classical Newtonian dynamics (nuclei, which are much heavier than electrons)

Coulomb-Potential (-> Particle-Mesh)

$$\phi_i(\mathbf{r}) = \frac{1}{4\pi\epsilon_0} \frac{q_i}{|\mathbf{r} - \mathbf{r}_i|}$$
$$\frac{1}{4\pi\epsilon_0} = c_0^2 \mu_0 = c_0^2 \mu_0 = c_0^2 \times 10^{-7} \text{ H}\cdot\text{m}^{-1} = 8.987 \times 10^9 \text{ Nm}^2\text{C}^{-2}$$

Lennard-Jones

Particle-mesh Ewald

Ewald summation

Usage: standard method for calculating long-range interactions in periodic systems

- rapid convergence of the energy compared with that of a direct summation: high accuracy and reasonable speed when computing long-range interactions
-

INTEGRATOR MODULE

See notebooks :

- Euler
- Langevin
- Velocity-verlet

MINIMISATION

???

Lecture notes

line_profiler

Piecewise

@njit

Recent body: not can change its structure , shake or radle

Hard spheres

Ewald analytically fouler (strait forward)

Particle mesh a numerical fouler (corektli sparred particle)

Gommecs

codacy : <https://app.codacy.com/login>

Compile?

Circle ci

Präsentation

