**Electrochemically induced protein-based process for rare earth extraction and separation**

*The purpose of this notebook is to document details regarding LanM dimer, its binding to Lanthanides, and eventually how we can leverage electrochemical setup to modulate the specificity.*

Color code: Harshit, Utkarsh

## 06/26/2024

### *Getting started…*

**Summary of the proposal:**

The **overarching goal** is to control and further enhance REE selectivity of LanM homodimer using an electrochemical scheme. The **central hypothesis** is that external stimulus (like electric field or charged surfaces) can be used to fine-tune mobility of LanM and the sensitivity of binding/desorption of the REE groups. We are particularly interested in evaluating a protein-based for separation of light-middle vs. heavy REEs and selectivity recovery between REE vs. non-REE cations using electrochemical system. To this end, we seeks to achieve the following **specific objectives**: 1) To gain a fundamental understanding of protein structure and dynamics in solution (e.g., fine-tuned immobilization onto a support vs. migration of protein towards REE ions) and light-middle vs. heavy REE ions vs. non-REE ions transport in solution, and 2) To characterize REE binding affinities to the LanM homodimer (e.g., interfacial stabilization between solvent, ion, and protein molecules, and solvation and de-solvation of the target REEs), both in response to electrochemical stimuli.

Overall, we will use computer simulations and employing state-of-the-art molecular dynamics simulations to cover the range of REEs binding to the LanM homodimer and its mutants (e.g., we will design surface mutants to create acidic/basic patches to fine-tune the migration of LanM and its binding to REEs) by examining the structure, local coordination environment, solvation energies, binding affinities and dynamics of how individual REE ions bind to free/immobilized LanM homodimer in presence of electrochemical stimuli, with the aim to identify molecular features that allows for selective binding of particular REEs.

**Initial plan of action:**

1. Ideally, we want to use AMBER03ws force field for Protein, TIP4P/2005 force field for Water – this folder also contains modified parameters for Na and Cl ions from Luo and Roux which is what we want to use for maintaining 0.1 M (100 mM) salt concentration (force fields can be found at <https://bitbucket.org/jeetain/all-atom_ff_refinements/src/master/>). Now, based on this, the 1st step is to find Lanthanide based Force Field models (as latest as possible) that are compatible with these or similar protein and water models.
2. Once we land on the force field model that we want to use, we want to perform a “short” basic unbiased simulation (similar to the workshop or mdtutorials lysosome/water simulation) using LanM Homodimer (PDB:8DQ2) structure, with all nvt, npt equilibration and production runs. Here we need to make sure that the Lanthanides present in the PDB:8DQ2 remain binded to where they are and their positions are not affected much. This is important for the next step.
3. Once this simulation is finished, we want to perform MD with enhanced sampling technique to understand the binding affinity of Lanthanide with the protein. Some thoughts on what can be done:
   1. Funnel Meta (similar to workshop) preferably, can also think about how umbrella sampling can help here.
   2. FEP simulation of adding Lanthanide in water to understand its solvation energy.

## 07/09/2024

Things to be performed:

* Perform the water – Lanthanide ion simulation and calculate the solvation energy using the FEP method. – Reference: GROMACS tutorials
* Use of the Alpha fold in superimposing the crystal structure available and then create the new structure.
* Once the structure is created use the Lanthanide ion and Protein structure see whether we ca get the theoretical binding affinity of Lanthanide using Auto-Dock vina.
* Once these are achieved now, perform the Unbiased Molecular dynamics simulations to see whether lanthanides get/remain binded or not during the simulation.

## 07/10/2024

Preparation of the force field files for the OPC-Water model and Lanthanide La3+ ion.

1. Create a Lanthanide pdb and Water pdb in the Avogadro.
2. Once both the files are created now create a tleap script used by the amber tools to create the force field files.
3. Water.pdb

1. HEADER water

2. COMPND

3. SOURCE

4. HETATM 1 H1 HOH 1 9.626 6.787 12.673

5. HETATM 2 H2 HOH 1 9.626 8.420 12.673

6. HETATM 3 O HOH 1 10.203 7.604 12.673

7. CONECT 1 3

8. CONECT 2 3

9. CONECT 3 1 2

10. END

11.

1. La.pdb

1. COMPND UNNAMED

2. AUTHOR GENERATED BY OPEN BABEL 2.3.90

3. HETATM 1 LA LA 1 0.368 2.251 0.000 1.00 0.00 LA

4. MASTER 0 0 0 0 0 0 0 0 1 0 1 0

5. END

6.

1. Create a tleap script and run the command tleap -f tleap.in

1. source leaprc.water.opc

2. loadOff atomic\_ions.lib

3. loadAmberParams frcmod.opc

4. loadAmberParams frcmod.ionslm\_126\_opc

5.

6. # Load PDB files for water and lanthanide ion

7. mol = loadPdb water.pdb

8. LA = loadPdb la.pdb

9.

10. # Combine molecules into a single unit

11. complex = combine {mol LA }

12.

13. # Save the coordinates and topology

14. saveAmberParm complex complex.prmtop complex.inpcrd

15.

16. # Quit tleap

17. quit

18.

## 07/13/2024

Superimposition of the protein structure with the alpha fold server and the crystal structure from the Protein data bank.

1. Firstly, download the PDB file of the targeted protein (8DQ2) from the PDB bank.
2. Open this PDB file in Chimera and it has shown as the tetramer. So, to make it a dimer remove the two chains and save it as the cleaned\_8DQ2.pdb.

A gold ribbon with lights

Description automatically generated with medium confidence

## The above image is the cleaned protein structure which is a dimer. Now using Alpha fold server and Alpha fold protein database copy the sequence of the protein used which also has the Intrinsically disordered protein region. (EF-hand domain-containing protein)

1. Using the Alpha fold server make the two copies of the alpha fold structure a dimer and download it.

A close-up of a string

Description automatically generated

1. The above image is the dimer downloaded from the alpha fold server. Now to create the superimposed structure use PyMOL and use the command super which is robust to superimpose the protein structures and save it as the superimposed PDB file.
2. Open PyMOL and open both the files the cleaned dimer and the alpha fold PDB structure.
3. Now in the PyMOL terminal type these commands:

Define the selections to ensure they come from single objects:

select cleaned\_8dq2\_dimer, cleaned\_8dq2

select dimer\_alpha\_fold\_dimer, dimer\_alpha\_fold

Perform the superimposition using the defined selections:

super cleaned\_8dq2\_dimer, dimer\_alpha\_fold\_dimer

1. Now save the superimposed structure as the superimposd.PDB file for the further processing for the MD Simulations.

A close-up of a string

Description automatically generated

The above image is the Final Superimposed protein structure from the Alpha fold and Protein Data bank.

**07/16/2024**

**Solvation free energy of La+3 ion in water.**

1. Once the Lanthanide and water files are obtained using Intermol convert them to the GROMACS. gro and .top file.
2. From the system.gro file create the water.pdb file and la.pdb file for the system to pack the molecules.
3. Water.pdb –

1. ATOM 1 O WAT 1 1.020 0.760 1.267

2. ATOM 2 H1 WAT 1 0.963 0.679 1.267

3. ATOM 3 H2 WAT 1 0.963 0.842 1.267

4. ATOM 4 EPW WAT 1 1.004 0.759 1.267

5. END

6.

1. Similarly create the la.pdb and run the packmol with the packmol.inp script where I kept 600 OPC water molecules.

1. tolerance 2.0

2. filetype pdb

3.

4. output system.pdb

5.

6. structure la.pdb

7. number 1

8. inside box 0. 0. 0. 50. 50. 50.

9. end structure

10.

11. structure water.pdb

12. number 600

13. inside box 0. 0. 0. 50. 50. 50.

14. end structure

15.

1. Using GROMACS run this command - gmx\_mpi editconf -f system.pdb -o system.gro
2. gmx\_mpi editconf -f system.gro -o system\_boxed.gro -c -d 1.5 -bt cubic which creates the cubic box with the box dimensions.
3. Now create the MDP folder create the .mdp files for the energy minimization and similarly do it for the NVT, NPT and Production run.
4. Followed the tutorial in the MD Tutorials for the Free energy calculations and with the following Perl script generated the empty .mdp files.

1. #!/usr/bin/perl -w

2.

3. use strict;

4.

5. # opens a generic .mdp file and replaces the values of

6. # 'init\_lambda\_state' with values at increments of 1

7.

8. unless (@ARGV) {

9. die "Usage: $0 input.mdp\n";

10. }

11.

12. my $mdp = $ARGV[0];

13.

14. my @temp = split('\.', $mdp);

15. my $base = $temp[0];

16.

17. open(IN, "<$mdp");

18. my @in = <IN>;

19. close(IN);

20.

21. # note we are specifying indices for states, not actual lambda values like in previous versions

22. for (my $i=0; $i<21; $i++) {

23.

24. my $filename = "${base}\_${i}.mdp";

25.

26. open(OUT, ">$filename");

27.

28. foreach $\_ (@in) {

29. unless ($\_ =~ /^init\_lambda\_state/) {

30. print OUT $\_;

31. }

32.

33. if ($\_ =~ /^init\_lambda\_state\s\*=/) {

34. printf OUT "%s %d\n", $&, $i;

35. }

36. }

37.

38. close(OUT);

39.

40. }

41.

42. exit;

43.

44.

1. Now based on the lambda states write the .mdp files for the EM, NVT, NPT and Production run.
2. Now submit the job using the bash file provided in the MD Tutorials and our requirements. This will create the folders for every lambda and does the EM, NVT, NPT and Production run. These repeats for every lambda and generates the data.
3. Link for the tutorial: <http://www.mdtutorials.com/gmx/free_energy/03_workflow.html>

## 07/18/2024

By fixing the Lanthanide ion in the center of the box and then placing the 600 molecules of water with OPC water model using packmol created the system.pdb.

Now submitted the calculations for the solvation free energy calculations with 21 lambda values as the states with the difference of 0.05 from Lambda = 0 to Lambda=1. For the NVT I am running NVT for 5ns, NPT 5ns and finally the production run for 20ns at 1fs.

## 07/30/2024

Initially, I submitted the calculations with only Van der waal and again I resubmitted the jobs as we discussed in the meeting which also includes Coulombic interactions. Computed the free energy for the Lanthanide ion in water. Once we use the bar module from GROMACS it prints the output given below.

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

A graph of different colored lines

Description automatically generated

This is the plot from the file generated by the gmx bar command. I have plotted the Free energy due to the Van der Waals and Coulombic interactions.

## **08/13/2024**

**Protein preparation for the Unbiased simulation:**

Using the modeller created the IDP region in the dimer of the PDB crystal structure. Once it is done extract the coordinates of the La and Na ions from the dimer using VMD and created the La.pdb file and Na.pdb files.

Once those files are created using the Packmol created the system\_modelled.pdb by fixing the positions of the ions and protein from the modeller.pdb.

**PACKMOL Script**

1. tolerance 2.0

2.

3.

4. output system\_modelled.pdb

5.

6. structure modeller.pdb

7. number 1

8. fixed 0. 0. 0. 50. 50. 50.

9. end structure

10.

11.

12. structure La.pdb

13. number 1

14. fixed 0. 0. 0. 50. 50. 50.

15. end structure

16.

17. structure Na.pdb

18. number 1

19. fixed 0. 0. 0. 50. 50. 50.

20. end structure

21.

22.

Once that is done now using the amber tools created the force field files for the final system\_modelled.pdb with the La, Na ions and in addition using the amber only solvated the box with water molecules and neutralized the system by adding the Na+ and Cl- ions.

Force field used for the Protein is: amber ff19sb

Force field used for the La, Na and Cl- ions are amber force field with OPC water model.

Force field for the water: OPC water model which works best with the amber ff19sb force field.

Now created the tleap.in script and generated the force field files.

1. # Load AMBER force fields

2. source leaprc.protein.ff19SB # Load the protein force field

3. source leaprc.water.opc # Load the OPC water model

4.

5. # Load ion parameters and atomic ions library

6. loadOff atomic\_ions.lib

7. loadAmberParams frcmod.opc

8. loadAmberParams frcmod.ionslm\_126\_opc

9.

10. # Load the protein structure

11. protein = loadPdb "system\_modelled.pdb"

12.

13. # Solvate the protein, centering it in the box with 1-2 nm from the edge

14. solvatebox protein OPCBOX 10.0 # 1.0 nm = 10.0 Å from the edge

15.

16. # Add ions to neutralize the system

17. addions protein Na+ 0

18. addions protein Cl- 0

19.

20. # Save the parameter and topology files

21. saveAmberParm protein protein\_opc\_neutral.prmtop protein\_opc\_neutral.inpcrd

22.

23. # Save the solvated structure

24. savepdb protein protein\_opc\_neutral\_solvated.pdb

25.

26. # Quit tleap

27. quit

28.

This tleap created the force field files in the amber format. prmtop and. inpcrd along with the solvation of the box, adding the charges for neutralizing the system.

Now using the amb2gro\_top\_gro\_cmap.py convert the. prmtop and. inpcrd files to the GROMACS format to do the protein water bulk simulation.

Command used: **python amb2gro\_top\_gro\_cmap.py -p protein\_opc\_neutral.prmtop -c protein\_opc\_neutral.inpcrd -t protein\_modelled.top -g protein\_modelled.gro -b protein\_modelled.pdb**

## **08/14/2024**

Things to be done:

1. With the position restraints for the heavy atoms and the LA ions do the short simulation of protein in water.
2. Once it is done from the equilibrated structure try to do the Umbrella sampling-based Replica exchange by pulling the La and distance between La and center of mass of the residues connected to the LA ion in the binding site.

## **08/20/2024**

The problem with the protein water simulation is the generation of the posre.itp files for the dimer structure. I have tried using the pdb2gmx and the command gmx genrestr for the position restraints file.

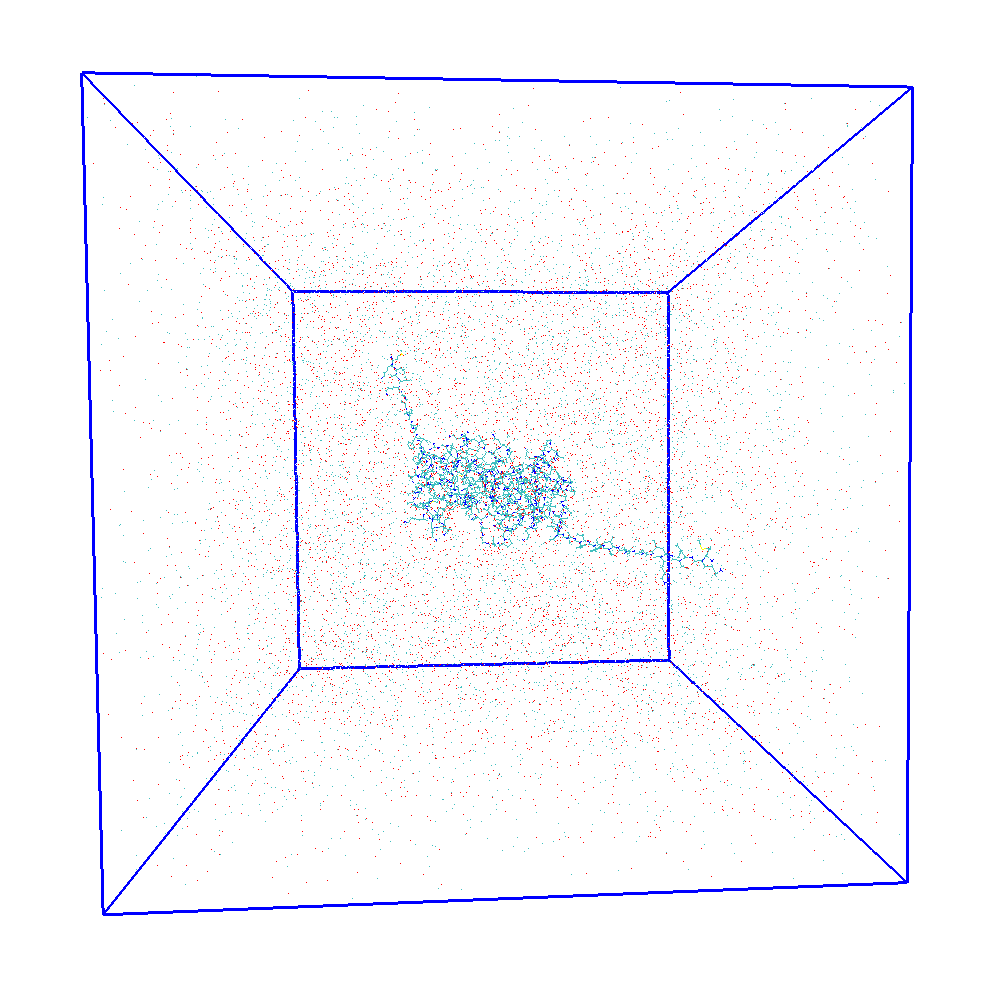
In most of the websites they have given this statement for the limitation use of the gmx genrestr. If you have multiple chains, you must renumber them in separate coordinate files. This is a major limitation of genrestr that people often encounter. Since we have the multiple chains, the problem is coming.

A white background with black text

Description automatically generated

## **08/27/2024**

Simulations for the Protein fixed are completed. Below are the snapshots of the initial configuration and the final snapshot after the production run.



A red and blue dotted square

Description automatically generated

## **09/30/2024**

* Bulk simulation is completed. Now submitted the calculations for computing the binding energy of the lanthanide from the protein. Since we have 3 lanthanides in one part of the dimer we have the 3 binding energies of the lanthanide.
* To compute the binding energy of the lanthanide I have used the decoupling method of the Thermodynamic integration method and additionally, I used the Hamiltonian replica exchange Molecular dynamics with 41 states.
* For every one lanthanide and the protein I have fixed the protein and lanthanide and then observing the sequence-based effect on the binding energy of the lanthanide.

This is the HREMD.mdp script with the decoupling of the lanthanide and keeping the Protein and Lanthanide restraint.

1. ; Production run in NPT ensemble (To be used for analysis)

4. ; Run parameters

5. integrator = sd

6. dt = 0.002 ; 2 fs

7. nsteps = 10000000 ; 20 ns

8.

9. constraints = h-bonds

10. constraint-algorithm= LINCS

11. continuation = yes

12.

13. ; Output control

14. nstxout = 500000

15. nstvout = 500000

16. nstfout = 500000

17. nstenergy = 500000

18.

19. ; Neighbor searching

20. nstlist = 5 ; 10 fs

21. ns\_type = grid

22. rlist = 1.4

23. rcoulomb = 1.2

24. rvdw = 1.2

25. DispCorr = EnerPres

26. cutoff-scheme = verlet

27. vdwtype = cutoff

28.

29. ; Electrostatics

30. coulombtype = PME

31. pme\_order = 4

32. fourierspacing = 0.1

33.

34. ; Temperature coupling

35. tc\_grps = System

36. tau\_t = 0.4 ; time constant in ps

37. ref\_t = 298.0

38.

39. ; Pressure coupling

40. pcoupl = Parrinello-Rahman

41. pcoupltype = isotropic

42. tau\_p = 2.0

43. compressibility = 1e-6

44. ref\_p = 1.0

45.

46. ; PBC

47. pbc = xyz

48.

49. nstcomm = 1

50. comm-mode = linear

51. comm-grps = System

52.

53. ; Position restraints for protein and lanthanide

54. define = -DPOSRES -DPOSRES\_LA ; Apply position restraints and Lanthanide constraint

55.

56. ; Free energy control

57. free\_energy = yes

58. init\_lambda\_state = 0

59. delta\_lambda = 0

60. calc\_lambda\_neighbors = 1 ; Only immediate neighboring windows

61.

62. ; init\_lambda\_state 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41

63.

64.

65. ; Lambda vectors (decouple Coulomb first, then vdW)

66. vdw\_lambdas = 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 0.95 0.90 0.85 0.80 0.75 0.70 0.65 0.60 0.55 0.50 0.45 0.40 0.35 0.30 0.25 0.20 0.15 0.10 0.05 0.00

67. coul\_lambdas = 1.00 0.95 0.90 0.85 0.80 0.75 0.70 0.65 0.60 0.55 0.50 0.45 0.40 0.35 0.30 0.25 0.20 0.15 0.10 0.05 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00

68.

69. ; No transformation of bonded or restrained interactions

70. bonded\_lambdas = 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00

71. restraint\_lambdas = 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00

72. mass\_lambdas = 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00

73. temperature\_lambdas = 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00

74.

75. ; Soft-core settings for the decoupling

76. sc-alpha = 0.5

77. sc-coul = no

78. sc-power = 1

79. sc-sigma = 0.3

80.

81. couple-moltype = LA ; name of moleculetype to decouple

82. couple-lambda0 = none ; no interaction

83. couple-lambda1 = vdw-q ; full interaction

84. couple-intramol = no

85.

86. nstdhdl = 100

87. dhdl-print-energy = total

## **10/03/2024**

I have changed the settings of the mentioned parameters to

1. couple-lambda0 = vdw-q ; full interaction

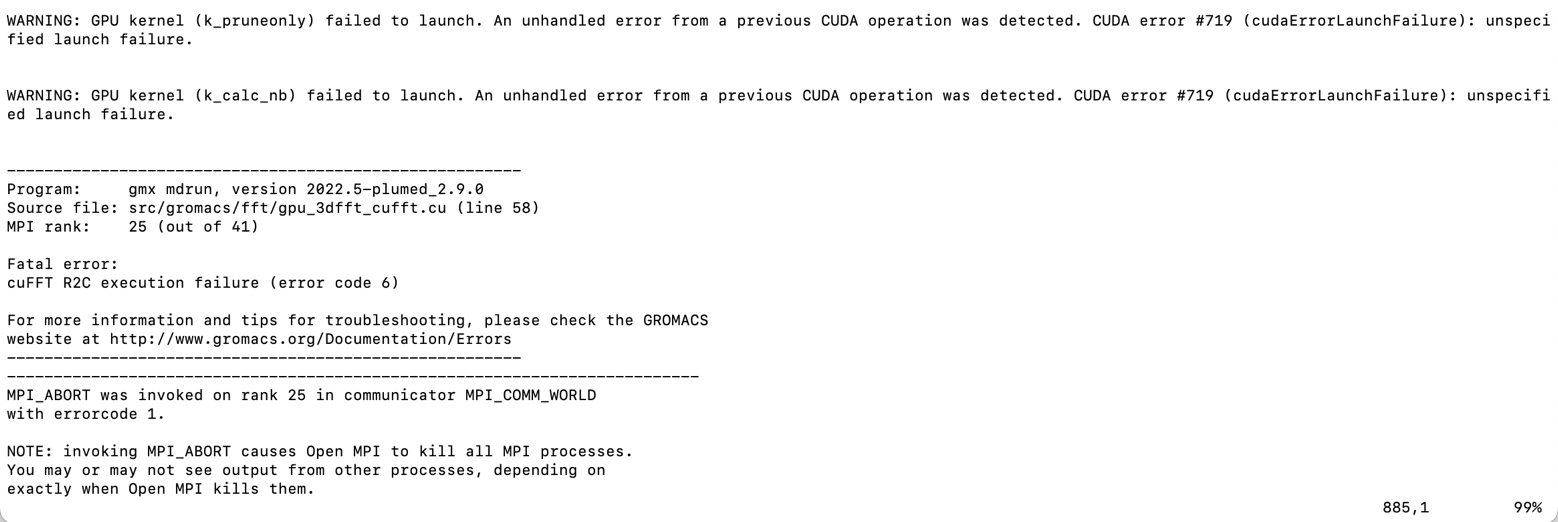
2. couple-lambda1 = none ; no interaction

3. couple-intramol = no

4.

## **10/04/2024**

While submitting the jobs for the Protein-LA binding energy with the decoupling approach with Hamiltonian replica exchange Molecular dynamics I got this error.



*If you go through the mdtutorials explanation regarding this type of setup, you will find that values of lambda only goes from 0 to 1 and perhaps cannot go from 1 to 0 (this is like double-counting!, that is why previous simulations worked and these are not working); you only take care of coupling vs. Decoupling using the interactions at intial and final state, i.e., using couple-lambda parameter. So lets try the following:*

**couple-lambda0** = vdw-q ; full interaction

**couple-lambda1** = none ; no interaction – which means you are decoupling

**coul\_lambdas**  = 0.00 0.05 0.10 0.15 0.20 0.25 0.30 0.35 0.40 0.45 0.50 0.55 0.60 0.65 0.70 0.75 0.80 0.85 0.90 0.95 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 ; decouple electrostatics first

**vdw\_lambdas**  = 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.05 0.10 0.15 0.20 0.25 0.30 0.35 0.40 0.45 0.50 0.55 0.60 0.65 0.70 0.75 0.80 0.85 0.90 0.95 1.00 ; decouple vdw after electrostatics are fully de-coupled

*If the above approach also does not work, then please point me to your simulation files and I would like to the files myself see what else we might be missing that is messing up the simulation.*

## **10/07/2024**

* Even previously, I followed the same framework we have used above with respect to the coul\_lambda’s and vdw-lambda’s. *Which was NOT correct, as your couple-lambda at the initial and final states were swapped! So essentially you were coupling and not de-coupling. It may feel similar because value of vdw/coul lambdas can only go in increasing order and not decreasing order. Either way, Recognize the difference!*
* But now it is working. I have not used the posre\_LA.itp file. I just created and then realized that it’s there in the ions.itp itself previously only.
* Used this command for numbering the residues from 1 with gmx\_mpi editconf resnr command. - **gmx\_mpi editconf -f init\_config.pdb -o init\_config\_modified.pdb -resnr 1**
* I have submitted the jobs for the 3 protein -LA binding interactions.
* Once it runs for more than 5ns probably will post the trajectory of the states.

## **10/18/2024**

Using the above approach computed the binding energies of the Lanthanide with respect to the loop attached in the protein using the decoupling method.

A close-up of a dna molecule

Description automatically generated

For this lanthanide position surrounded by the amino acids the binding energy is found to be **3457.45 +/- 3.47 kJ/mol**

A close-up of a dna molecule

Description automatically generated

For this lanthanide position surrounded by the amino acids the binding energy is found to be **3588.80 +/- 0.30 kJ/mol**

## **11/04/2024**

I completed the calculations for the Lanthanum (La+3) – and doing the analysis now. Currently few jobs are running for the Neodymium (Nd+3) ion. For the Yttrium (Y+3) the error is coming the same when I am using completely the 41 cores of the CPU without GPU.

## **11/07/2024**

|  |  |  |
| --- | --- | --- |
| **Sequence** | **Binding free energy** | **Ion and Ionic radii** |
| **NKDNDDSLE** | **3457.45 +/- 3.47 kJ/mol** | **La+ 3 – 1.16 Angstroms** |
| **AINPDGDTTLE** | **3588.17 +/- 0.30 kJ/mol** | **La+ 3 – 1.16 Angstroms** |
| **ANKDGDQTLEMD** | **3550.80 +/- 6.45 kJ/mol** | **La+ 3 – 1.16 Angstroms** |
| **DANKDGKLTAAE** | **479.17+/-0.48 kJ/mol** | **Na+ - 0.95 Angstroms** |
| **DANKDGKLTAAE** | **3255.26 +/- 1.07 kJ/mol** | **Na with La+3 – 1.16 Angstroms** |

Analysis for the first Lanthanide (La+3 ion) – sequence is **NKDNDDSLE**

* For the first one the nearest distance of the La+3 ion is to the **LA - GLU 42 – OE1 is 2.096 Å**
* For the first one the largest distance of the La+3 ion is to the **LA – GLU 42 - CA is 5.489 Å**
* For the same loop the nearest one is **LA – ASP 36 -OD2 is 1.950 Å**.
* For the same loop the La+3 is also having a closest contact to **ASN 34 HD21 – LA+3 and the distance is 1.586 Å.**

Analysis for the second Lanthanide (La+3 ion) – sequence is **AINPDGDTTLE**

* For the second La+3 ion firstly, it is binded to the **ASP 62 – OD1 --- LA+3 and the distance is 2.404 Å.**
* For the same La+3 ion it is also binded to **the GLU 66 OE1 --- LA+3 and the distance is 2.573 Å.**
* For the same La+3 ion the largest distance in that loop surrounding it is from the **GLU 66 - CA ---- LA+3 and the distance is 5.430 Å.**
* Finally, it is binded to the one more amino acid near to it is **ASN 58 – HD21 --- LA+3 and the distance between them is 2.981 Å.**

## **11/08/2024**

Analysis for the 3rd La+3 ion – sequence is **ANKDGDQTLEMD**

* For the third La+3 ion firstly, nearest to it is the **ASN 83 – OD1 and the distance of the La+3 from it is 2.461 Å.**
* After that it has the **ASP 87 - OD1** and the distance is 2.447 **Å.**
* The nearest distance in this loop is with the **GLU 91 – OE2** and the distance is 2.322 **Å.**
* The largest distance given the loop surrounding the La+3 ion is with the GLU 91 – CA and the distance is 5.405 **Å.**

**Key observations:**

1. Around the La+3 ion it has a loop around it and mainly 4 amino acids are crucially playing an important role in binding the Ion. From the above distances we can clearly see that ASN – Asparagine, GLU – Glutamate, and ASP – Aspartate and from all the three amino acids the oxygen atoms are nearest to the La+3 ion so my thought was that the lone pairs in the Oxygen atom of the amino acids are interacting with theLa+3 ion thus forms a binding complex surrounding those amino acids.

For the same sequence **NKDNDDSLE** instead of the La+3 ion if we have the Nd+3 ion the binding energy is **3533.24 +/- 2.93 kJ/mol.**

This is the image prepared with respect to the binding energies of the La+3 ion in the LanM protein.

A structure of a molecule

Description automatically generated with medium confidence

## **11/11/2024**

For the same sequence **NKDNDDSLE** instead of the La+3 ion if we have the Y+3 ion the binding energy is **3689.80 +/- 0.75 kJ/mol.**

## **11/12/2024**

|  |  |  |
| --- | --- | --- |
| **Sequence** | **Binding free energy** | **Ion and Ionic radii** |
| **N**K**DNDDSLEIAE** | **3533.24 +/- 2.93 kJ/mol.** | **Nd+ 3 – 1.12 Angstroms** |
| **NPDGDTTLESGE** | **3660.10 +/- 0.44 kJ/mol.** | **Nd+ 3 – 1.12 Angstroms** |
| **N**K**DGDQTLEMDE** | **3614.03 +/- 5.32 kJ/mol** | **Nd+ 3 – 1.12 Angstroms** |
| **DAN**K**DG**K**LTAAE** | **479.17+/-0.48 kJ/mol** | **Na+ - 0.95 Angstroms** |
| **DAN**K**DG**K**LTAAE** | **3327.26 +/- 0.88 kJ/mol** | **Na with Nd+3 – 1.12 Angstroms** |
| **Sequence** | **Binding free energy** | **Ion and Ionic radii** |
| **N**K**DNDDSLEIAE** | **3689.80 +/- 0.75 kJ/mol.** | **Y+ 3 – 1.03 Angstroms** |
| **NPDGDTTLESGE** | **3830.20 +/- 1.04 kJ/mol** | **Y+ 3 – 1.03 Angstroms** |
| **N**K**DGDQTLEMDE** | **3614.03 +/- 5.32 kJ/mol** | **Y+ 3 – 1.03 Angstroms** |
| **DAN**K**DG**K**LTAAE** | **479.17+/-0.48 kJ/mol** | **Na+ - 0.95 Angstroms** |

From the paper titled: **Lanmodulin’s EF 2−3 Domain: Insights from Infrared Spectroscopy and Simulations**

The MD structural analysis classified each ligand as monodentate, bidentate, intermediate, or unbound, using a two-step classification protocol that was used on LanM previously. Firstly, the α-carbon to ion distance of each ligand was calculated, with a distance larger than 0.4 nm being classified as ‘unbound.’ Then, the difference in oxygen-to-ion distance was used to classify each ligand as monodentate (>0.15 nm), bidentate (0−0.05nm), or intermediate (0.05−0.15 nm). Since we are not doing the MD Simulations indeed, we are only computing the binding free energy based on the position of the Lanthanide so classifying the ligand as monodentate, bidentate is not necessary. But we have the information of the distances of the alpha carbon from the ion and the oxygen-ion distance.

## **11/13/2024**

Points discussed and things to be done are:

1. Do the Hydrogen bond analysis for the state\_0 - trajectory of the HREMD simulation.
2. Do the analysis for pairwise interactions which contributes from Vander Waal and Electrostatic interactions.
3. Calculate the binding free energy for La+3 with only EF hand separately for every EF hand and compare with the binding energies when we have the entire structure.
4. Calculate the binding affinities for Na+ by replacing La+3 in the positions of EF 1-3 hands. – Submitted the jobs – Done
5. Use of Graph theory and clustering to see the interactions contributions among the residues.
6. Use of Meta-dynamics and Replica exchange temperature based with 16 replicas between 300-500K by making the protein as disordered as we can and from that final structure and the crystal structure we want to calculate the folding free energy.
7. As far as the analysis part is concerned do the sequence-based analysis like what are exactly EF 1-4 hands sequences and what contributing to what so that these will help us in mutating the protein sequence and calculate the binding affinity.

## **11/18/2024**

1. Submitted the jobs for the Na+ replacing the La+3 in the EF hands 1-3 where La+3 ion is binded.
2. From the paper below are the sequences for the EF hands in the Hans-LanM protein.

**EF hand-1: NKDNDDSLEIAE**

**EF hand-2: NPDGDTTLESGE**

**EF hand -3: NKDGDQTLEMDE**

**EF hand-4: DANKDGKLTAAE**

|  |  |  |
| --- | --- | --- |
| **Sequence** | **Binding Affinity** | **Ion and Ionic radii** |
| **N**K**DNDDSLEIAE** | **3457.45 +/- 3.47 kJ/mol** | **La+ 3 – 1.16 Angstroms** |
| **NPDGDTTLESGE** | **3588.17 +/- 0.30 kJ/mol** | **La+ 3 – 1.16 Angstroms** |
| **N**K**DGDQTLEMDE** | **3550.80 +/- 6.45 kJ/mol** | **La+ 3 – 1.16 Angstroms** |
| **DAN**K**DG**K**LTAAE** | **479.17+/-0.48 kJ/mol** | **Na+ - 0.95 Angstroms** |
| **DAN**K**DG**K**LTAAE** | **3255.26 +/- 1.07 kJ/mol** | **Na with La+3 – 1.16 Angstroms** |

## **11/25/2024**

Completed the 3 simulations to see if we replace the La3+ with Na+ in the EF hands 1-3 and see the binding affinity through Hamiltonian Replica Exchange Molecular Dynamics.

|  |  |  |
| --- | --- | --- |
| **Sequence** | **Binding Affinity** | **Ion and Ionic radii** |
| **N**K**DNDDSLEIAE** | **500.55 +/- 0.73 (kJ/mol)** | **Na+ - 0.95 Angstroms** |
| **NPDGDTTLESGE** | **531.77 +/- 0.76 (kJ/mol)** | **Na+ - 0.95 Angstroms** |
| **N**K**DGDQTLEMDE** | **547.49 +/- 1.61 (kJ/mol)** | **Na+ - 0.95 Angstroms** |

The binding affinity changes a lot from 3457 to 500.55 by changing the La3+ to Na+ in EF hands 1-3. So, the EF hands 1-3 selectively binding the REE’s.

In the EF hands 1-3 sequences the common sequence is: **N\_D\_D\_ \_LE\_ \_ E**

## **12/06/2024**