Possible Homework Exercises in BIO 692

1) **Simulation of an entropic spring -** apply a stretching force to both ends of a single polymer in a DPD simulation and measure the end-end length as a function of the force and so extract the spring constant for the *entropic spring*.

Create a single polymer in a box of water with distinct beads on the two ends. Apply a constant stretching force F to the ends and let it equilibrate, then measure the end-to-end length L (it probably has to be a small force). Repeat for several forces:

How do you choose a sensible range for F?

Invert the data to get F(L) and plot it. Repeat the runs several times with different random number seeds to get a mean and standard deviation for L. Include error bars of the statistical errors in your plot.

What would you need to know before you can convert the results to physical units?

Now make some of the beads in the polymer sticky (so that the polymer tends to stick to itself) and see how this changes the F(L) curve. You will need to vary the number of sticky beads to find an interesting regime (too few and nothing will happen, too many and the polymer will just stick together in a tight ball). A possible molecular architecture is: " (BH (14 B) BT) " and you can apply the force to the BH and BT beads. *Interesting* means that the system shows some unusual, non-linear behaviour. (For a biological example see Graeter et al. JACS 13:11578 2008 on moodle.)

Output: Plot of F(L) against L for the normal polymer over a range that shows the complete response of the polymer to the force. Second plot showing the effects of the sticky domain on the F(L) curve from the first part.

(See the User Guide for the commands in DPD to apply a force to polymer.)

2) Tethered FRET experiment

In a FRET experiment, two parts of a molecule (or two molecules) have half a fluorescent group attached so that when the molecules are close enough, the fluorescent group emits light: it is a proximity sensor.

Here we use a linear molecule with two *sticky* end caps (E), and we measure the average separation of the endcaps as their stickiness and backbone stiffness is varied. The minimum value for the conservative interaction parameter is the same as the water-water parameter (25). In this case the endcaps are attracted equally to each other as they are to water: the polymer should behave as a random walk. Reducing the conservative parameter makes the endcaps more likely to stick to each other, and so reduces their mean separation (NB all DPD forces are repulsive, so a smaller parameter means a less repulsive force). Conversely, increasing the stiffness of the backbone tends to extend the molecule and should push the end caps farther apart.

Note Adjust the number fractions to ensure that you have only a single molecule in the box.

The command SavePolymerBeadRDF should be used as described in the User Guide.

Output: A plot of the radial distribution function (RDF) of the endcaps for several values of aEE including 25 as the null case. Integrate the RDF to get the mean separation as a mean and standard deviation. How does this compare to the mean polymer length in the dmpcas file?

Plot the RDF for several values of the backbone stiffness (in the BondPair lines). Note that 5 kBT is the usual value. Try 0 kBT (random walk) and higher values (stiffer molecules).

3) **Effects of a bulky GFP tag** on molecular fluctuations in solution - many fluorescence microscopy experiments attach a GFP protein to a molecule so that its motion may be tracked. But GFP is a bulky protein that may affect the dynamics of the molecule. This exercise measures the effect of a bulky rigid object on the fluctuations of a polymer in solvent.

- Simulate a single polymer of length N = 20 in solution, and plot its diffusion constant and end-to-end length as time series from the dmpchs file. How long do you need to simulate? How big a box do you need?
- Attach a GFP domain to one end of the polymer from the first part, and repeat the measurements of diffusion and end-to-end length time series. Quantify the difference compared to the original polymer. Comment on this difference.
- Increase the polymer length to N = 30, and repeat.
- Repeat the simulations with the GFP tag in the middle of the polymer and measure diffusion and end-to-end length time series. Comment on any difference from the first part.
- Extract the **mean** end-to-end length from the dmpcas file for all cases and compare them..

Output: Time series of diffusion constant and end-to-end length for the polymers without and with the GFP. The mean values of end-to-end length in all cases.

4) **Effects of a bulky GFP protein** on lipid diffusion in a membrane - measure the effects of a bound GFP tag on the in-plane diffusion of a lipid molecule, and the effective interactions when a fraction of the membrane lipids are bound to GFP.

This is similar to the previous exercise but you need to use the planar membrane initial state. Create a membrane from one lipid type that also contains a single lipid with a different type (so it can be monitored individually) but with the same interactions as the other lipids. Bind the GFP to the single lipid and allow the system to equilibrate. The box size should be at least 24 x 24 x 24 and preferably 32 x 32 x 32.

- Measure the diffusion of the GFP-lipid compared to the normal lipids in the membrane
- Run a series of simulations with increasing numbers of GFP-lipids in the membrane, and compare how the diffusion changes as the fraction of GFP-lipids increases.
- At what lipid fraction in the membrane do the GFP domains start to notice each other?

Output: Table of the diffusion constants when the membrane is in equilibrium (how can you tell?) for the cases a) no GFP, b) 1 lipid-GFP conjugate, c) several cases where an increasing fraction of the lipids have a bound GFP. Comment on the fraction at which the GFP's start to interact.

(Ask me for the membrane initial state, and GFP domain to add to the lipid(s) in the input file.)

5) **Integrate a Langevin equation and compare it to a simulation:** Consider a molecular motor dragging a vesicle along a microtubule. Model it as a Langevin equation with a constant force, use the RNG from the lectures, and integrate it to find the distance moved as a function of time, and so get the speed and see how speed varies with the applied force.

Then create a rigid nanoparticle in a simulation box of water and apply a constant force to it. Measure its position as a function of the applied force. From its position as a function of time, calculate its speed. Compare with the Langevin solution.

Output: The Langevin equation you created; a graph of the force against the speed of motor from the Langevin equation and the same graph from the simulation.

6) **Molecular force spectroscopy -** calculate the work done in pulling a lipid out of a membrane at different rates. Create a membrane from one lipid type that also contains a single lipid with a different type (so it can be monitored individually) but with the same interactions as the other lipids. Then apply a force perpendicular to the plane of the membrane to the labelled lipid and simulate it long enough to pull the lipid out of the membrane. Repeat for several values of the force. Note. Because the DPD integrator conserves momentum, pulling on the lipid will tend to pull the whole membrane and it will start to translate in the direction of the force. You will need to prevent this. How?

You will need to create an initial membrane state from a given number of lipids plus the one labelled one. Let it equilibrate (how will you know when this is?) and measure the end-to-end length of the labelled lipid, and its time-averaged value, and compare to the unlabelled lipids.

NB. For efficiency, each member of a group can simulate the membrane with different pulling force values and you can share the results.

Once you have several values of the force that pull the lipid out of the membrane, repeat the runs with a distance moved and work done decorator applied to the force. Measure the work done during the pulling experiment, and plot it as a function of the force. What force minimises the work done in pulling out the lipid?

Output: Example graph showing the position of the lipid as a function of time for one case. Graph of the work done against the pulling force for a range of forces.

7) **Construct a thermodynamic model of a polymer droplet in solvent** - Construct the free energy $F = U - TS$ for a model of a spherical droplet immersed in a solvent composed of polymers with sticky patches that make them bind weakly. You must include at least the following effects/ parameters:

- polymer volume fraction (or concentration)
- the variation in the droplet's binding energy with its radius
- the variation in a polymer's conformational entropy with droplet radius

Optional: Calculate the osmotic pressure of the droplet from your free energy using the relation

$$p = \varphi^2 \, d/d\varphi \, (f / \varphi)$$

where f = F / V is the free energy per unit volume, φ is the volume fraction of the polymers, and p is the pressure.

8) Compare the diffusion constant of a hard sphere with that of a fluctuating polymer with the same radius of gyration in a DPD simulation, and separately the same hydrodynamic radius. See Lecture 4 for the definitions of the radius of gyration of a polymer, and the hydrodynamic radius of a hard sphere and a polymer. Create a hard sphere in a DPD simulation with a (small) radius R and measure its equilibrium diffusion constant from the dmpchs file (how do you know when it's equilibrated?). In an independent simulation, create a single polymer out of N beads (of type B, say), set the self interaction of the beads (a_{BB}) to a high value so it is self-avoiding, and their interaction with the water beads equal to the water-water interaction ($a_{BB} > a_{BW} = a_{WW}$). Choose the length of the polymer (N) so that its radius of gyration is equal to that of the hard sphere. You will probably have to try several values of N until you find the best fit.

Useful relations: Lee = a $N^{0.6}$ and $L_{ee}/R_g = 2.5$ for a self-avoiding polymer, where a is the monomer diameter; $R_g^2 = 3/5\ R_h^2$ for a hard sphere;
You can take the hydrodynamic radius of the sphere to be equal to its actual radius. If you need to assign a value to a, the bead diameter in the polymer, then simulate a single polymer of N beads until it is equilibrated and use the value of the mean B-B bond length that is in the dmpcas file. The end-to-end length is also in the dmpcas file. You can use the command *SavePolymerBeadRDF* to measure the radius of gyration of the polymer in the simulation (see DPD User Guide).

Output: A graph comparing the diffusion constant (i.e., Mean square displacement / 6*time) of the hard sphere and polymer as a function of time for a) same radius of gyration, and b) same hydrodynamic radius. The values of the radius of gyration of the sphere and the polymer (which should be the same, of course, except for experimental error …) The hydrodynamic radius of both: a) calculated theoretically from the results in lecture 2, and b) measured from the apparent diffusion constant in the simulations using $D = k_B T/6\pi\eta a$, where a is the hydrodynamic radius, η is the viscosity of water, and T is room temperature.

And a comment explaining why you believe your results (or not as the case may be.)

9) What are the dense and dilute phase concentrations of a phase separated droplet of IDP?

- Select a molecular structure for the IDP (How? see Lecture 5)
- Find a concentrations at which it self-assembles (preferably a low one)
- Simulate the IDPs and measure the radius of gyration in the code
- Repeat for a second concentration that is higher than the first

  Create a command target that selects one bead in the molecules and invoke the radius of gyration command on that target (see User Guide for these commands). Make sure you simulate long enough that the droplet has formed and is stable, and average over enough samples to get good statistics.

- Command SelectBeadTypeInSphere 100 sphere T 0.5 0.5 0.5 0.0. 10.0
  Command RgOfBeadTarget 100 sphere rg1 100 200

Hint. Practise with a spherical vesicle (dmpci.rg1) and 1timestep to ensure the Rg commands work as expected.

Simulate the IDP in water until it forms a "spherical" droplet near the centre of the box. Then restart the simulation and issue the above command to measure Rg for a certain number of time steps (how long?)

Calculate the concentration of IDPs inside the droplet and in the dilute phase outside. Note that you need the number of molecules in the dense phase and the volume of the droplet to calculate its concentration (how will you get these values?) .

The concentration will be dimensionless, so include the factors r_0 in your answer.

NB. Depending on your choice of IDP molecular structure and conservative interactions, there may be no polymers in the dilute phase. This is fine, just report zero.

Output: Choice of the IDP molecular structure, snapshots of the final droplets, and calculations for the dense and dilute phase concentrations in the two cases.

10) Characterise the type of self-assembled aggregate produced by a comb polymer as its molecular structure and interactions are varied. Comb polymers have a backbone with side chains sticking off at regular intervals. The length of the side chains, their separation, and hydrophilicity/hydrophobicity all impact the type of aggregates that they self-assemble into. The aim here is to see how many distinct kinds of aggregate can be made from the same one or two polymer types.

Possible aggregates include: spherical micelles, wormlike micelles, doughnut micelles, lamella phase, vesicles, etc.

Output: One or several types of base comb polymer and the characteristics/appearance of their self-assembled aggregates. A typical polymer could be:

```
" (H1 (8 (B B B (* (S (6 S) S)) B B B)) T1) "

or the pair of slightly different polymer types:

" (H1 (8 (B B B (* (S (6 S) S)) B B B)) T1) "
" (H1 (8 (B1 B1 B1 (* (S1 S1 S1 S1)) B1 B1 B1)) T1) "
```