5 Project

5.1 Introduction

Now that you have worked through setting up simulations for both protein peptides and of nucleic acids in a variety of conditions you should be ready to start exploring molecular dynamics simulations related to your own research interests. While we have not discussed all of the powerful techniques available to veteran users, you should have an understanding of how to set-up and run simulations as well as an idea of some of the common pitfalls that might come up along the way. For real-world applications of MD you would likely employ more advanced simulation protocols to enhance local sampling of important areas of phase space or alter the nature of the Hamiltonian (potential energy landscape) that you are sampling to obtain a better understanding of the free energies working upon your biochemical systems of interest.

For this final assignment we will perform a simple molecular dynamics simulation, much like those that you have performed previously in the tutorials. You will use the CaN.pdb file (the structure of a bound peptide) to run two different simulations, one in explicit tip3p water and Cl^- ions and another in a General Born implicit solvent model.

5.2 Background

Calmodulin (CaM) is a ubiquitous protein that plays a key role in calcium-mediated signal transduction. It has been shown that CaM binds and regulates more than 300 target proteins and that its structural plasticity is crucial for enabling its interaction with the diverse partners (Ikura and Ames, 2006). CaM consists of two homologous domains, the N-terminal domain (NTD) and the C-terminal domain (CTD), which are separated by an interdomain linker. Each domain is composed of two EF-hand helix-loop-helix motifs. These motifs occupy a "closed" conformation in the calcium-free (apo-CaM) state, in which the helices in the two pairs of EF hands are closely packed together. Ca^{2+} ligation during a calcium spike leads to an "open" state (Ca^{2+}-CaM) in which significant changes in conformation in each EF-hand pair result in the exposure of a hydrophobic cleft in both domains (Zhang et al., 1995; Ikura, 1996). The exposure of this cleft increases the affinity of CaM for a wide range of binding partners (Osawa et al., 1998; Crivici and Ikura, 1995; Bayley et al., 1996).

One of the binding partners is calcineurin (CaN), a serine/threonine phosphatase. CaN has, among other elements, a catalytic and regulatory domain. The regulatory domain contains an autoinhibitory region and a CaM-binding regions. At low calcium concentrations, CaN is in an inactive state, with its autoinhibitory region occluding the active-site cleft of the catalytic domain. As a result of an increase in calcium concentration, CaM binds four calcium ions, which changes the structure of CaM and enables binding of the CaM-binding

region in CaN. CaM binding releases autoinhibition and activates CaN. Experiments have shown that, like many other CaM-binding regions, the one of CaN is in an intrinsically disordered protein part. Thus, it lacks a tertiary structure before partner binding. Upon CaM binding, the region folds into a helix.

The goal of the assignment is to compare the dynamics of the CaN binding partner in its unbound state in both implicit and explicit solvent models, specifically to interrogate how the dynamics of this small peptide are affected by the choice of solvent model. Key resources to use which can facilitate this enquiry could be:

- To use the process_mdout.perl script to look at energies and how they change over the simulation.
- To examine how the secondary structures of the two simulations differ. To do this in VMD use `Extensions`→`Analysis`→`Timeline` then `Calculate`→`Calc. Sec. Struct`. Information of the Timeline extension can be found here.
- The RMSD from your starting structure of CaN over the simulation, barring in mind that the initial structure is that of the bound peptide. Does this differ between the solvent models?
- Comparing the radius of gyration of the peptide over the simulation between the two different solvent models. You can do this analysis quickly in cpptraj (much like the RMSD) by loading in the parameter and the trajectory files then issuing the command `radgyr out <output_filname.rgy>`.
- Looking at the Contact Map in VMD under `Extensions`→`Analysis`→`Contact Map` then `Calculate`→`Calc. Res-Res Dist.`. Repeat this calculation for different windows in the simulation and compare.

Some special consideration should be placed into the questions:

- To what extent do your simulations agree with the experimental observation that the CaM-binding region in CaN is intrinsically disordered and folds upon binding with CaM? If not why? Which solvation model is giving a better representation of reality?
- Is the force field used adequate for the simulation? The ff14SB is designed particularly well for general simulations, however these parameters might not be adequate for proteins with high intrinsic disorder.
- How does the computational efficiency of these simulations change with the solvent model? How important is model accuracy vs. sampling time? Under what cases would an implicit or explicit solvent model be preferred?

5.3 Simulation parameters

You have learned how to setup and run appropriate simulations to make the comparisons asked for in this project, however, you might want to think critically about the simulations before you start them. What conditions should be applied to the minimization, heating, and production steps. Furthermore, we want to run *very long* simulations (comparatively) for this project. Ideally, you will have a total simulation time of 40ns for your implicit solvent simulations, for instance. It is therefore recommended to use a 2fs timestep for your dynamics propagation. Also, in order to not make your files unmanageable, you would also want to adjust how frequently you print information to your *.nc and *.out files.

5.4 Deliverables

Report your findings to me (erich.kuechler@msl.ubc.ca) as you did for the previous tutorials. The questions above are *suggestions* and do not need to be followed directly or in that order. Additionally, you are welcome to add in other information (specific notes on visualizations of the trajectory for instance) that you think are important. To screen capture a specific window: make sure the window you want to capture is the active window them hit `Atl+Prt Scr`. After a few seconds a pop-up should appear to save an image.