# Practical Course on Molecular Dynamics and Trajectory Analysis

## Episode 3: Preparing and editing the system

Jordi Villà i Freixa

Universitat de Vic - Universitat Central de Catalunya
Facultat de Ciències, Tecnologia i Enginyeries (FCTE)

*jordi.villa@uvic.cat*

MD Course and Trajectory Analysis
Concepcion, January 2026

# Contents

# Episode objectives

- Prepare systems ready for simulation in OpenMM.
- Standardize topologies and coordinates.
- Produce reproducible, verifiable inputs.

# Preparation workflow

1. Read the structure (PDB/MOL2/SDF).
2. Repair and complete missing residues.
3. Protonate and assign charges.
4. Solvate, add ions, and define the box.
5. Minimize and validate.

# Input files

- PDB: coordinates and experimental metadata.
- MOL2/SDF: ligands and small molecules.
- Topology and parameters are assigned after loading.

# Essential identifiers

$$\text{Residue} = (\text{chain}, \text{number}, \text{name}),$$
$$\text{Atom} = (\text{type}, \text{element}, \text{charge}).$$

- Consistent identifiers prevent errors in the force field.

# Reproducibility

- Save preparation scripts.
- Document library versions and parameters.
- Avoid non-traceable manual steps.

# Official modeling workflow

- The OpenMM User Guide (§4.1-4.6) recommends reconnecting hydrogens, adding solvent, and membranes before parameterization.
- The examples repo includes `simulateAmber.py` + `simulateCharmm.py`, and `argon-chemical-potential.py` to validate free energies in simple liquids.
- We reuse that infrastructure for alanine dipeptide and the protein–ligand complex, extending scripts with membrane variables or coarse-grained polymers.

# Additional systems

- `coarse_grained_polymer.py` (OpenMM Cookbook) builds bead-spring topologies that show how to define masses and bonds manually.
- The same approach links to `argon-chemical-potential.py`, where Lennard-Jones forces are parameterized and insertion free energy is measured.
- Biomolecular models benefit from combining these lightweight examples with Amber-style solvation and heavy ligand handling.

# Membranes and solvents

- Add solvent with ‘Modeller.addSolvent‘ and choose OPC/TIP3P (User Guide §4.2).
- For membranes, use ‘Modeller.addMembrane‘ and then the `simulateAmber.py` script with an anisotropic barostat.
- Save the final topology to reproduce the system exactly (OpenMM App §4.6).

# PDB quality

- Resolution and B-factors indicate uncertainty.
- Flexible regions often show gaps or low occupancy.

# Missing residues

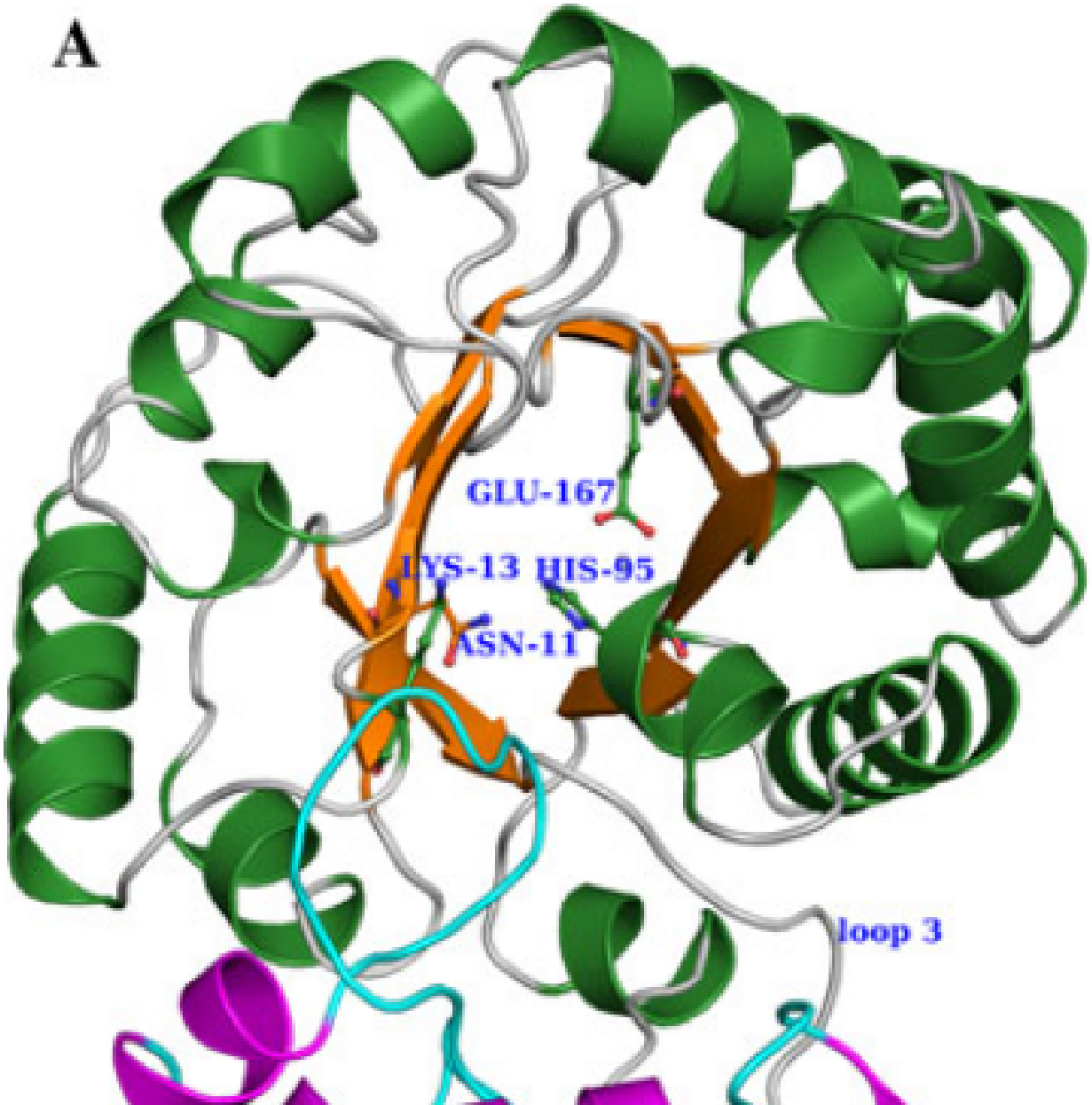
- PDB files may lack entire segments.
- Reconstruction requires inferring geometry and stereochemistry.

# AltLocs and occupancy

$$\sum_k \text{occ}_k \leq 1.$$

- Choose the dominant conformation or average according to the goal.

# Active site in PDB



# Superposition and RMSD

$$\text{RMSD} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\|\mathbf{r}_i - \mathbf{r}_i^{\text{ref}}\|^2}.$$

- Verify consistency with reference structures.

# PDBFixer/Modeller

- Fills residues, corrects names, and removes unwanted ligands.
- Prepares the system for force field assignment.

# Residue insertion

- Uses geometric and stereochemical information.
- Local minimization relieves clashes.

# Protonation and pH

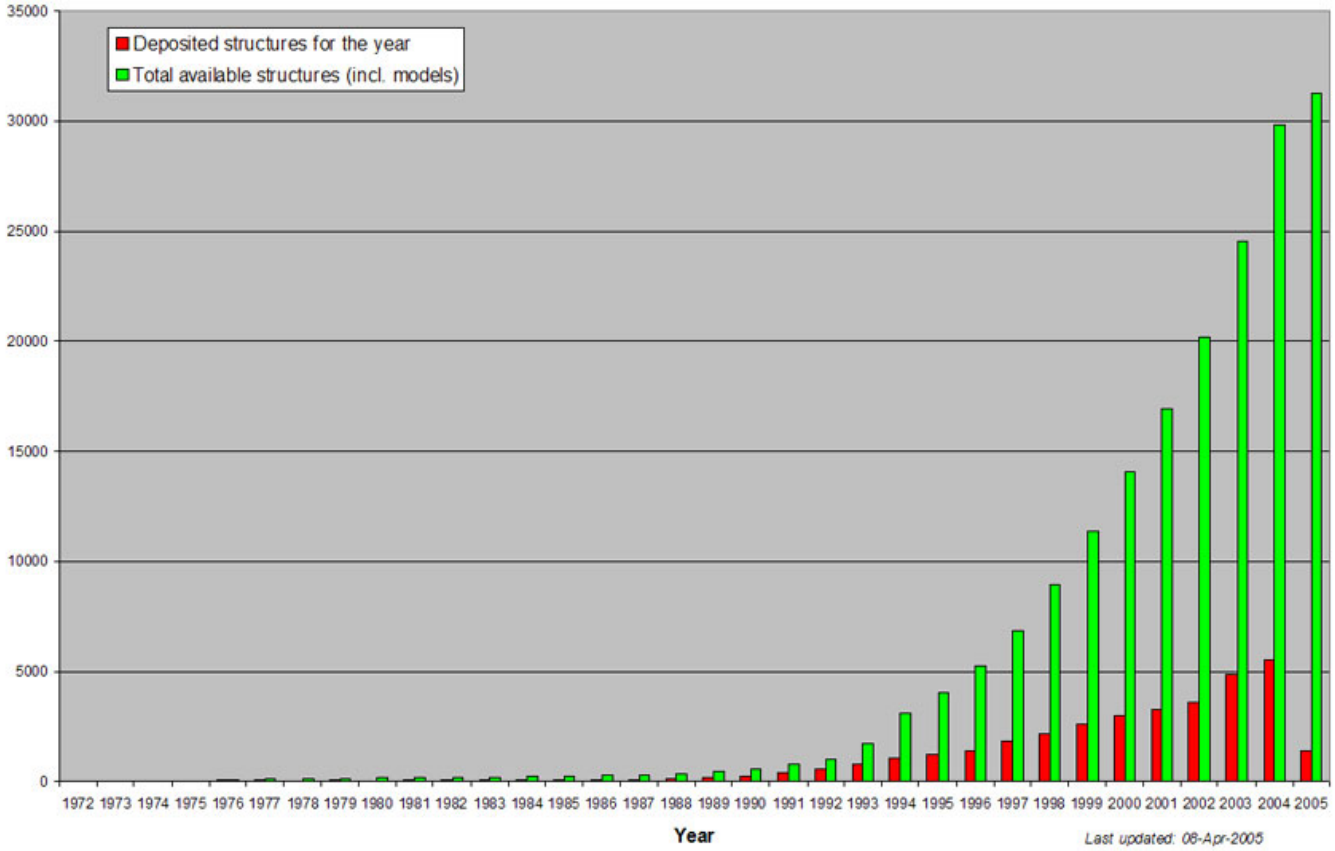
$$\frac{[A^-]}{[HA]} = 10^{\mathrm{pH}-\mathrm{p}K_a}.$$

- Determines charge states of titratable residues.

# Charge states

- Tune histidines (HID/HIE/HIP), ASP/GLU, LYS/ARG.
- Maintain consistency with the active site environment.

# Structural growth



# Potential decomposition

$$U = U_{\text{bond}} + U_{\text{angle}} + U_{\text{dihedral}} + U_{\text{nonbonded}}.$$

- Basis for computing forces and energies.

# Bonds and angles

$$U_{\text{bond}} = \sum_b k_b (r_b - r_b^0)^2,$$
$$U_{\text{angle}} = \sum_a k_a (\theta_a - \theta_a^0)^2.$$

- Harmonic approximation around the equilibrium.

# Dihedrals

$$U_{\text{dihedro}} = \sum_d \frac{V_d}{2} \left[1 + \cos(n_d \phi_d - \gamma_d)\right].$$

- Control rotational barriers and conformations.

# Nonbonded terms

$$U_{\mathrm{LJ}} = 4\epsilon \left[ \left( \frac{\sigma}{r} \right)^{12} - \left( \frac{\sigma}{r} \right)^{6} \right],$$

$$U_{\mathrm{C}} = \frac{1}{4\pi\varepsilon_0} \frac{q_i q_j}{r_{ij}}.$$

- LJ and Coulomb dominate long-range interactions.

# Force field selection

- Proteins: AMBER/CHARMM/OPLS.
- Ligands: OpenFF or other parameterizers.
- Validate compatibility with the rest of the system.

# Parameter consistency

- Avoid mixing force fields without clear rules.
- Check units and energy scales.

# Partial charges

$$\sum_i q_i = Q_{\text{total}}.$$

- The total must match the protonation state.

# Electrostatic fitting

- RESP and AM1-BCC derive charges from electrostatic potentials.
- The charge distribution affects binding energies.

# Dihedral scans

$$E(\phi) = E_0 + \sum_k V_k \cos(k\phi - \gamma_k).$$

- Fit rotational profiles on ligands.

# Type assignment

- Atomic types determine LJ and bond parameters.
- Inconsistencies yield unphysical energies.

# Ligand validation

- Check geometry, chirality, and net charges.
- Compare with experimental or QM references.

# Box choice

- Cubic, orthorhombic, or dodecahedral.
- Trade-off between cost and distance to the periodic image.

# Number of water molecules

$$N_{\mathrm{H2O}} \approx \rho\, V\, \frac{N_A}{M_{\mathrm{H2O}}}.$$

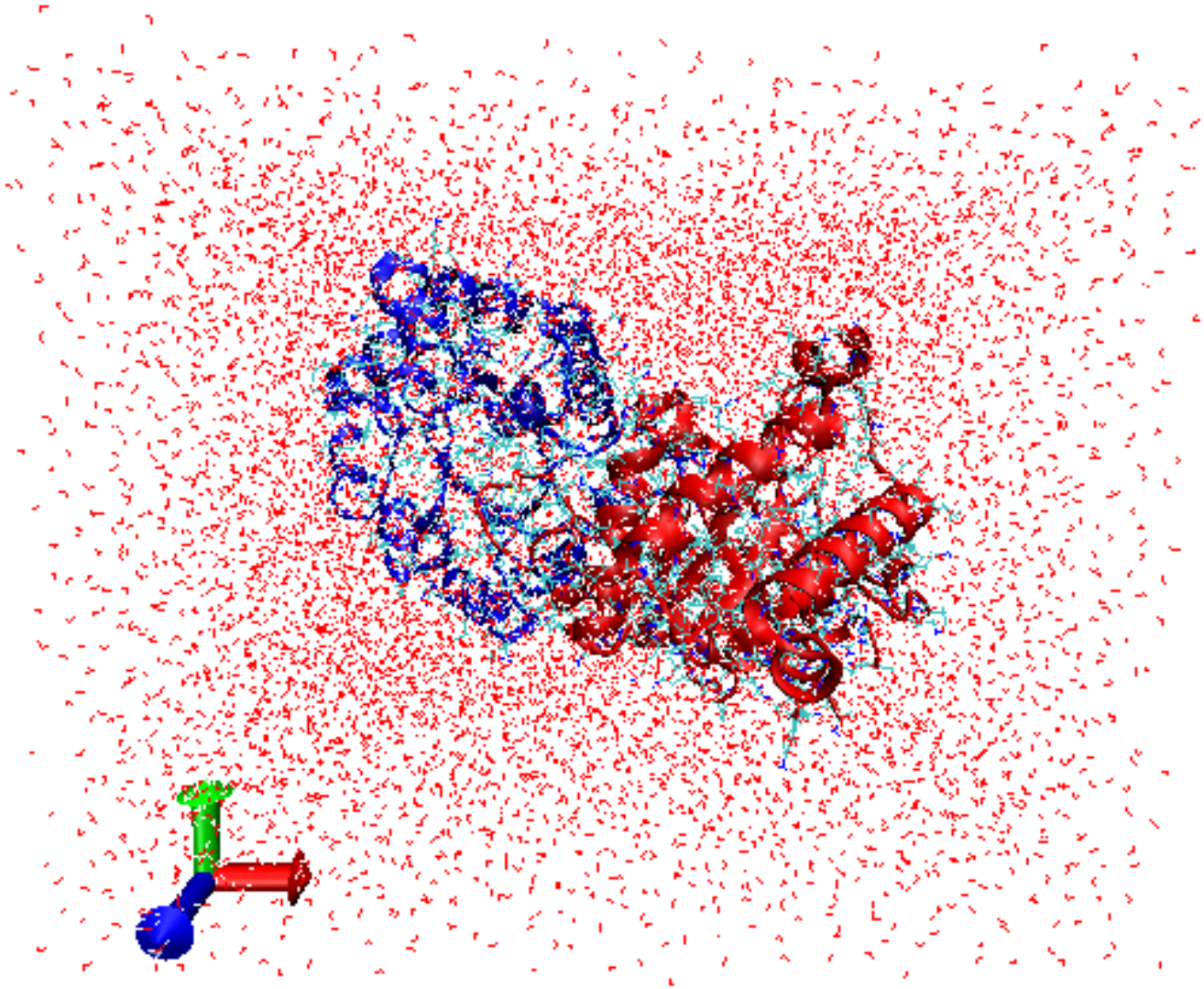
- Estimate the box size according to the target density.

# Ionic concentration

$$N_{\text{ion}} = C\ V\ N_A.$$

- Adjust molarity for physiological conditions.

# Solvated box



# Water models

- TIP3P, SPC/E, TIP4P: different densities and dynamics.
- Choose the model compatible with the force field.

# Solvent trimming

- Keep a minimum buffer around the solute.
- Prevent artificial interactions with the periodic image.

# Density check

- Verify density after NPT equilibration.
- Adjust the size if the density drifts.

# Neutralization

- Neutralizing the net charge improves numerical stability.

# Ionic strength

$$I = \frac{1}{2} \sum_i c_i z_i^2 .$$

- Controls electrostatic screening.

# Debye length

$$\lambda_D = \sqrt{\frac{\varepsilon k_B T}{2 N_A e^2 I}}.$$

- Sets the range of electrostatic interactions in solution.

# PME and cutoff

- PME handles long-range Coulomb interactions efficiently.
- Choose a cutoff consistent with the box.

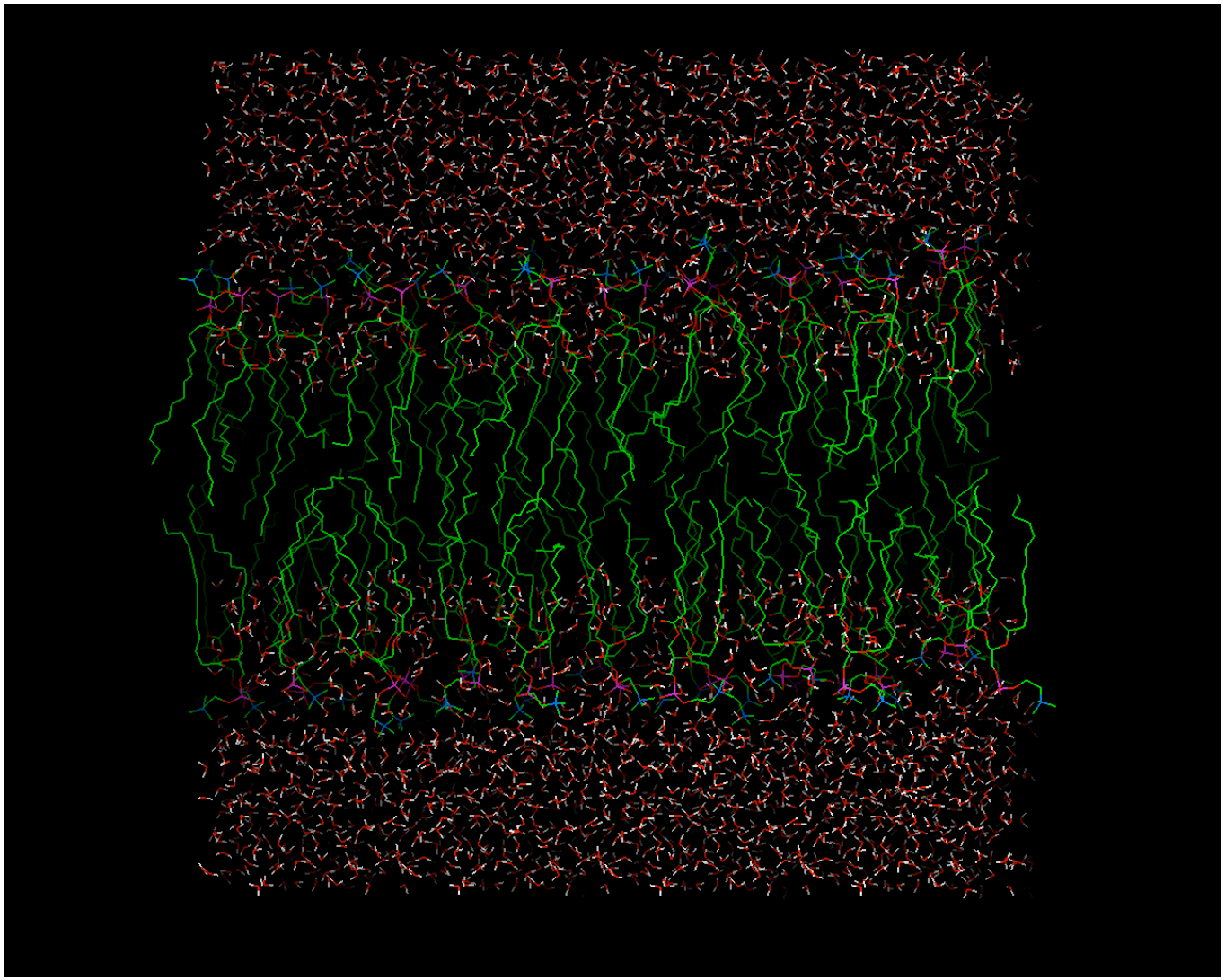
# Charge/solvent balance

- Check that the system is not overloaded with ions.

# Membrane models

- Lipid bilayers require anisotropic boxes.
- The orientation must align with the protein.

# Lipid bilayer



# Area per lipid

$$A_{\text{lip}} = \frac{A_{\text{caja}}}{N_{\text{lip}}/2}.$$

- Key metric for membrane stability.

# Membrane thickness

- Controls the exposure of hydrophobic domains.
- Adjust with minimization and equilibration.

# Initial restraints

- Restraining the protein prevents collapse during equilibration.

# Objective function

$$\min_{\mathbf{r}} \; U(\mathbf{r}).$$

- Reduces clashes and improves initial stability.

# Gradient descent

$$\mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k \nabla U(\mathbf{r}_k).$$

- Robust method for minimizing high energies.

# Conjugate gradient

- More efficient near the minimum.
- Reduces the number of force evaluations.

# Harmonic restraints

$$U_{\text{rest}} = \frac{k}{2} \sum_i \|\mathbf{r}_i - \mathbf{r}_i^0\|^2.$$

- Keep the structure while the solvent relaxes.

# Convergence criteria

- Gradient norms and energy drop.
- Stop when forces are small.

# Geometric check

- Reasonable bond lengths and angles.
- No steric clashes.

# Energy by component

- Review LJ, Coulomb, and bond contributions.
- Identify outliers.

# Comparison with reference

$$\Delta U = U_{\text{nuevo}} - U_{\text{ref}}.$$

- Large differences signal parameterization issues.

# Stability check

- Run a short NVT to catch numerical explosions.
- Inspect potential energy and temperature.

# Stable solvent

- Ensure there are no voids or overlaps.

# Checklist

- Clean PDB, correct protonation, neutralized system.
- Topology file saved.

# Save the system

- Export PDB and topology with parameters.
- Save the minimized state.

# Output formats

- PDB, Amber prmtop/inpcrd, OpenMM XML.

# Traceability

- Log scripts and configurations used.
- Version the parameters.

# Seeds and reproducibility

- Save random generator seeds.
- Document initial conditions.

# Folder structure

- Separate inputs, results, and temporary files.
- Simplifies auditing and system reuse.

# Episode summary

- Preparing the system is key for reliable results.
- Protonation and solvation decisions impact the dynamics.
- Thorough validation avoids costly mistakes.

# References I