Setting up MD Simulations of Biomolecules

Stefan Boresch
stefan@mdy.univie.ac.at

Department of Computational Biological Chemistry
Faculty of Chemistry, University of Vienna

Vienna Summer School on Drug Design — September 19, 2019

Molecular dynamics (MD) in a nutshell

One particle

force=mass×acceleration

$$\mathbf{F} = m\,\mathbf{a}$$

i.e.

$$\frac{d^2\mathbf{r}}{dt^2} = \ddot{\mathbf{r}} = \frac{1}{m}\mathbf{F}$$

The position $\mathbf{r}(t)$ of the particle is described by a 2nd order differential equation (Initial condition: $\mathbf{r}$ and $\mathbf{v}$ at $t = 0$)

N particles

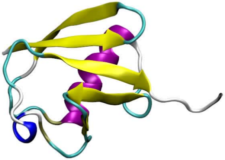
$$\ddot{\mathbf{r}}_1 = \frac{1}{m_1}\mathbf{F}_1(\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_N)$$

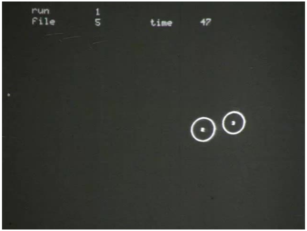
$$\ddot{\mathbf{r}}_2 = \frac{1}{m_1}\mathbf{F}_2(\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_N)$$

$$\ldots$$

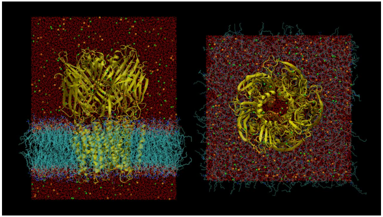
$$\ddot{\mathbf{r}}_N = \frac{1}{m_N}\mathbf{F}_N(\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_N)$$

Numerical integration:





(Martin Karplus & co-workers, 1964-67)



(Courtesy: Marco Cecchini 2013)

Ingredients of MD simulations of biomolecular systems

- Approximate, fast description of interactions
  $\Rightarrow$ force fields
- Numerical integration of the (classical) equations of motion
  $\Rightarrow$ several excellent programs available
- Analysis of the data (i.e., trajectories)
  $\Rightarrow$ still the “final frontier”

Ingredients of MD simulations of biomolecular systems

- Approximate, fast description of interactions
  ⇒ force fields
- Numerical integration of the (classical) equations of motion
  ⇒ several excellent programs available
- Analysis of the data (i.e., trajectories)
  ⇒ still the "final frontier"
- Assembling and setting up the simulation system
  - Build a meaningful representation of the real problem/system
  - Fill in missing pieces (e.g., missing coordinates)
  - Make any necessary adjustments of state of your system (either based on experimental info or "educated guesses")

Analysis — the “final frontier”

Think about it before you start simulating:*What quantities do you want to compute — how are you going to extract them from the simulation “raw data” (i.e., trajectories)*?

Monitor your system (routine analysis) throughout all stages of the project!

Running simulations . . .

ACEMD, AMBER, CHARMM, DESMOND, GENESIS, GROMACS, GROMOS, LAMMPS, NAMD, OpenMM, TINKER

. . .

Running simulations . . .

ACEMD, AMBER, CHARMM, DESMOND, GENESIS, GROMACS, GROMOS, LAMMPS, NAMD, OpenMM, TINKER . . .

- Stick with the program for which experience is available in your group / nearby!
- Avoid hunting for the “fastest” MD engine

Running simulations . . .

ACEMD, AMBER, CHARMM, DESMOND, GENESIS, GROMACS, GROMOS, LAMMPS, NAMD, OpenMM, TINKER

. . .

- Stick with the program for which experience is available in your group / nearby!
- Avoid hunting for the “fastest” MD engine

- Do not underestimate cost of MD based studies
  - CPU/GPU resources for running simulations
  - Disk space for storing / analyzing data
  - Analysis may also be very costly!

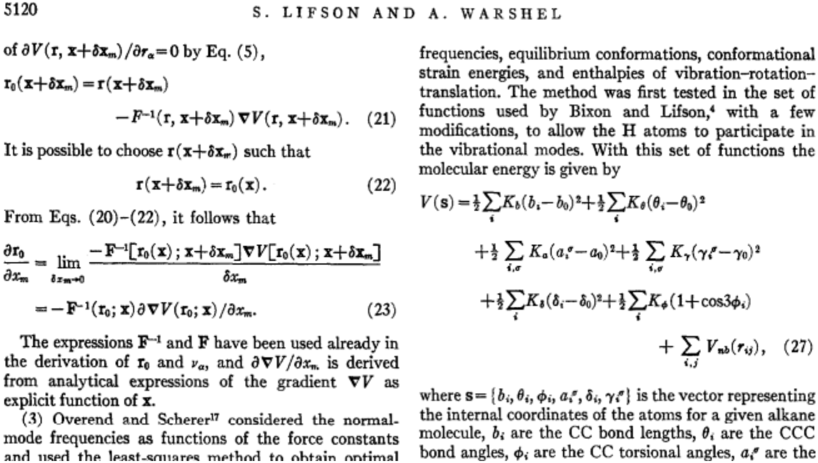
Running simulations — the more you know . . .

- *The good, the bad and the user in soft matter simulations* BBA 1858 (2016), 2529-38.
- *Real Cost of Speed: The Effect of a Time-Saving Multiple-Time-Stepping Algorithm on the Accuracy of Molecular Dynamics Simulations* JCTC 13 (2017), 2367-72

- User beware! If something's too good to be true, it may be a program bug or wrong setting!

Force fields

Pairwise, additive force fields: Eq. 27 of **Lifson** and Warshel, JCP 49, 5116 (1968)



of $\partial V(\mathbf{r}, \mathbf{x}+\delta\mathbf{x}_m)/\partial r_\alpha = 0$ by Eq. (5),

$$\mathbf{r}_0(\mathbf{x}+\delta\mathbf{x}_m) = \mathbf{r}(\mathbf{x}+\delta\mathbf{x}_m) - F^{-1}(\mathbf{r}, \mathbf{x}+\delta\mathbf{x}_m)\nabla V(\mathbf{r}, \mathbf{x}+\delta\mathbf{x}_m). \quad (21)$$

It is possible to choose $\mathbf{r}(\mathbf{x}+\delta\mathbf{x}_m)$ such that

$$\mathbf{r}(\mathbf{x}+\delta\mathbf{x}_m) = \mathbf{r}_0(\mathbf{x}). \quad (22)$$

From Eqs. (20)–(22), it follows that

$$\frac{\partial \mathbf{r}_0}{\partial x_m} = \lim_{\delta x_m \to 0} \frac{-\mathbf{F}^{-1}[\mathbf{r}_0(\mathbf{x}); \mathbf{x}+\delta\mathbf{x}_m]\nabla V[\mathbf{r}_0(\mathbf{x}); \mathbf{x}+\delta\mathbf{x}_m]}{\delta x_m}$$
$$= -\mathbf{F}^{-1}(\mathbf{r}_0; \mathbf{x})\partial\nabla V(\mathbf{r}_0; \mathbf{x})/\partial x_m. \quad (23)$$

The expressions $\mathbf{F}^{-1}$ and $\mathbf{F}$ have been used already in the derivation of $\mathbf{r}_0$ and ν_α, and $\partial\nabla V/\partial x_m$ is derived from analytical expressions of the gradient ∇V as explicit function of $\mathbf{x}$.

(3) Overend and Scherer[17] considered the normal-mode frequencies as functions of the force constants and used the least-squares method to obtain optimal frequencies, equilibrium conformations, conformational strain energies, and enthalpies of vibration–rotation–translation. The method was first tested in the set of functions used by Bixon and Lifson,[4] with a few modifications, to allow the H atoms to participate in the vibrational modes. With this set of functions the molecular energy is given by

$$V(\mathbf{s}) = \tfrac{1}{2}\sum_i K_b(b_i - b_0)^2 + \tfrac{1}{2}\sum_i K_\theta(\theta_i - \theta_0)^2$$
$$+ \tfrac{1}{2}\sum_{i,\sigma} K_a(a_i^\sigma - a_0)^2 + \tfrac{1}{2}\sum_{i,\sigma} K_\gamma(\gamma_i^\sigma - \gamma_0)^2$$
$$+ \tfrac{1}{2}\sum_i K_\delta(\delta_i - \delta_0)^2 + \tfrac{1}{2}\sum_i K_\phi(1+\cos 3\phi_i)$$
$$+ \sum_{i,j} V_{nb}(r_{ij}), \quad (27)$$

where $\mathbf{s} = \{b_i, \theta_i, \phi_i, a_i^\sigma, \delta_i, \gamma_i^\sigma\}$ is the vector representing the internal coordinates of the atoms for a given alkane molecule, b_i are the CC bond lengths, θ_i are the CCC bond angles, ϕ_i are the CC torsional angles, a_i^σ are the

Force fields

- AMBER, CHARMM, GROMOS, OPLS-AA/M
- OPLS3e (Schrödinger)

Standard force fields typically provide parameters for (in approximately descending order of quality):

- Proteins
- DNA/RNA
- Fatty acids, membranes
- Carbohydrates
- Drug-like small molecules
- Modifications of amino acids etc.

Force fields — Dos and Donts

- Do not mix and match parameters; do not substitute a “better” water model
- Do respect cut-offs, tapering functions etc. of your force field; they are part of the parameterization
- Do use the available “educated guess” generators for “your” force field. (Similarly, if you optimize an “educated guess” further or develop parameters on your own, follow the rules of the force field you are targeting to remain consistent with it.)

Force fields: If there are no standard parameters . . .

"Educated guess" generators and (semi-)automated optimization of parameters

- CGenFF ⇒ CHARMM
- SwissParam ⇒ CHARMM
- Automated Topology Builder ⇒ GROMOS
- ANTECHAMBER & GAFF ⇒ AMBER
- LigParGen ⇒ OPLS-AA/M
- ffTK ⇒ CHARMM
- GAAMP⇒ CHARMM/AMBER (try also: gaamp.lcrc.anl.gov)
- . . .

- Open Force Field Initiative

Force fields: If there are no standard parameters . . .

"Educated guess" generators and (semi-)automated optimization of parameters

- CGenFF ⇒ CHARMM
- SwissParam ⇒ CHARMM
- Automated Topology Builder ⇒ GROMOS
- ANTECHAMBER & GAFF ⇒ AMBER
- LigParGen ⇒ OPLS-AA/M
- ffTK ⇒ CHARMM
- GAAMP⇒ CHARMM/AMBER (try also: gaamp.lcrc.anl.gov)
- . . .
- Open Force Field Initiative

Force fields — an opinionated summary

Things *can* go wrong:

- Intrinsically disordered proteins (Nature Meth. 2017, 14, 71)
- Protein association(JCTC 2014, 10, 5113)
- . . . [Insert your favorite force field bug!]

Force fields — an opinionated summary

Things *can* go wrong:

- Intrinsically disordered proteins (Nature Meth. 2017, 14, 71)
- Protein association(JCTC 2014, 10, 5113)
- . . . [Insert your favorite force field bug!]

Do blame the force field only after you have eliminated all other sources of error!

Setting up the System — Then and Now

Before ≈2005

- Setting up, e.g., a protein–ligand complex, and completing a few nanosecond simulation is a “major undertaking”

After ≈2005

- Setup of even complex simulations has become “easy”
- GPUs have changed routine simulation lengths from ns to μs

Setting up the System — Then and Now

Before ≈2005

- Setting up, e.g., a protein–ligand complex, and completing a few nanosecond simulation is a “major undertaking”

After ≈2005

- Setup of even complex simulations has become “easy”
- GPUs have changed routine simulation lengths from ns to μs
- *We now have the time (and obligation) to think what we are actually doing!*

Setting up the System

We are restricted to toy models of reality

- Approximate nature of force fields
- System size and composition

The challenge is to represent the real system of interest as best as we can

Setting up the System

We are restricted to toy models of reality

- Approximate nature of force fields
- System size and composition

The challenge is to represent the real system of interest as best as we can

Errors/omissions during system set-up make your simulation questionable if not wrong, regardless of any computational effort!

Setting up the System

- Any MD simulation requires reasonable starting coordinates for all atoms . . .
  - X-ray, NMR
  - Cryo-electron microscopy
  - Integrative/hybrid (I/H) methods
  - Homology modeling
  - Assembling a larger structure from "bits and pieces"
- . . . ⇒ Deal with missing coordinates!
- What to include in the simulation?
- Add water box and ions / embed biomolecule in membrane, micelle etc
- Reflect experimental conditions in your simulation system
  - protonation states
  - membrane composition

Learn as much as possible about your system!

Setting up the system

Even when starting from a "traditional", experimental pdb file, you have to watch out for:

- Missing coordinates
  - Missing backbone coordinates / gaps
    ⇒ Loop modeling
  - Missing side chain coordinates
  - Missing hydrogens
  - Quality of ligand coordinates may be doubtful

Setting up the system

Even when starting from a “traditional”, experimental pdb file, you have to watch out for:

- Missing coordinates
  - Missing backbone coordinates / gaps
    ⇒ Loop modeling
  - Missing side chain coordinates
  - Missing hydrogens
  - Quality of ligand coordinates may be doubtful
  - Ambiguities, e.g. side chain 'flips'
    ⇒ Run WHAT_CHECK, MolProbity, NQ-Flipper etc.
- Protonation/tautomeric state(s) (protein *and* ligand!)

Setting up the system

Even when starting from a "traditional", experimental pdb file, you have to watch out for:

- Missing coordinates
  - Missing backbone coordinates / gaps
    ⇒ Loop modeling
  - Missing side chain coordinates
  - Missing hydrogens
  - Quality of ligand coordinates may be doubtful
  - Ambiguities, e.g. side chain 'flips'
    ⇒ Run WHAT_CHECK, MolProbity, NQ-Flipper etc.
- Protonation/tautomeric state(s) (protein *and* ligand!)
- Phosphorylation, glycolysation, . . .

Setting up the system

Protonation (+ tautomeric state)

- Proteins: PROPKA (& PDB2PQR)
- Organic molecules:
  - Various (empirical) tools, e.g., ChemAxon, OpenEye, Epik (Schrödinger), ACD pKA, S+pKa ...
  - Fast QM based methods, e.g., JPC A 2017, 121, 699
- Protein–ligand complexes, e.g., Protoss/Proteins*Plus*
- **Challenge:** When assigning protonation states and choosing tautomers, your choice for one site affects (in principle) all others.
- Constant pH methods

Setting up the system — tools to the rescue

Setting up the system — tools to the rescue

- **CHARMMing**
- **CHARMM-GUI**

Setting up the system — tools to the rescue

- **CHARMMing**
- **CHARMM-GUI**
  - Proteins
  - Membranes
  - Multi-component systems

See CHARMM-GUI's "Video Demos" for (more) things to consider when setting up a simulation system!

Closing thoughts I

- I had to leave out lots of stuff:
  - Long range electrostatics *and* Lennard-Jones
  - Thermostats, barostats, constraints, multiple timestep integrators
  - Other tricks of the trade
  - Polarizable force fields, QM/MM etc.
  - ...

Closing thoughts I

- I had to leave out lots of stuff:
    - Long range electrostatics *and* Lennard-Jones
    - Thermostats, barostats, constraints, multiple timestep integrators
    - Other tricks of the trade
    - Polarizable force fields, QM/MM etc.
    - . . . but I really think system setup is most important!

Closing thoughts I

- I had to leave out lots of stuff:
  - Long range electrostatics *and* Lennard-Jones
  - Thermostats, barostats, constraints, multiple timestep integrators
  - Other tricks of the trade
  - Polarizable force fields, QM/MM etc.
  - ... but I really think system setup is most important!

- Some opinionated advice
  - Be cognizant of the pitfalls/difficulties!
  - Be pragmatic, e.g., focus on active site
  - Document you decisions and, ideally, have someone else check
  - If necessary ("strange" results), rethink your choices!

Closing thoughts II

- MD simulations of biomolecular systems are becoming a routine method and are not for specialists only

Closing thoughts II

- MD simulations of biomolecular systems are becoming a routine method and are not for specialists only
- Use the available tools to help set up simulations – just don't trust them blindly!

Closing thoughts II

- MD simulations of biomolecular systems are becoming a routine method and are not for specialists only
- Use the available tools to help set up simulations – just don't trust them blindly!
- Protein Dynamics: Moore's Law in Molecular Biology (Current Biology 2011, 21, R68)

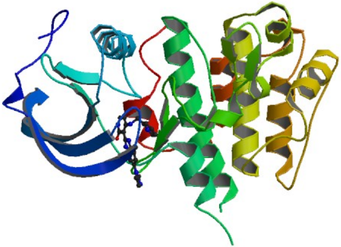
Closing thoughts II

- MD simulations of biomolecular systems are becoming a routine method and are not for specialists only
- Use the available tools to help set up simulations – just don't trust them blindly!
- Protein Dynamics: Moore's Law in Molecular Biology (Current Biology 2011, 21, R68)

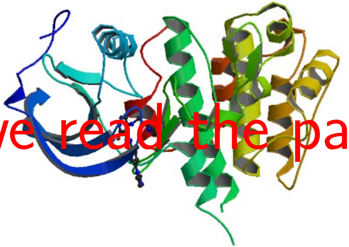
Carrying out simulations has become "relatively easy"; thus, we can and should concentrate on carrying out meaningful simulations!

Thank you for your attention!

Setting up PDB: 2OJ9 with charmm-gui

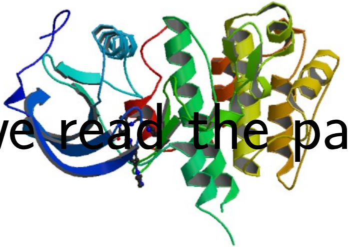


Setting up PDB: 2OJ9 with charmm-gui

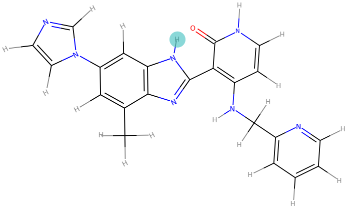


Did we read the paper??

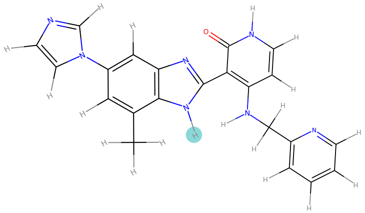
Setting up PDB: 2OJ9 with charmm-gui



Did we read the paper??



Assumed bound form



But: structure found in PDB