**MD simulation of PtxPi1kda in octanol**

**All-atom simulation**

1. **Model**

*/opt/jchem/bin/msketch ptxpi1k.cdx → ptxpi1k.mol2*

*chimera ptxpi1k.mol2 → add H and charge*

*acpype.py -i ptxpi1k.mol2 → ptxpi1kda\_GMX.gro and ptxpi1kda\_GMX.itp* *(ptxpi1kda.acpype)*

ssh

1. **Setup**

**### ./setup.sh ###**

# Place the solute in a simulation box

gmx editconf -f ptxpi1k\_GMX.gro -bt cubic -d 1.4 -o ptxpi1k\_inbox.gro

# Prepare minimization in vacuum

gmx grompp -f em\_vacuum.mdp -c ptxpi1k\_inbox.gro -p topol.top -o em\_vacuum.tpr

# Make minimization in vacuum

gmx mdrun -s em\_vacuum.tpr -deffnm em\_vacuum -v

# Fill the box with water

gmx insert-molecules -f em\_vacuum.gro -ci octanol\_GMX.gro -o ptxpi1k\_inslv.gro -nmol 5000

# Prepare the minimization in the solvent

gmx grompp -f em\_solvent.mdp -c ptxpi1k\_inslv.gro -p topol.top -o em\_solvent.tpr

edit topol.top: add octanol 1314 in topol.top (water don’t need)

# Make minimzation in the solvent

gmx mdrun -s em\_solvent.tpr -deffnm em\_solvent -v

1. **Equil**

**### ./equil.sh ###**

# Prepare Berendsen equilibration (1000ps)

gmx grompp -f berendsen.mdp -c em\_solvent.gro -p topol.top -o berendsen.tpr

# Make Berendsen equilibration

gmx mdrun -s berendsen.tpr -deffnm berendsen -v

# Prepare Nose-Hoover equilibration (1000ps)

gmx grompp -f nosehoover.mdp -c berendsen.gro -p topol.top -o nosehoover.tpr

# Make Nose-Hoover equilibration

gmx mdrun -s nosehoover.tpr -deffnm nosehoover -v

1. **Prod**

**### nohup ./prod.sh & ###**

# Prepare production run (1000ns)

gmx grompp -f md\_prod.mdp -c nosehoover.gro -p topol.top -o md\_prod.tpr

# Make production run

gmx mdrun -s md\_prod.tpr -deffnm md\_prod -v

**### ./rstr.sh ###**

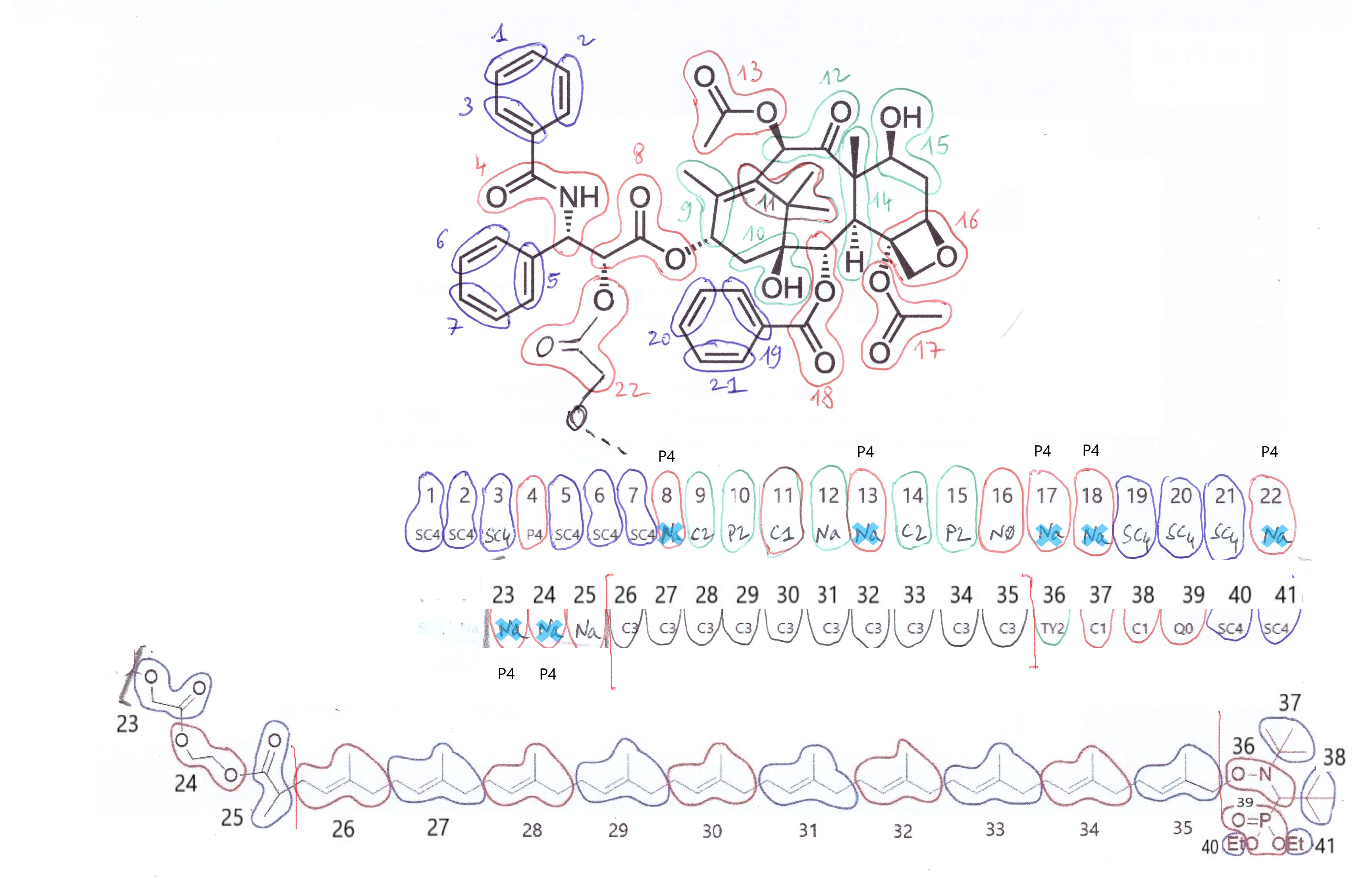
# Restart production run

gmx mdrun -s md\_prod.tpr -deffnm md\_prod -v -cpi md\_prod.cpt

**Mapping**

1.Download Prod/ and Alys/ files from byrd03

2. the mapping files from Tap

1. **Choose an atom-to-bead mapping, create mapped.ndx file**
2. **Put PtxPi1kda in the center of box**

**#### ./extr.sh ###**

*gmx convert-tpr -s md\_prod.tpr -o md\_solute.tpr*

*gmx trjconv -f nosehoover.gro -s md\_prod.tpr -o md\_solute.gro -center -pbc whole*

*gmx trjconv -f md\_prod.xtc -s md\_prod.tpr -o md\_solute.xtc -center -pbc whole*

1. Coarse-grain target atomistic data.

Create a **.itp** file with a directive for 46 "[bonds]" containing (multiple) pairs of CG beads, 69"[angles]" containing triples and 80"[dihedrals]" quartets.

**### ./mapp.sh ###**

*seq 0 40 |gmx traj -f md\_solute.gro -s md\_solute.tpr -oxt md\_mapped.gro -n mapped.ndx -com -ng 41*

*seq 0 40 |gmx traj -f md\_solute.xtc -s md\_solute.tpr -oxt md\_mapped.xtc -n mapped.ndx -com -ng 41*

*gmx grompp -f 1kda\_minim.mdp -c md\_mapped.gro -p 1kda\_mapped.top -o 1kda\_mapped.tpr*

*gmx mdrun -s 1kda\_mapped.tpr -deffnm 1kda\_mapped -v*

1. Bond and Angle analyse

1.Creat bonds.ndx and angles.ndx (make sure the order of bond and angle should correspond with 1kda\_mapped.itp)

**2. Analyze Bond and angle distribution** ( ./bondanalysis.sh > BONDAVER.dat)

**### ./bondanalysis.sh ###**

rm -rf BONDDISTRIBUTIONS

mkdir BONDDISTRIBUTIONS

NBONDS=46

IBOND=0

while [ $IBOND -lt $NBONDS ]

do

echo $IBOND | gmx distance -f md\_mapped.xtc -n bonds.ndx -s 1kda\_mapped.tpr -o all BONDDISTRIBUTIONS/bond\_$IBOND.xvg

gmx analyze -f BONDDISTRIBUTIONS/bond\_$IBOND.xvg -dist BONDDISTRIBUTIONS/distr\_$IBOND.xvg -bw 0.01

rm -rf \#\*

let IBOND=$IBOND+1

done

exit

**### ./angleanalysis.sh ###**

rm -rf ANGLEDISTRIBUTIONS

mkdir ANGLEDISTRIBUTIONS

NANGLES=69

IANGLE=0

while [ $IANGLE -lt $NANGLES ]

do

echo $IANGLE | gmx angle -f md\_mapped.xtc -n angles.ndx -type angle -ov ANGLEDISTRIBUTIONS/angle\_$IANGLE.xvg

gmx analyze -f ANGLEDISTRIBUTIONS/angle\_$IANGLE.xvg -dist ANGLEDISTRIBUTIONS/distr\_$IANGLE.xvg -bw 1.0

rm -rf \#\*

let IANGLE=$IANGLE+1

done

exit

To make sure the result of atomistic simulation by analyzing radius of gyration and RMSD:

*gmx trjconv -f nosehoover.gro -s md\_prod.tpr -o md\_solute\_nojump.gro -center -pbc nojump*

*gmx trjconv -f md\_prod.xtc -s md\_prod.tpr -o md\_solute\_nojump.xtc -center -pbc nojump*

*gmx rms -s md\_solute\_nojump.gro -f md\_solute\_nojump.xtc -o rmsd\_nojump.xvg*

*gmx gyrate -s md\_solute\_nojump.gro -f md\_solute\_nojump.xtc -o gyrate\_nojump.xvg*

**6. CG simulation**

1. Create the CG itp file

We can use the itp file of CG simulation in octanol

1. Mapping Octanol beads and GemPi1kda beads

For each octanol, there are 2 beads in one molecular

**### indexer.py ###**

#!/usr/bin/env python

nmols = 1138 # Number of molecules

atspermol = 9 # AA atoms per molecule

beadspermol = 2 # CG beads per molecule

firstbead = 4 # AA atoms in the first bead

atsperbead = 4 # AA atoms per CG bead

lastbead = 5 # AA atoms in the last bead

outname = "mapping.ndx" # Output filename

with open(outname,'w') as NDX:

for mol in range(nmols):

for bead in range(beadspermol):

# The number of atoms (nats) might depend on whether it's a first,

# last, or middle bead. NDXs are 1-based

if bead==0:

nats = firstbead

startat = 1

elif bead==beadspermol-1:

nats = lastbead

startat = atspermol-lastbead+1

else:

nats = atsperbead

startat = (bead-1)\*atsperbead + firstbead + 1

startat += mol\*atspermol

ats = range(startat, startat + nats)

ats = " ".join(map(str, ats))

NDX.write("[ mol%dbead%d ]\n%s\n" % (mol+1,bead+1, ats))

**7. Mapping all the octanol beads into the gro file**

**### ./mapp.sh ###**

*seq 0 2275 | gmx traj -f octanolnoh.gro -s octanolnoh.gro -oxt cg\_octanol.gro -n mapped.ndx -com -ng 2276*

where 2275 is the number of CG beads in your molecule and the 'seq 0 2274' avoids having to type all index groups. Inspect the file mapping.ndx to see how the atoms of toluene are grouped into the CG beads

**8. put cg\_ptxpi1kda.gro and cg\_octanol.gro into the same cg\_gemocta.gro, and edit cg\_gemocta.gro, detect the line between two files**

### ./mono.sh ###

*#!/bin/bash*

*cat md\_mapped.gro cg\_octanol.gro > cg\_ptxocta.gro*

*edit cg\_gemocta.gro*

**9. Create CG simulation**

Extract one frame from your mapped trajectory and solvate it in the same solvent as the atomistic simulation. **Create a top file** (include the general martini itp files and the newly created molecule itp and add the right number of molecules (solute and solvent))

**We can use the itp file of all-atom simulation in octanol**

**### ./cgmd.sh ###**

### minimization ###

*gmx grompp -f minim.mdp -c cg\_ptxocta.gro -p cg\_ptxocta.top -o minim.tpr*

*gmx mdrun -v -deffnm minim*

(if the atom is not cooperated, run the commands below)

*gmx grompp -f minim.mdp -c cg\_ptxocta.gro -p cg\_ptxocta.top -o minim.tpr -maxwarn 10)*

### relax the trajectory in the same solvent ###

*gmx grompp -f relax.mdp -c minim.gro -p cg\_ptxocta.top -o relax.tpr*

*gmx mdrun -v -deffnm relax*

### run MD simulation ###

*gmx grompp -f cg\_md.mdp -c relax.gro -p cg\_ptxocta.top -o cg\_md.tpr*

*gmx mdrun -v -deffnm cg\_md*

**10. Calculate the average and standard deviation for all the distributions**

### ./bondanalysis.sh > BONDAVER.dat###

### ./angleanalysis.sh > ANGLEAVER.dat###

**11. Analyze the bond distribution and angle distribution, change the value of itp files, repeat step 9 until get a good result.**

**After we get a good itp file, we use it to simulate 24 PtxPi1kda in the box to explore the aggregation of them.**

**REDO PtxPi1kda\_redo/**

**MD simulation of PtxPi1kda in octanol**

**All-atom simulation**

1. **Model**

*/opt/jchem/bin/msketch ptxpi1k.cdx → ptxpi1k.mol2*

*chimera ptxpi1k.mol2 → add H and charge*

***acpype.py -i ptxpi1k.mol2 -c user****→ ptxpi1kda.acpype (ptxpi1kda\_GMX.gro and ptxpi1kda\_GMX.itp)*

from Tap

1. **Setup**

**### ./setup.sh ###**

# Place the solute in a simulation box

gmx editconf -f ptxpi1k\_GMX.gro -bt cubic -d 1.4 -o ptxpi1k\_inbox.gro

# Prepare minimization in vacuum

gmx grompp -f em\_vacuum.mdp -c ptxpi1k\_inbox.gro -p topol.top -o em\_vacuum.tpr

# Make minimization in vacuum

gmx mdrun -s em\_vacuum.tpr -deffnm em\_vacuum -v

# Fill the box with water

gmx insert-molecules -f em\_vacuum.gro -ci octanol\_GMX.gro -o ptxpi1k\_inslv.gro -nmol 5000

# Prepare the minimization in the solvent

gmx grompp -f em\_solvent.mdp -c ptxpi1k\_inslv.gro -p topol.top -o em\_solvent.tpr

edit topol.top: add octanol **1294** in topol.top (water don’t need)

# Make minimzation in the solvent

gmx mdrun -s em\_solvent.tpr -deffnm em\_solvent -v

1. **Equil**

**### ./equil.sh ###**

# Prepare Berendsen equilibration (1000ps)

gmx grompp -f berendsen.mdp -c em\_solvent.gro -p topol.top -o berendsen.tpr

# Make Berendsen equilibration

gmx mdrun -s berendsen.tpr -deffnm berendsen -v

# Prepare Nose-Hoover equilibration (1000ps)

gmx grompp -f nosehoover.mdp -c berendsen.gro -p topol.top -o nosehoover.tpr

# Make Nose-Hoover equilibration

gmx mdrun -s nosehoover.tpr -deffnm nosehoover -v

1. **Prod**

**### nohup ./prod.sh & ###**

# Prepare production run (1000ns)

gmx grompp -f md\_prod.mdp -c nosehoover.gro -p topol.top -o md\_prod.tpr

# Make production run

gmx mdrun -s md\_prod.tpr -deffnm md\_prod -v

**### ./rstr.sh ###**

# Restart production run

gmx mdrun -s md\_prod.tpr -deffnm md\_prod -v -cpi md\_prod.cpt

**Mapping in PtxPi1kda\_redo/Alys/**

1. **Choose an atom-to-bead mapping, copy mapped.ndx file from PtxPi1kda/**
2. **Put PtxPi1kda in the center of box**

**#### ./extr.sh ###**

*gmx convert-tpr -s md\_prod.tpr -o md\_solute.tpr*

*gmx trjconv -f nosehoover.gro -s md\_prod.tpr -o md\_solute.gro -center -pbc whole*

*gmx trjconv -f md\_prod.xtc -s md\_prod.tpr -o md\_solute.xtc -center -pbc whole*

1. Coarse-grain target atomistic data.

Copy a **.itp** file from PtxPi1kda/ with a directive for 46 "[bonds]" containing (multiple) pairs of CG beads, 69"[angles]" containing triples and 80"[dihedrals]" quartets.

**### ./mapp.sh ###**

*seq 0 40 |gmx traj -f md\_solute.gro -s md\_solute.tpr -oxt md\_mapped.gro -n mapped.ndx -com -ng 41*

*seq 0 40 |gmx traj -f md\_solute.xtc -s md\_solute.tpr -oxt md\_mapped.xtc -n mapped.ndx -com -ng 41*

*gmx grompp -f 1kda\_minim.mdp -c md\_mapped.gro -p 1kda\_mapped.top -o 1kda\_mapped.tpr*

1. Bond and Angle analyse

1.copy **bonds.ndx** and **angles.ndx** from PtxPi1kda/ (make sure the order of bond and angle should correspond with 1kda\_mapped.itp)

**2. Analyze Bond and angle distribution** ( ./bondanalysis.sh > BONDAVER.dat)

**### ./bondanalysis.sh ###**

rm -rf BONDDISTRIBUTIONS

mkdir BONDDISTRIBUTIONS

NBONDS=46

IBOND=0

while [ $IBOND -lt $NBONDS ]

do

echo $IBOND | gmx distance -f md\_mapped.xtc -n bonds.ndx -s 1kda\_mapped.tpr -o all BONDDISTRIBUTIONS/bond\_$IBOND.xvg

gmx analyze -f BONDDISTRIBUTIONS/bond\_$IBOND.xvg -dist BONDDISTRIBUTIONS/distr\_$IBOND.xvg -bw 0.01

rm -rf \#\*

let IBOND=$IBOND+1

done

exit

**### ./angleanalysis.sh ###**

rm -rf ANGLEDISTRIBUTIONS

mkdir ANGLEDISTRIBUTIONS

NANGLES=69

IANGLE=0

while [ $IANGLE -lt $NANGLES ]

do

echo $IANGLE | gmx angle -f md\_mapped.xtc -n angles.ndx -type angle -ov ANGLEDISTRIBUTIONS/angle\_$IANGLE.xvg

gmx analyze -f ANGLEDISTRIBUTIONS/angle\_$IANGLE.xvg -dist ANGLEDISTRIBUTIONS/distr\_$IANGLE.xvg -bw 1.0

rm -rf \#\*

let IANGLE=$IANGLE+1

done

exit

**6. CG simulation in PtxPi1kda\_redo/Alys/CG/**

1. copy the CG itp file from PtxPi1kda/Take5/

We can use the itp file of CG simulation in octanol

1. Mapping Octanol beads and GemPi1kda beads

For each octanol, there are 2 beads in one molecular

**### indexer.py ###**

#!/usr/bin/env python

nmols = 1138 # Number of molecules

atspermol = 9 # AA atoms per molecule

beadspermol = 2 # CG beads per molecule

firstbead = 4 # AA atoms in the first bead

atsperbead = 4 # AA atoms per CG bead

lastbead = 5 # AA atoms in the last bead

outname = "mapped.ndx" # Output filename

with open(outname,'w') as NDX:

for mol in range(nmols):

for bead in range(beadspermol):

# The number of atoms (nats) might depend on whether it's a first,

# last, or middle bead. NDXs are 1-based

if bead==0:

nats = firstbead

startat = 1

elif bead==beadspermol-1:

nats = lastbead

startat = atspermol-lastbead+1

else:

nats = atsperbead

startat = (bead-1)\*atsperbead + firstbead + 1

startat += mol\*atspermol

ats = range(startat, startat + nats)

ats = " ".join(map(str, ats))

NDX.write("[ mol%dbead%d ]\n%s\n" % (mol+1,bead+1, ats))

**7. Mapping all the octanol beads into the gro file**

**### ./mapp.sh ###**

*seq 0 2275 | gmx traj -f octanolnoh.gro -s octanolnoh.gro -oxt cg\_octanol.gro -n mapped.ndx -com -ng 2276*

where 2275 is the number of CG beads in your molecule and the 'seq 0 2274' avoids having to type all index groups. Inspect the file mapping.ndx to see how the atoms of toluene are grouped into the CG beads

**8. put cg\_ptxpi1kda.gro (md\_mapped.gro) and cg\_octanol.gro into the same cg\_gemocta.gro, and edit cg\_gemocta.gro, detect the line between two files**

### ./mono.sh ###

*#!/bin/bash*

*cat* ***md\_mapped.gro*** *cg\_octanol.gro > cg\_ptxocta.gro*

*edit cg\_ptxocta.gro*

**9. Create CG simulation**

Extract one frame from your mapped trajectory and solvate it in the same solvent as the atomistic simulation. **Create a top file** (include the general martini itp files and the newly created molecule itp and add the right number of molecules (solute and solvent))

**We can use the itp file of all-atom simulation in octanol**

**### ./cgmd.sh ###**

### minimization ###

*gmx grompp -f minim.mdp -c cg\_ptxocta.gro -p cg\_ptxocta.top -o minim.tpr -maxwarn 1*

*gmx mdrun -v -deffnm minim*

### relax the trajectory in the same solvent ###

*gmx grompp -f relax.mdp -c minim.gro -p cg\_ptxocta.top -o relax.tpr*

*gmx mdrun -v -deffnm relax*

### run MD simulation ###

*gmx grompp -f cg\_md.mdp -c relax.gro -p cg\_ptxocta.top -o cg\_md.tpr*

*gmx mdrun -v -deffnm cg\_md*

**10. Calculate the average and standard deviation for all the distributions**

### ./bondanalysis.sh ###

### ./angleanalysis.sh ###

**11. Analyze the bond distribution and angle distribution, change the value of itp files, repeat step 9 until get a good result.**

**After we get a good itp file, we use it to simulate 24 PtxPi1kda in the box to explore the aggregation of them.**

**2. 1chain in water ( in byrd02, 3h)**

~/ProDrug/PtxPi1kDa\_redo/Alys/CG/Ptx1\_water

conditions:

1. 150mM NaCl + water
2. 1mM ProDrug (1kda)
3. 1microsecond expected
4. same 1kda\_mapped.itp file in octanol

Copy cg\_md.xtc, cg\_md.tpr, 1kda\_mapped\_1.itp from ../../PtxPi1kda\_redo/Alys/CG/

*#creat prodrug gro file#*

*gmx trjconv -f cg\_md.xtc -s cg\_md.tpr -o prodrug.gro*

# Place the solute in a simulation box

*gmx editconf -f prodrug.gro -bt cubic -d 1 -o prodrug\_inbox.gro*

# define the size of box

*gmx editconf -f prodrug\_inbox.gro -o system\_temp.gro -d 1.0 -bt cubic -box 11.84 11.84 11.84*

# Minimization in vaccum following by minimization in solution

*gmx grompp -f minimization-vac.mdp -c system\_temp.gro -p system.top -o minimization-vac.tpr*

*gmx mdrun -deffnm minimization-vac -v*

in system.top

Gempi1kda 1

# Solvatation

*gmx solvate -cp minimization-vac.gro -cs water.gro -radius 0.21 -o system\_W.gro*

*gmx grompp -f ions.mdp -c system\_W.gro -p system.top -o ions.tpr*

in system.top

Gempi1kda 1

W 13102

# Add ions into the box

*gmx genion -s ions.tpr -p system.top -neutral -conc 0.15 -pname NA+ -nname CL- -o system\_WI.gro*

Then select the water group, here it’s the #3.

# Add anti-freeze water into the box

Gempi1kda 1

;W 12802

W 11522

WF 1280

NA+ 150

CL - 150

change the last 1281 W to WF in *system\_WI.gro*

:11566,12845s/W W/WF WF/g

# minimization

*gmx grompp -f minimization.mdp -c system\_WI.gro -p system.top -o minimization.tpr -maxwarn 1*

*gmx mdrun -deffnm minimization -v*

*#define the set in equilibration.mdp#*

*vi equilibration.mdp, check the set as below;*

tc-grps = ProDrug Solvent

*gmx make\_ndx -f minimization.gro -o system.ndx*

*name 2(1kda) ProDrug*

*3(W) | 4(WF) | 5(ION)*

*name 6 Solvent*

# Equil

*gmx grompp -f equilibration.mdp -c minimization.gro -p system.top -o equilibration.tpr -n system.ndx*

gmx mdrun -deffnm equilibration -v

# Prod

*gmx grompp -f dynamic.mdp -c equilibration.gro -p system.top -o dynamic.tpr -n system.ndx*

*gmx mdrun -deffnm dynamic -v*

*Analysis*

#SASA#

*gmx sasa -f dynamic.xtc -s dynamic.tpr -n 1chain.ndx -o 1chain\_W\_sasa.xvg -or 1chain\_W\_res\_sas.xvg -pbc -surface -output -ndots 25*

#Average#

*awk '{sum+=sprintf("%f",$5)}END{printf "%.6f\n%.6f\n",sum,sum/NR}' 1chain\_W\_sasa.xvg*

***3. SAS of Just Ptx in water***

[*ping@jarre*](mailto:ping@jarre)*:~/ProDrug/Ptx\_PI\_SG1/PtxPi1kda\_redo/Solution/Ptx1/Biphase\_Ptx/SASA/Analysis\_1/*

*Analysis\_1 (from water to OCO)*

*ln -s ../../Biphase\_Ptx\_W\_OCO\_P4/Production/0.0/md.xtc md.gro md.tpr*

*Analysis\_2 (from OCO to water)*

*ln -s ../../Biphase\_Ptx\_OCO\_W/Production/11.8/md.tpr md.gro md.xtc*

*gmx make-ndx -f md.gro -o ptx.ndx*

*gmx sasa -f md.xtc -s md.tpr -n ptx.ndx -o Ptx\_W\_sasa.xvg -or Ptx\_W\_res\_sas.xvg -pbc -surface -output -ndots 25*

**Change the name of beads in 1kda\_mapped.itp and md\_mapped.gro files**

**Mapping in PtxPi1kda\_redo/Alys\_1/**

1. **Choose an atom-to-bead mapping, copy mapped.ndx file from PtxPi1kda/**
2. **Put PtxPi1kda in the center of box**

**#### ./extr.sh ###**

1. Coarse-grain target atomistic data.

Copy a **.itp** file from PtxPi1kda/ with a directive for 46 "[bonds]" containing (multiple) pairs of CG beads, 69"[angles]" containing triples and 80"[dihedrals]" quartets.

**### ./mapp.sh ###**

*seq 0 40 |gmx traj -f md\_solute.gro -s md\_solute.tpr -oxt md\_mapped.gro -n mapped.ndx -com -ng 41*

*seq 0 40 |gmx traj -f md\_solute.xtc -s md\_solute.tpr -oxt md\_mapped.xtc -n mapped.ndx -com -ng 41*

***edit md\_mapped.gro and 1kda\_mapped.itp***

*gmx grompp -f 1kda\_minim.mdp -c md\_mapped.gro -p 1kda\_mapped.top -o 1kda\_mapped.tpr*

1. Bond and Angle analyse

1.copy **bonds.ndx** and **angles.ndx** from PtxPi1kda/ (make sure the order of bond and angle should correspond with 1kda\_mapped.itp)

**2. Analyze Bond and angle distribution** ( ./bondanalysis.sh > BONDAVER.dat)

**### ./bondanalysis.sh ###**

rm -rf BONDDISTRIBUTIONS

mkdir BONDDISTRIBUTIONS

NBONDS=46

IBOND=0

while [ $IBOND -lt $NBONDS ]

do

echo $IBOND | gmx distance -f md\_mapped.gro -n bonds.ndx -s 1kda\_mapped.tpr -o all BONDDISTRIBUTIONS/bond\_$IBOND.xvg

gmx analyze -f BONDDISTRIBUTIONS/bond\_$IBOND.xvg -dist BONDDISTRIBUTIONS/distr\_$IBOND.xvg -bw 0.01

rm -rf \#\*

let IBOND=$IBOND+1

done

exit

**### ./angleanalysis.sh ###**

rm -rf ANGLEDISTRIBUTIONS

mkdir ANGLEDISTRIBUTIONS

NANGLES=69

IANGLE=0

while [ $IANGLE -lt $NANGLES ]

do

echo $IANGLE | gmx angle -f md\_mapped.gro -n angles.ndx -type angle -ov ANGLEDISTRIBUTIONS/angle\_$IANGLE.xvg

gmx analyze -f ANGLEDISTRIBUTIONS/angle\_$IANGLE.xvg -dist ANGLEDISTRIBUTIONS/distr\_$IANGLE.xvg -bw 1.0

rm -rf \#\*

let IANGLE=$IANGLE+1

done

exit

**6. CG simulation in PtxPi1kda\_redo/Alys\_1/CG/**

1. copy the CG itp file from PtxPi1kda/Take5/

We can use the itp file of CG simulation in octanol

1. Mapping Octanol beads and GemPi1kda beads

For each octanol, there are 2 beads in one molecular

**### indexer.py ###**

#!/usr/bin/env python

nmols = 1138 # Number of molecules

atspermol = 9 # AA atoms per molecule

beadspermol = 2 # CG beads per molecule

firstbead = 4 # AA atoms in the first bead

atsperbead = 4 # AA atoms per CG bead

lastbead = 5 # AA atoms in the last bead

outname = "mapped.ndx" # Output filename

with open(outname,'w') as NDX:

for mol in range(nmols):

for bead in range(beadspermol):

# The number of atoms (nats) might depend on whether it's a first,

# last, or middle bead. NDXs are 1-based

if bead==0:

nats = firstbead

startat = 1

elif bead==beadspermol-1:

nats = lastbead

startat = atspermol-lastbead+1

else:

nats = atsperbead

startat = (bead-1)\*atsperbead + firstbead + 1

startat += mol\*atspermol

ats = range(startat, startat + nats)

ats = " ".join(map(str, ats))

NDX.write("[ mol%dbead%d ]\n%s\n" % (mol+1,bead+1, ats))

**7. Mapping all the octanol beads into the gro file**

**### ./mapp.sh ###**

*seq 0 2275 | gmx traj -f octanolnoh.gro -s octanolnoh.gro -oxt cg\_octanol.gro -n mapped.ndx -com -ng 2276*

where 2275 is the number of CG beads in your molecule and the 'seq 0 2274' avoids having to type all index groups. Inspect the file mapping.ndx to see how the atoms of toluene are grouped into the CG beads

**8. put cg\_ptxpi1kda.gro (md\_mapped\_1.gro) and cg\_octanol.gro into the same cg\_gemocta.gro, and edit cg\_gemocta.gro, detect the line between two files**

### ./mono.sh ###

*#!/bin/bash*

*cat md\_mapped.gro cg\_octanol.gro > cg\_ptxocta.gro*

*edit cg\_ptxocta.gro*

**9.  Create CG simulation**

Extract one frame from your mapped trajectory and solvate it in the same solvent as the atomistic simulation. **Create a top file** (include the general martini itp files and the newly created molecule itp and add the right number of molecules (solute and solvent))

**We can use the itp file of all-atom simulation in octanol**

**### ./cgmd.sh ###**

### minimization ###

*gmx grompp -f minim.mdp -c cg\_ptxocta.gro -p cg\_ptxocta.top -o minim.tpr -maxwarn 1*

*gmx mdrun -v -deffnm minim*

### relax the trajectory in the same solvent ###

*gmx grompp -f relax.mdp -c minim.gro -p cg\_ptxocta.top -o relax.tpr*

*gmx mdrun -v -deffnm relax*

### run MD simulation ###

*gmx grompp -f cg\_md.mdp -c relax.gro -p cg\_ptxocta.top -o cg\_md.tpr*

*gmx mdrun -v -deffnm cg\_md*

**10. Calculate the average and standard deviation for all the distributions**

### ./bondanalysis.sh ###

### ./angleanalysis.sh ###

**11. Analyze the bond distribution and angle distribution, change the value of itp files, repeat step 9 until get a good result.**

**After we get a good itp file, we use it to simulate 24 PtxPi1kda in the box to explore the aggregation of them.**

**~/ProDrug/Ptx\_PI\_SG1/PtxPi1kDa\_redo/Solution/Ptx1\_water**

2. 1chain in water ( jarre, 6h)

conditions:

1. 150mM NaCl + water
2. 1mM ProDrug (1kda)
3. 1microsecond expected
4. same 1kda\_mapped\_1.itp file in octanol

Copy cg\_md.xtc, cg\_md.tpr, 1kda\_mapped\_1.itp from ../../PtxPi1kda\_redo/Alys\_1/CG/

*#creat prodrug gro file#*

*gmx trjconv -f cg\_md.xtc -s cg\_md.tpr -o prodrug.gro*

# Place the solute in a simulation box

*gmx editconf -f prodrug.gro -bt cubic -d 1 -o prodrug\_inbox.gro*

# define the size of box

*gmx editconf -f prodrug\_inbox.gro -o system\_temp.gro -d 1.0 -bt cubic -box 11.84 11.84 11.84*

# Minimization in vaccum following by minimization in solution

*gmx grompp -f minimization-vac.mdp -c system\_temp.gro -p system.top -o minimization-vac.tpr*

*gmx mdrun -deffnm minimization-vac -v*

in system.top

Gempi1kda 1

# Solvatation

*gmx solvate -cp minimization-vac.gro -cs water.gro -radius 0.21 -o system\_W.gro*

*gmx grompp -f ions.mdp -c system\_W.gro -p system.top -o ions.tpr*

in system.top

Gempi1kda 1

W 13107

# Add ions into the box

*gmx genion -s ions.tpr -p system.top -neutral -conc 0.15 -pname NA+ -nname CL- -o system\_WI.gro*

Then select the water group, here it’s the #3.

# Add anti-freeze water into the box

Gempi1kda 1

;W 12807

W 11526

WF 1281

NA+ 150

CL - 150

change the last 1281 W to WF in *system\_WI.gro*

:11570,12850s/W W/WF WF/g

# minimization

*gmx grompp -f minimization.mdp -c system\_WI.gro -p system.top -o minimization.tpr -maxwarn 1*

*gmx mdrun -deffnm minimization -v*

*#define the set in equilibration.mdp#*

*vi equilibration.mdp, check the set as below;*

tc-grps = ProDrug Solvent

*gmx make\_ndx -f minimization.gro -o system.ndx*

*name 2(1kda) ProDrug*

*3(W) | 4(WF) | 5(ION)*

*name 6 Solvent*

# Equil

*gmx grompp -f equilibration.mdp -c minimization.gro -p system.top -o equilibration.tpr -n system.ndx*

gmx mdrun -deffnm equilibration -v

# Prod

*gmx grompp -f dynamic.mdp -c equilibration.gro -p system.top -o dynamic.tpr -n system.ndx*

*gmx mdrun -deffnm dynamic -v*

**#SASA#** *in Ptx1\_water/Analysis*

*gmx make\_ndx -f ../dynamic.gro -o 1chain.ndx*

*a C64 C60 C62 C66 C68 C74 C72 C69 C76 C77 C79 C82 O9 C87 C88 C91 O17 C93 C95 C98 C97 O7 C109 O21 C111*

group 6, all the beads of Ptx

*a O1 C56 C52 C116 C118 P1*

group 7, al the beads of SG1

*a C64 C74 C79 C91 C97 C64 C64 C36 C41 C46*

group 8, all the beads of Link

*a C111*

*gourp 9, all the Amide*

*gmx sasa -f ../dynamic.xtc -s ../dynamic.tpr -n 1chain.ndx -o 1chain\_W\_sasa.xvg -or 1chain\_W\_res\_sas.xvg -pbc -surface -output -ndots 25*

surface:1kda

output:Ptx, SG1,Link,Amide

SAS\_all.xvg is the All SAS distribute as function of times

*RES\_SAS\_all.xvg is the average SAS distribution as function of chains*

#Average#

*awk '{sum+=sprintf("%f",$5)}END{printf "%.6f\n%.6f\n",sum,sum/NR}' 1chain\_W\_sasa.xvg*

#Average#

*awk '{sum+=sprintf("%f",$5)}END{printf "%.6f\n%.6f\n",sum,sum/NR}' 1chain\_W\_sasa.xvg*

*#radius of gyration# (0,668 nm)*

*gmx trjconv -f ../dynamic.xtc -s ../dynamic.tpr -o cluster.gro -pbc cluster*

*gmx convert-tpr -s ../dynamic.tpr -o cluster.tpr*

*gmx gyrate -f cluster.gro -s cluster.tpr -n 1chain.ndx -o gyrate.xvg*

***average:*** *cat gyrate.xvg|awk '{sum+=$2} END {print "Average = ", sum/NR}'*

*#Radical distribution#*

*gmx rdf -selrpos mol\_com -seltype res\_com -ref 'group 2' -sel 'group 6 ' -n 1chain.ndx -o rdf\_COM\_Ptx.xvg -f ../dynamic.gro -s ../dynamic.tpr -bin 0.05 -rmax 5*

*gmx rdf -selrpos mol\_com -seltype res\_com -ref 'group 2' -sel 'group 7' -n 1chain.ndx -o rdf\_COM\_SG1.xvg -f ../dynamic.gro -s ../dynamic.tpr -bin 0.05 -rmax 5*

*gmx rdf -selrpos mol\_com -seltype res\_com -ref 'group 2' -sel 'group 8' -n 1chain.ndx -o rdf\_COM\_Link.xvg -f ../dynamic.gro -s ../dynamic.tpr -bin 0.05 -rmax 5*

gmx rdf -selrpos mol\_com -seltype res\_com -ref 'group 2' -sel 'group 9' -n 1chain.ndx -o rdf\_COM\_amide.xvg -f ../dynamic.gro -s ../dynamic.tpr -bin 0.05 -rmax 5

***Biphase\_Ptx(water to octanol)***

**# put Ptx in the center of box**

gmx make\_ndx -f dynamic.gro -o Ptx.ndx

*gmx trjconv -f dynamic.xtc -s dynamic.tpr -o Ptx.gro -center -pbc mol -n Ptx.ndx*

*Select group for centering: 6;(Ptx)*

*Select group for output: 8 (Ptx\_Solvent)*

***~~#Ptx minimization~~***

*~~gmx grompp -f minimization.mdp -c Ptx.gro -p Ptx.top -o Ptx\_minim.tpr -n Ptx.ndx -maxwarn 1~~*

*~~gmx mdrun -deffnm Ptx\_minim -v~~*

*~~gmx grompp -f equilibration.mdp -c Ptx\_minim.gro -p Ptx.top -o Ptx\_equil.tpr -n Ptx\_equil.ndx -maxwarn 1~~*

*~~gmx mdrun -deffnm Ptx\_equil -v~~*

*~~gmx trjconv -f Ptx\_equil.xtc -s Ptx\_equil.tpr -o Ptx\_center.gro -center -pbc mol -n Ptx\_equil.ndx~~*

*~~Select group for centering: 2;(Ptx)~~*

*~~Select group for output: 0 (system)~~*

**#extend the box size (*11.84 11.84 23.68*)**

**and define the position of Ptx\_water (5.92 5.92 5,92)**

*gmx editconf -f Ptx.gro -o Ptx\_newbox.gro -box 11.84 11.84 23.68 -center 5.92 5.92 5.92*

**#creat octanol box (size *11.84 11.84 11.84)***

*gmx solvate -cp cg\_octanol.gro -cs cg\_octanol.gro -o octanol\_newbox.gro -box 11.84 11.84 11.84*

*NB: cg\_octanol.gro is from CG simulation f 1 chain in octanol*

***#octanol minimization***

*gmx grompp -f minimization.mdp -c octanol\_newbox.gro -p octanol.top -o oct\_minim.tpr -n index.ndx -maxwarn 1*

*gmx mdrun -deffnm oct\_minim -v*

***#octanol equilibration***

*gmx grompp -f oct\_equil.mdp -c oct\_minim.gro -p octanol.top -o oct\_equil.tpr -n index.ndx -maxwarn 1*

*gmx mdrun -deffnm oct\_equil -v*

***#extend octanol box with same dimensions of Ptx1\_water (11.84 11.84 23.68)***

*gmx editconf -f oct\_equil.gro -o octanol\_new.gro -box 11.84 11.84 23.68 -center 5.92 5.92 17.76*

***#assemble two box***

gmx solvate -cp Ptx\_newbox.gro -cs octanol\_new.gro -o Ptx\_Solv.gro

*> gmx make\_ndx -f Ptx\_Solv.gro -o system.ndx*

*name 2(1kda) ProDrug*

*3(W) | 4(WF) | 5(ION) | 6 (OCO) ------- Solvent*

***3(W) | 4(WF) | 5(ION) ------------ WATER***

***# Biphase minimization***

*gmx grompp -f minimization.mdp -c Ptx\_Solv.gro -p system.top -o minimization.tpr -n system.ndx -maxwarn 1*

*gmx mdrun -deffnm minimization -v*

***# position restraint***

*gmx genrestr -f minimization.gro -n system.ndx -o Ptx\_res.itp -fc 100000 100000 100000*

***# Biphase equilibration***

*gmx grompp -f* ***equilibration****.mdp -c minimization.gro -p system.top -o* ***equilibration****.tpr -n system.ndx*

*gmx mdrun -deffnm* ***equilibration*** *-v*

***Umbrella sampling***

***#define the em.mdp, em2.mdp, eq.mdp file.***

*Add restraint conformation and pull code*

***; Bond parameters***

*continuation = no ; continuing from NPT*

*constraints =* ***none***

*constraint-algorithm = lincs*

***; pressure***

*Pcoupl = parrinello-rahman*

*Pcoupltype =* ***semiisotropic*** *; semiisotropic*

*tau-p = 12.0*

*compressibility =* ***4e-5 4e-5*** *;3e-4*

*ref-p =* ***1.0 1.0*** *;1.0*

***; Pull code***

*pull = yes*

*pull\_ncoords = 1 ; only one reaction coordinate*

*pull\_ngroups = 2 ; two groups defining one reaction coordinate*

*pull\_group1\_name = WATER*

*pull\_group2\_name = ProDrug*

*pull\_coord1\_type = umbrella ; harmonic potential*

*pull\_coord1\_geometry = distance ; simple distance increase*

*pull\_coord1\_dim = N N Y*

*pull\_coord1\_groups = 1 2*

***pull\_coord1\_start = no ; define initial COM distance > 0***

*pull-coord1-init = 0.0*

***pull\_coord1\_rate = 0,0 ; 0.01 nm per ps = 10 nm per ns***

*pull\_coord1\_k = 1000 ; kJ mol^-1 nm^-2*

***pull-pbc-ref-prev-step-com = yes***

***pull-group1-pbcatom = 8627***

***#pull.mdp#***

*> gmx grompp -f pull.mdp -c ../minimization.gro -p ../../system.top -o pull.tpr -n ../system.ndx*

*> gmx mdrun -nt 1 -deffnm pull -v &> pullrun.log*

**#em.mdp#**

*> gmx grompp -f em.mdp -c pull.gro -p ../../system.top -o em.tpr -n ../system.ndx*

*> gmx mdrun -nt 1 -deffnm em -v &> emrun.log*

**#em2.mdp#**

*> gmx grompp -f em2.mdp -c em.gro -p ../../system.top -o em2.tpr -n ../system.ndx*

*> gmx mdrun -nt 1 -deffnm em2 -v &> em2run.log*

***#eq.mdp#***

*> gmx grompp -f eq.mdp -c em2.gro -p ../../system.top -o eq.tpr -n ../system.ndx*

*> gmx mdrun -nt 1 -deffnm eq -v -px eq\_x -pf eq\_f &> eqrun.log*

*gmx grompp -f pull.mdp -c ../equilibration.gro -p ../../Parameters/system.top -o pull.tpr -n ../system.ndx*

*gmx mdrun -nt 1 -deffnm pull -v &> pullrun.log*

*gmx grompp -f em.mdp -c pull.gro -p ../../Parameters/system.top -o em.tpr -n ../system.ndx*

*gmx mdrun -nt 1 -deffnm em -v &> emrun.log*

*gmx grompp -f em2.mdp -c em.gro -p ../../Parameters/system.top -o em2.tpr -n ../system.ndx*

*gmx mdrun -nt 1 -deffnm em2 -v &> em2run.log*

*gmx grompp -f eq.mdp -c em2.gro -p ../../Parameters/system.top -o eq.tpr -n ../system.ndx*

*gmx mdrun -nt 1 -deffnm eq -v -px eq\_x -pf eq\_f &> eqrun.log*

***Production***

*gmx grompp -f md.mdp -c ../../Initial/$window/eq.gro -p ../../Parameters/system.top -o md.tpr -n ../../Initial/system.ndx*

*gmx mdrun -nt 2 -deffnm md -v -px md\_x -pf md\_f -cpi md.cpt &> mdrun.log*

***Analysis***

***ls -d ../Production/\*/md.tpr > tpr-files.dat***

**ls -d ../Production/\*/md\_x.xvg > pullx-files.dat**

***gmx wham -ix pullx-files.dat -it tpr-files.dat -bsres -bins 200 -temp 300 -unit kJ -b 100 -nBootstrap 100 -zprof0 0.0 -min 0 -max 13***

***Ptx from OCO to Water (change beads type(8, 13,17,18) from Na to P4)***

**1;** cp **eq.gro** from*Biphase\_Ptx\_W\_OCO\_P4/Initial/11.6*

***Umbrella sampling***

***# copy em.mdp, em2.mdp, eq.mdp file.***

*Add restraint conformation and pull code*

***; Bond parameters***

*continuation = no ; continuing from NPT*

*constraints =* ***none***

*constraint-algorithm = lincs*

***; Pull code***

*pull = yes*

*pull\_ncoords = 1 ; only one reaction coordinate*

*pull\_ngroups = 2 ; two groups defining one reaction coordinate*

***pull\_group1\_name = OCO***

***pull\_group2\_name = ProDrug***

*pull\_coord1\_type = umbrella ; harmonic potential*

*pull\_coord1\_geometry = distance ; simple distance increase*

*pull\_coord1\_dim = N N Y*

*pull\_coord1\_groups = 1 2*

***pull\_coord1\_start = no ; define initial COM distance > 0***

*pull-coord1-init = 0.0*

***pull\_coord1\_rate = 0,0 ; 0.01 nm per ps = 10 nm per ns***

*pull\_coord1\_k = 1000 ; kJ mol^-1 nm^-2*

***pull-pbc-ref-prev-step-com = yes***

***pull-group1-pbcatom = 22425***

***#pull.mdp#***

*> gmx grompp -f pull.mdp -c ../eq.gro -p ../../Parameters/system.top -o pull.tpr -n ../system.ndx*

*> gmx mdrun -nt 1 -deffnm pull -v &> pullrun.log*

**#em.mdp#**

*> gmx grompp -f em.mdp -c pull.gro -p ../../system.top -o em.tpr -n ../system.ndx*

*> gmx mdrun -nt 1 -deffnm em -v &> emrun.log*

**#em2.mdp#**

*> gmx grompp -f em2.mdp -c em.gro -p ../../system.top -o em2.tpr -n ../system.ndx*

*> gmx mdrun -nt 1 -deffnm em2 -v &> em2run.log*

***#eq.mdp#***

*> gmx grompp -f eq.mdp -c em2.gro -p ../../system.top -o eq.tpr -n ../system.ndx*

*> gmx mdrun -nt 1 -deffnm eq -v -px eq\_x -pf eq\_f &> eqrun.log*

*gmx grompp -f pull.mdp -c ../eq.gro -p ../../Parameters/system.top -o pull.tpr -n ../system.ndx*

*gmx mdrun -nt 1 -deffnm pull -v &> pullrun.log*

*gmx grompp -f em.mdp -c pull.gro -p ../../Parameters/system.top -o em.tpr -n ../system.ndx*

*gmx mdrun -nt 1 -deffnm em -v &> emrun.log*

*gmx grompp -f em2.mdp -c em.gro -p ../../Parameters/system.top -o em2.tpr -n ../system.ndx*

*gmx mdrun -nt 1 -deffnm em2 -v &> em2run.log*

*gmx grompp -f eq.mdp -c em2.gro -p ../../Parameters/system.top -o eq.tpr -n ../system.ndx*

*gmx mdrun -nt 1 -deffnm eq -v -px eq\_x -pf eq\_f &> eqrun.log*

***Production***

*gmx grompp -f md.mdp -c ../../Initial/$window/eq.gro -p ../../Parameters/system.top -o md.tpr -n ../../Initial/system.ndx*

*gmx mdrun -nt 2 -deffnm md -v -px md\_x -pf md\_f -cpi md.cpt &> mdrun.log*

***Analysis***

***ls -d ../Production/\*/md.tpr > tpr-files.dat***

**ls -d ../Production/\*/md\_x.xvg > pullx-files.dat**

***gmx wham -ix pullx-files.dat -it tpr-files.dat -bsres -bins 200 -temp 300 -unit kJ -b 100 -nBootstrap 100 -zprof0 0.0 -min 0 -max 12***

***Biphase\_PtxPi1kda(water to octanol)***

***ln -s ../Ptx1\_water\_redo/dynamic.xtc***

***ln -s ../Ptx1\_water\_redo/dynamic.tpr***

***ln -s ../Ptx1\_water\_redo/dynamic.gro***

**# put PtxPi1kda in the center of box**

**gmx make\_ndx -f dynamic.gro -o PtxPi1kda.ndx**

***gmx trjconv -f dynamic.xtc -s dynamic.tpr -o PtxPi1kda.gro -center -pbc mol -n PtxPi1kda.ndx***

*Select group for centering: 2 ;(1kda)*

*Select group for output: 0 (system)*

**#extend the box size (*11.84 11.84 23.68*)**

**and define the position of PtxPi1kda (5.92 5.92 5,92)**

*gmx editconf -f PtxPi1kda.gro -o Ptx\_newbox.gro -box 11.84 11.84 23.68 -center 5.92 5.92 5.92*

*Note: cope* ***octanol\_new.gro*** *from ../Biphase\_Ptx/Biphase\_Ptx\_W\_OCO\_P4/*

***#assemble two box***

gmx solvate -cp Ptx\_newbox.gro -cs octanol\_new.gro -o Ptx\_Solv.gro

*> gmx make\_ndx -f Ptx\_Solv.gro -o system.ndx*

*name 2(1kda) ProDrug*

*3(W) | 4(WF) | 5(ION) | 6 (OCO) ------- Solvent*

***3(W) | 4(WF) | 5(ION) ------------ WATER***

***# Biphase minimization***

*gmx grompp -f minimization.mdp -c Ptx\_Solv.gro -p system.top -o minimization.tpr -n system.ndx -maxwarn 1*

*gmx mdrun -deffnm minimization -v*

***# position restraint***

*gmx genrestr -f minimization.gro -n system.ndx -o Ptxpi1kda\_res.itp*

***edit system.top***

*; Ligand position restraints*

*#ifdef POSRES\_ptxpi1kda*

*#include "Ptxpi1kda\_res.itp"*

*#endif*

***edit equilibration.mdp***

*define = -DSTRONG\_POSRES\_ptxpi1kda*

***# Biphase equilibration***

*gmx grompp -f* ***equilibration****.mdp -c minimization.gro -p system.top -o* ***equilibration****.tpr -n system.ndx*

*gmx mdrun -deffnm* ***equilibration*** *-v*

***Umbrella sampling***

***#define the em.mdp, em2.mdp, eq.mdp file.***

*Add restraint conformation and pull code*

***; Bond parameters***

*continuation = no ; continuing from NPT*

*constraints =* ***none***

*constraint-algorithm = lincs*

***; pressure***

*Pcoupl = parrinello-rahman*

*Pcoupltype =* ***semiisotropic*** *; semiisotropic*

*tau-p = 12.0*

*compressibility =* ***4e-5 4e-5*** *;3e-4*

*ref-p =* ***1.0 1.0*** *;1.0*

***; Pull code***

*pull = yes*

*pull\_ncoords = 1 ; only one reaction coordinate*

*pull\_ngroups = 2 ; two groups defining one reaction coordinate*

*pull\_group1\_name = WATER*

*pull\_group2\_name = ProDrug*

*pull\_coord1\_type = umbrella ; harmonic potential*

*pull\_coord1\_geometry = distance ; simple distance increase*

*pull\_coord1\_dim = N N Y*

*pull\_coord1\_groups = 1 2*

***pull\_coord1\_start = no ; define initial COM distance > 0***

*pull-coord1-init = 0.0*

***pull\_coord1\_rate = 0,0 ; 0.01 nm per ps = 10 nm per ns***

*pull\_coord1\_k = 1000 ; kJ mol^-1 nm^-2*

***pull-pbc-ref-prev-step-com = yes***

***pull-group1-pbcatom = 396***

***#pull.mdp#***

*> gmx grompp -f pull.mdp -c ../minimization.gro -p ../../system.top -o pull.tpr -n ../system.ndx*

*> gmx mdrun -nt 1 -deffnm pull -v &> pullrun.log*

**#em.mdp#**

*> gmx grompp -f em.mdp -c pull.gro -p ../../system.top -o em.tpr -n ../system.ndx*

*> gmx mdrun -nt 1 -deffnm em -v &> emrun.log*

**#em2.mdp#**

*> gmx grompp -f em2.mdp -c em.gro -p ../../system.top -o em2.tpr -n ../system.ndx*

*> gmx mdrun -nt 1 -deffnm em2 -v &> em2run.log*

***#eq.mdp#***

*> gmx grompp -f eq.mdp -c em2.gro -p ../../system.top -o eq.tpr -n ../system.ndx*

*> gmx mdrun -nt 1 -deffnm eq -v -px eq\_x -pf eq\_f &> eqrun.log*

*gmx grompp -f pull.mdp -c ../equilibration.gro -p ../../Parameters/system.top -o pull.tpr -n ../system.ndx*

*gmx mdrun -nt 1 -deffnm pull -v &> pullrun.log*

*gmx grompp -f em.mdp -c pull.gro -p ../../Parameters/system.top -o em.tpr -n ../system.ndx*

*gmx mdrun -nt 1 -deffnm em -v &> emrun.log*

*gmx grompp -f em2.mdp -c em.gro -p ../../Parameters/system.top -o em2.tpr -n ../system.ndx*

*gmx mdrun -nt 1 -deffnm em2 -v &> em2run.log*

*gmx grompp -f eq.mdp -c em2.gro -p ../../Parameters/system.top -o eq.tpr -n ../system.ndx*

*gmx mdrun -nt 1 -deffnm eq -v -px eq\_x -pf eq\_f &> eqrun.log*

***Production***

*gmx grompp -f md.mdp -c ../../Initial/$window/eq.gro -p ../../Parameters/system.top -o md.tpr -n ../../Initial/system.ndx*

*gmx mdrun -nt 2 -deffnm md -v -px md\_x -pf md\_f -cpi md.cpt &> mdrun.log*

***Analysis***

***ls -d ../Production/\*/md.tpr > tpr-files.dat***

***ls -d ../Production/\*/md\_x.xvg > pullx-files.dat***

***gmx wham -ix pullx-files.dat -it tpr-files.dat -bsres -bins 200 -temp 300 -unit kJ -b 100 -nBootstrap 100 -zprof0 0.0 -min 0 -max 12***

***PtxPi1kda from OCO to Water (bead types P4)***

**1;** cp **eq.gro** from*Biphase\_PtxPi1kda\_W\_OCO\_P4/Initial/12.0*

***# Biphase equilibration***

***postion restaint***

*1.gmx genrestr -f eq.gro -n system.ndx -o Ptxpi1kda\_res.itp*

*gmx genrestr -f minimization.gro -n system.ndx -o Gem\_res.itp -fc 100000 100000 100000*

*2. edit equilibration.mdp*

*define = DSTRONG\_POSRES\_ptxpi1kda*

*3. edit system.top ;*

*Ligand position restraints*

*#ifdef POSRES\_ptxpi1kda*

*#include "Ptxpi1kda\_res.itp"*

*#endif*

*gmx grompp -f equilibration.mdp -c eq.gro -p system.top -o equilibration.tpr -n system.ndx*

*gmx mdrun -deffnm equilibration -v*

***Umbrella sampling***

***# copy em.mdp, em2.mdp, eq.mdp file.***

*Add restraint conformation and pull code*

***; Bond parameters***

*continuation = no ; continuing from NPT*

*constraints =* ***none***

*constraint-algorithm = lincs*

***; Pull code***

*pull = yes*

*pull\_ncoords = 1 ; only one reaction coordinate*

*pull\_ngroups = 2 ; two groups defining one reaction coordinate*

***pull\_group1\_name = OCO***

***pull\_group2\_name = ProDrug***

*pull\_coord1\_type = umbrella ; harmonic potential*

*pull\_coord1\_geometry = distance ; simple distance increase*

*pull\_coord1\_dim = N N Y*

*pull\_coord1\_groups = 1 2*

***pull\_coord1\_start = no ; define initial COM distance > 0***

*pull-coord1-init = 0.0*

***pull\_coord1\_rate = 0,0 ; 0.01 nm per ps = 10 nm per ns***

*pull\_coord1\_k = 1000 ; kJ mol^-1 nm^-2*

***pull-pbc-ref-prev-step-com = yes***

***pull-group1-pbcatom = 24797 ; 24583***

***#pull.mdp#***

*> gmx grompp -f pull.mdp -c ../eq.gro -p ../../Parameters/system.top -o pull.tpr -n ../system.ndx*

*> gmx mdrun -nt 1 -deffnm pull -v &> pullrun.log*

**#em.mdp#**

*> gmx grompp -f em.mdp -c pull.gro -p ../../system.top -o em.tpr -n ../system.ndx*

*> gmx mdrun -nt 1 -deffnm em -v &> emrun.log*

**#em2.mdp#**

*> gmx grompp -f em2.mdp -c em.gro -p ../../system.top -o em2.tpr -n ../system.ndx*

*> gmx mdrun -nt 1 -deffnm em2 -v &> em2run.log*

***#eq.mdp#***

*> gmx grompp -f eq.mdp -c em2.gro -p ../../system.top -o eq.tpr -n ../system.ndx*

*> gmx mdrun -nt 1 -deffnm eq -v -px eq\_x -pf eq\_f &> eqrun.log*

*gmx grompp -f pull.mdp -c ../equilibration.gro -p ../../Parameters/system.top -o pull.tpr -n ../system.ndx*

*gmx mdrun -nt 1 -deffnm pull -v &> pullrun.log*

*gmx grompp -f em.mdp -c pull.gro -p ../../Parameters/system.top -o em.tpr -n ../system.ndx*

*gmx mdrun -nt 1 -deffnm em -v &> emrun.log*

*gmx grompp -f em2.mdp -c em.gro -p ../../Parameters/system.top -o em2.tpr -n ../system.ndx*

*gmx mdrun -nt 1 -deffnm em2 -v &> em2run.log*

*gmx grompp -f eq.mdp -c em2.gro -p ../../Parameters/system.top -o eq.tpr -n ../system.ndx*

*gmx mdrun -nt 1 -deffnm eq -v -px eq\_x -pf eq\_f &> eqrun.log*

***Production***

*gmx grompp -f md.mdp -c ../../Initial/$window/eq.gro -p ../../Parameters/system.top -o md.tpr -n ../../Initial/system.ndx*

*gmx mdrun -nt 2 -deffnm md -v -px md\_x -pf md\_f -cpi md.cpt &> mdrun.log*

***Analysis***

***ls -d ../Production/\*/md.tpr > tpr-files.dat***

**ls -d ../Production/\*/md\_x.xvg > pullx-files.dat**

***gmx wham -ix pullx-files.dat -it tpr-files.dat -bsres -bins 200 -temp 300 -unit kJ -b 100 -nBootstrap 100 -zprof0 0.0 -min 0 -max 12***

***Solution/Ptx24/***

**3. ProDrug-subject MAE abstract**

Coarse-grained MD simulation of PtxPi1kda in solution (Water+150mM NaCl)

Conditions:

* 150mM NaCl + water
* 1mM ProDrug (1kda)
* 325140 beads with Martini V2.1\_dna FF
* 1microsecond expected

1. **Preparing step**

a. Here we used the optimised **itp** obtained from multiple coarsed-grained MD simulations of **1 PtxPi1kda in octonol**, here the selected parameters from simulation #6.

cp cg\_md.xtc cg\_md.tpr ~/ProDrug/Ptx\_PI\_SG1/PtxPi1kda\_redo/Alys\_1/CG

# GemPi1kda has been extract from the last frame ~ 5000 ns, choose #2 – non.

*gmx trjconv -f cg\_md.xtc -s cg\_md.tpr -o prodrug.gro*

b. Then using some unix routines, we replicate the chain until 24 polymers into the box. The box size has been fix at ~ 34.16nm for x,y,z dimensions for a concentration of [ProDrug] at 1mM.

# Place the solute in a simulation box

*gmx editconf -f prodrug.gro -bt cubic -d 1 -o prodrug\_inbox.gro*

# put additional 23 prodrug in the box

*gmx insert-molecules -f prodrug\_inbox.gro -ci prodrug.gro -o system0.gro -nmol 23 -box 32 32 32*

# define the size of box

*gmx editconf -f system0.gro -o system\_temp.gro -d 1.0 -bt cubic -box 34.16 34.16 34.16*

1. **Minimization steps**

# Minimization in vaccum following by minimization in solution

*gmx grompp -f minimization-vac.mdp -c system\_temp.gro -p system.top -o minimization-vac.tpr*

*gmx mdrun -deffnm minimization-vac -v*

*system.top*

*ptxpi1kda 24*

# Solvatation

*gmx solvate -cp minimization-vac.gro -cs water.gro -radius 0.21 -o system\_W.gro*

gmx grompp -f ions.mdp -c system\_W.gro -p system.top -o ions.tpr

*system.top*

*ptxpi1kda 24*

*W 324344*

# Add ions into the box

*gmx genion -s ions.tpr -p system.top -neutral -conc 0.15 -pname NA+ -nname CL- -o system\_WI.gro*

Then select the water group, here it’s the **#3.**

*system.top*

*ptxpi1kda 24*

*;W 324344*

W 317142

NA+ 3601

CL- 3601

# Adding anti-freeze water:

To take account of anti-freeze water into the simulation, it is necessary to edit the topology file.

Ex:

if the system contained 100000 water molecules, the final system\_WI.gro should contain 10% of anti-freeze water molecules (WF) and 90% of martini water.

# extract of topology file:

*ptxpi1kda 24*

*;W 317142*

W 285428

WF 31714

NA+ 3601

CL- 3601

NB: the gro file atomname changes is not necessary if the use a index file.

But you can edit the name of water molecule W to WF using bash command as vi.

:286415,318128s/W/WF/g

# minimization

*gmx grompp -f minimization.mdp -c system\_WI.gro -p system.top -o minimization.tpr -maxwarn 1*

*gmx mdrun -deffnm minimization -v*

1. **Equilibration and production step**

Set 50000 step for the equilibration following by 1µs of production

***nohup ./run\_ProDrug.sh > run\_ProDrug.out &* (4 days)**

*#define the set in equilibration.mdp#*

*vi equilibration.mdp, check the set as below;*

tc-grps = ProDrug Solvent

*gmx make\_ndx -f minimization.gro -o system.ndx*

*name 2(1kda) ProDrug*

*3(W) | 4(WF) | 5(ION)*

*name 6 Solvent*

# Equil

*gmx grompp -f equilibration.mdp -c minimization.gro -p system.top -o equilibration.tpr -n system.ndx*

gmx mdrun -deffnm equilibration -v

# Prod

*gmx grompp -f dynamic.mdp -c equilibration.gro -p system.top -o dynamic.tpr -n system.ndx*

*gmx mdrun -deffnm dynamic -v*

*Production* ***15 us:***

*12/06/2020 Ptx24\_traj1 in ping@byrd03:~/ProDrug/Ptx\_PI\_SG1/Ptx24\_traj1*

*will finish in 26/07/2020*

*12/06/2020 Ptx24\_traj2 in ping@byrd02:~/PtxPi1kda\_redo/Ptx24\_traj2*

*will finish in 26/07/2020*

***###run\_ProDrug.sh###***

***gmx mdrun -deffnm dynamic -cpi dynamic.cpt -s dynamic1.tpr -c dynamic\_15us.gro -noappend***

***nohup ./run\_ProDrug.sh > run\_ProDrug.out&***

# DATA Analysis

1. After simulation finished, download the **dynamic.xtc dynamic.tpr dynamic.gro** files from byrd03

**conbine all the trajectory files**

*gmx trjcat -f dynamic.xtc dynamic.part000\*.xtc -o prodrug\_traj1.xtc*

check the length of trajectory

*gmx check -f prodrug\_traj1.xtc*

**2. Analyze aggregate size distribution in the whole trajectory**

gmx make\_ndx -f ../minimization.gro -o traj1.ndx

***gmx clustsize -f prodrug\_traj1.xtc -s dynamic.tpr -mc maxclust.xvg -ac avcluste.xvg -nc nclust.xvg -cut 5 -n traj1.ndx -pbc***

##wacth the movie.

*vmd equilibration.gro, open prodrug\_traj1.xtc;*

c. Analyze the **RMSD** to proof whether the aggregation is stable.

#Convergence is also checked in terms of the structure, through the root mean square deviation (RMSD) against the starting structure and the average structure

*gmx rms -s dynamic.tpr -f traj1\_fin.gro -o rmsd\_1.xvg -pbc*

*xmgrace rmsd\_1.xvg*

**7. Analyze the plateau of aggregate**

**a. Choose the stable time, put it in the centre of box**

*gmx trjconv -f prodrug\_traj1.xtc -s dynamic.tpr -tu us -b 13.0 -e 15.0 -o traj1\_fin.gro -pbc cluster*

**b. creat the index.ndx file of the biggest aggregation**

*gmx make\_ndx -f traj1\_fin.gro -o traj1.ndx*

*name Ptx*

--a 1-21 | a 42-62 | a 83-103 | a 124-144 | a 165-185 | a 206-226 | a 247-267 | a 288-308 | a 329-349 | a 370-390 | a 411-431 | a 452-472 | a 493-513 | a 534-554 | a 575-595 | a 616-636 | a 657-677| a 698-718 | a 739-759 | a 780-800 | a 821-841 | a 862-882 | a 903-923 | a 944-964

serial 1 to 21 42 to 62 83 to 103 124 to 144 165 to 185 206 to 226 247 to 267 288 to 308 329 to 349 370 to 390 411 to 431 452 to 472 493 to 513 534 to 554 575 to 595 616 to 636 657 to 677 698 to 718 739 to 759 780 to 800 821 to 841 862 to 882 903 to 923 944 to 964

*name SG1*

a O1 C52 C56 P1 C116 C118

*name Links*

--a 22-35 | a 63-76 | a 104-117 | a 145-158 | a 186-199 | a 227-240 | a 268-281 | a 309-322 | a 350-363 | a 391-404 | a 432-445 | a 473-486 | a 514-527 | a 555-568 | a 596-609 | a 637-650 | a 678-691| a 719-732 | a 760-773 | a 801-814 | a 842-855 | a 883-896 | a 924-937 | a 965-978

serial 22 to 35 63 to 76 104 to 117 145 to 158 186 to 199 227 to 240 268 to 281 309 to 322 350 to 363 391 to 404 432 to 445 473 to 486 514 to 527 555 to 568 596 to 609 637 to 650 678 to 691 719 to 732 760 to 773 801 to 814 842 to 855 883 to 896 924 to 937 965 to 978

**traj1: 3 us - 9 us**

**maxcluster.ndx**

*name Ptx*

--a 1-21 | a 42-62 | a 124-144 | a 165-185 | a 206-226 | a 247-267 | a 329-349 | a 370-390 | a 411-431 | a 534-554 | a 575-595 | a 698-718 | a 739-759 | a 780-800 | a 903-923 | a 944-964

*name SG1*

a O1 C52 C56 P1 C116 C118

a 22-35 | a 63-76 | a 145-158 | a 186-199 | a 227-240 | a 268-281 | a 350-363 | a 391-404 | a 432-445 | a 555-568 | a 596-609 | a 719-732 | a 760-773 | a 801-814 | a 924-937 | a 965-978

cat maxcluster.ndx ../maxclust.ndx > max.ndx

**Radius of gyration of bigger clusters**

*gmx gyrate -f traj1\_fin.gro -s traj1\_fin.gro -o gyrate.xvg -n traj1.ndx*

**calculate the average of rdf distribution of Ptx. Link. End; (COM of Clusters)**

*1kda:*

*gmx convert-tpr -s dynamic.tpr -o cluster.tpr -n max.ndx*

*choose 2 : 1kda*

*Chains:*

*gmx rdf -selrpos mol\_com -seltype res\_com -ref 'group 2' -sel 'group 2' -n traj1.ndx -o rdf\_COM\_Chains.xvg -f traj1\_fin.gro -s cluster.tpr -bin 0.05 -rmax 5*

*Ptx:*

*gmx rdf -selrpos mol\_com -seltype res\_com -ref 'group 2' -sel 'group 3' -n traj1.ndx -o rdf\_COM\_Ptx.xvg -f traj1\_fin.gro -s cluster.tpr -bin 0.05 -rmax 5*

*SG1:*

*gmx rdf -selrpos mol\_com -seltype res\_com -ref 'group 2' -sel 'group 4' -n traj1.ndx -o rdf\_COM\_SG1.xvg -f traj1\_fin.gro -s cluster.tpr -bin 0.05 -rmax 5*

*Link:*

*gmx rdf -selrpos mol\_com -seltype res\_com -ref 'group 2' -sel 'group 5' -n traj1.ndx -o rdf\_COM\_link.xvg -f traj1\_fin.gro -s cluster.tpr -bin 0.05 -rmax 5*

*Diglycolate:*

*gmx rdf -selrpos mol\_com -seltype res\_com -ref 'group 2' -sel -n traj1.ndx -o rdf\_COM\_digly.xvg -f traj1\_fin.gro -s cluster.tpr -bin 0.05 -rmax 5*

*Bead\_25:*

*gmx rdf -selrpos mol\_com -seltype res\_com -ref 'group 2' -sel -n traj1.ndx -o rdf\_COM\_Bead\_25.xvg -f traj1\_fin.gro -s cluster.tpr -bin 0.05 -rmax 5*

**Calculate the Radius distribution between each PTX, LINK, SG1 (COM of themselves)**

***Chains:***

*gmx rdf -selrpos res\_com -seltype res\_com -ref 'group 2' -sel 'group 2' -n traj1.ndx -o rdf\_self\_Chains.xvg -f traj1\_fin.gro -s cluster.tpr -bin 0.05 -rmax 5*

*Ptx:*

*gmx rdf -selrpos res\_com -seltype res\_com -ref 'group 3' -sel 'group 3' -n traj1.ndx -o rdf\_self\_Ptx.xvg -f traj1\_fin.gro -s cluster.tpr -bin 0.05 -rmax 5*

*SG1:*

*gmx rdf -selrpos res\_com -seltype res\_com -ref 'group 4' -sel 'group 4' -n traj1.ndx -o rdf\_self\_SG1.xvg -f traj1\_fin.gro -s cluster.tpr -bin 0.05 -rmax 5*

*Link:*

*gmx rdf -selrpos res\_com -seltype res\_com -ref 'group 5' -sel 'group 5' -n traj1.ndx -o rdf\_self\_link.xvg -f traj1\_fin.gro -s cluster.tpr -bin 0.05 -rmax 5*

**e. Calculate the distance as function of time between COM of aggregation and each of Gems**

*gmx pairdist -f traj1\_fin.gro -s cluster.tpr -n max.ndx -o Gem\_pairdist.xvg -selrpos whole\_mol\_com -seltype mol\_com -pbc yes*

cat gem | *gmx pairdist -f trjconv\_stable.gro -s cluster.tpr -n trjconv\_stable.ndx -o Gem\_pairdist.xvg -selrpos whole\_mol\_com -seltype mol\_com -pbc yes*

*#calculate the rough distance by chimera#*

calculate the distance between center of mass of cluster1 and Gem1 in one time, to make sure if it a good result.

*chimera trjconv\_cluster0.gro*

**g. Calculate the Radius distribution of Gem. Link. End; (COM of Clusters)**

*vi Gem-----edit the reference group, like 2*

seq 30 53 >> Gem; put the calculate group in to Gem; like Gem group from 30 to 53

cat ../gem | gmx rdf -f traj1\_7\_15.xtc -s prodrug.tpr -n ../traj1.ndx -pbc yes -selrpos whole\_mol\_com -seltype res\_com -bin 0.05 -o Gem\_res.xvg

*~~gmx rdf -f trjconv\_stable.gro -s cluster.tpr -n trjconv\_stable.ndx -pbc yes -selrpos whole\_mol\_com -seltype mol\_com -bin 0.05 -o rdf\_End.xvg~~*

**calculate the average of rdf distribution**

*gmx rdf -selrpos mol\_com -seltype res\_com -ref 'group 2' -sel 'group 2' -n traj1.ndx -o rdf\_COM\_Chains\_r.xvg -f traj1\_fin.xtc -s traj1\_fin.tpr -bin 0.05 -rmax 5*

*gmx rdf -selrpos mol\_com -seltype res\_com -ref 'group 2' -sel 'group 78' -n traj1.ndx -o rdf\_COM\_Gem\_r.xvg -f traj1\_fin.xtc -s traj1\_fin.tpr -bin 0.05 -rmax 5*

*gmx rdf -selrpos mol\_com -seltype res\_com -ref 'group 2' -sel 'group 79' -n traj1.ndx -o rdf\_COM\_PI\_r.xvg -f traj1\_fin.xtc -s traj1\_fin.tpr -bin 0.05 -rmax 5*

*gmx rdf -selrpos mol\_com -seltype res\_com -ref 'group 2' -sel 'group 80' -n traj1.ndx -o rdf\_COM\_SG1\_r.xvg -f traj1\_fin.xtc -s traj1\_fin.tpr -bin 0.05 -rmax 5*

*# compare the rdf distribution between COM of aggregate and Gem, PI, SG1, Amide, Water#*

*gmx rdf -selrpos mol\_com -seltype res\_com -ref -sel -n Water.ndx -o rdf\_COM\_compare.xvg -f traj1\_15us.xtc -s ../dynamic.tpr -bin 0.05 -rmax 5 -b 7 -e 15 -tu us*

choose group 6 as ref, 78, 79, 80, 81, 82, 3

in traj1.ndx

group 6, all the beads of aggregate

group 79, all the beads of Gem

group 80, al the beads of PI

group 81, all the beads of SG1

gourp 82, all the Amide

group 3, all the Water

**Calculate the Radius distribution between COM of GEM, LINK, SG1**

gmx rdf -selrpos res\_com -seltype res\_com -ref 'group 6' -sel 'group 6' -n max.ndx -o rdf\_self\_Chains.xvg -f traj1\_fin.gro -s cluster.tpr -bin 0.05 -rmax 5

gmx rdf -selrpos res\_com -seltype res\_com -ref 'group 6' -sel 'group 3' -n max.ndx -o rdf\_self\_Gem.xvg -f traj1\_fin.gro -s cluster.tpr -bin 0.05 -rmax 5

gmx rdf -selrpos res\_com -seltype res\_com -ref 'group 6' -sel 'group 4' -n max.ndx -o rdf\_self\_SG1.xvg -f traj1\_fin.gro -s cluster.tpr -bin 0.05 -rmax 5

gmx rdf -selrpos res\_com -seltype res\_com -ref 'group 6' -sel 'group 5' -n max.ndx -o rdf\_self\_link.xvg -f traj1\_fin.gro -s cluster.tpr -bin 0.05 -rmax 5

**I. calculate the minium distance between Gem/SG1 and Gem/SG1, chains and chains**

gmx mindist -f traj1\_fin.xtc -s traj1\_fin.tpr -n traj1.ndx -on dist\_mindist.xvg

**or**

gmx pairdist -f traj1\_fin.xtc -s traj1\_fin.tpr -n traj1.ndx -o dist5.xvg -selrpos atom -seltype atom

#### Gem\_Gem.sh #####

#!/bin/bash

rm -rf Mini\_GEMDISTRIBUTIONS

mkdir Mini\_GEMDISTRIBUTIONS

rm -rf Mini\_GEMDAT

mkdir Mini\_GEMDAT

NGEM=29

IGEM=6

while [ $IGEM -lt $NGEM ]

do

seq $IGEM 29 >> Mini\_GEMDAT/gem\_$IGEM.dat

cat Mini\_GEMDAT/gem\_$IGEM.dat | gmx pairdist -f traj2\_7\_15.xtc -s ../traj2.tpr -n ../../Anne\_7to10/traj2.ndx -o Mini\_GEMDISTRIBUTIONS/GEM\_$IGEM.xvg -selrpos atom -seltype atom

sed -i '/^\s\*[@ s]/ d' Mini\_GEMDISTRIBUTIONS/GEM\_$IGEM.xvg

rm -rf \#\*

let IGEM=$IGEM+1

done

rm -rf Mini\_GEMDAT

exit

###Gem\_analys.sh###

#!/bin/bash

rm \*dat

NGEM=29

for ((IGEM=6;IGEM<NGEM;IGEM++));

do

cp GEM\_$IGEM.xvg GEM\_$IGEM.dat

awk '{print $1}' GEM\_6.dat > GEM\_6\_dist.dat

awk '{f=""; for (i=2; i<=NF; i++) f=f" "$i;} f!=last{print f;} {last=f;}' GEM\_$IGEM.dat > GEM\_d$IGEM.dat

rm GEM\_d27.dat

awk '{print $2,$3}' GEM\_27.dat > GEM\_d27.dat

awk '{print $2}' GEM\_28.dat > GEM\_d28.dat

wc -l GEM\_d\*.dat

paste GEM\_6\_dist.dat GEM\_d\*.dat > GEM\_ALL2.xvg

awk '{for (i=2;i<=NF; i++) print $1, $i}' GEM\_ALL2.xvg > GEM\_ALL\_dist2.xvg

done

rm \*dat

exit

###Gem\_SG1.sh###

#!/bin/bash

rm -rf Mini\_GEM\_SG1DISTRIBUTIONS

mkdir Mini\_GEM\_SG1DISTRIBUTIONS

rm -rf Mini\_GEM\_SG1DAT

mkdir Mini\_GEM\_SG1DATdynamic.tpr

NGEM=29

for ((IGEM=6;IGEM<=NGEM;IGEM++));

do

seq $IGEM $IGEM >> Mini\_GEM\_SG1DAT/GEM\_SG1\_$IGEM.dat

seq 54 77 >> Mini\_GEM\_SG1DAT/GEM\_SG1\_$IGEM.dat

cat Mini\_GEM\_SG1DAT/GEM\_SG1\_$IGEM.dat | gmx pairdist -f traj2\_7\_15.xtc -s ../traj2.tpr -n ../../Anne\_7to10/traj2.ndx -o Mini\_GEM\_SG1DISTRIBUTIONS/GEM\_SG1\_$IGEM.xvg -selrpos atom -seltype atom

sed -i '/^\s\*[@ s]/ d' Mini\_GEM\_SG1DISTRIBUTIONS/GEM\_SG1\_$IGEM.xvg

rm -rf \#\*

done

rm -rf Mini\_GEM\_SG1DAT

exit

###Gem\_SG1\_analysis.sh###

#!/bin/bash

NGEM=29

for ((IGEM=6;IGEM<=NGEM;IGEM++));

do

cp GEM\_SG1\_$IGEM.xvg GEM\_SG1\_$IGEM.dat

awk '{print $1}' GEM\_SG1\_6.dat > GEM\_SG1\_6\_dist.dat

awk '{print ($2,$3,$4,$5,$6,$7,$8,$9,$10,$11,$12,$13,$14,$15,$16,$17,$18,$19,$20,$21,$22,$23,$24,$25)}' GEM\_SG1\_$IGEM.dat > GEM\_SG1\_dd$IGEM.dat

paste GEM\_SG1\_6\_dist.dat GEM\_SG1\_dd\* > GEM\_SG1\_ALL2.xvg

wc -l GEM\_SG1\_dd\*

awk '{for (i=2;i<=NF; i++) print $1, $i}' GEM\_SG1\_ALL2.xvg > GEM\_SG1\_ALL\_dist2.xvg

done

rm \*.dat

exit

###SG1\_SG1.sh###

#!/bin/bash

rm -rf Mini\_SG1DISTRIBUTIONS

mkdir Mini\_SG1DISTRIBUTIONS

rm -rf Mini\_SG1DAT

mkdir Mini\_SG1DAT

NSG1=77

ISG1=54

while [ $ISG1 -lt $NSG1 ]

do

seq $ISG1 77 >> Mini\_SG1DAT/SG1\_$ISG1.dat

cat Mini\_SG1DAT/SG1\_$ISG1.dat | gmx pairdist -f traj2\_7\_15.xtc -s ../traj2.tpr -n ../../Anne\_7to10/traj2.ndx -o Mini\_SG1DISTRIBUTIONS/SG1\_$ISG1.xvg -selrpos atom -seltype atom

sed -i '/^\s\*[@ s]/ d' Mini\_SG1DISTRIBUTIONS/SG1\_$ISG1.xvg

rm -rf \#\*

let ISG1=$ISG1+1

done

rm -rf Mini\_SG1DAT

exit

###SG1\_SG1\_analysis.sh###

#!/bin/bash

rm \*dat

NGEM=77

for ((IGEM=54;IGEM<NGEM;IGEM++));

do

cp SG1\_$IGEM.xvg SG1\_$IGEM.dat

awk '{print $1}' SG1\_54.dat > SG1\_54\_dist.dat

awk '{f=""; for (i=2; i<=NF; i++) f=f" "$i;} f!=last{print f;} {last=f;}' SG1\_$IGEM.dat > SG1\_d$IGEM.dat

awk '{print $2}' SG1\_76.dat > SG1\_d76.dat

wc -l SG1\_d\*.dat

paste SG1\_54\_dist.dat SG1\_d\*.dat > SG1\_ALL2.xvg

awk '{for (i=2;i<=NF; i++) print $1, $i}' SG1\_ALL2.xvg > SG1\_ALL\_dist2.xvg

done

rm \*dat

exit

###Chains.sh###

#!/bin/bash

rm -rf Mini\_CHAINSDISTRIBUTIONS

mkdir Mini\_CHAINSDISTRIBUTIONS

rm -rf Mini\_CHAINSDAT

mkdir Mini\_CHAINSDAT

NCHAINS=26

ICHAINS=3

while [ $ICHAINS -lt $NCHAINS ]

do

seq $ICHAINS 26 >> Mini\_CHAINSDAT/CHAINS\_$ICHAINS.dat

cat Mini\_CHAINSDAT/CHAINS\_$ICHAINS.dat | gmx pairdist -f traj2\_7\_15.xtc -s ../traj2.tpr -n ../../Anne\_7to10/Chains.ndx -o Mini\_CHAINSDISTRIBUTIONS/CHAINS\_$ICHAINS.xvg -selrpos atom -seltype atom

sed -i '/^\s\*[@ s]/ d' Mini\_CHAINSDISTRIBUTIONS/CHAINS\_$ICHAINS.xvg

rm -rf \#\*

let ICHAINS=$ICHAINS+1

done

rm -rf Mini\_CHAINSDAT

exit

###Chains\_analysis.sh###

#!/bin/bash

rm \*dat

NCHAINS=25

for ((ICHAINS=3;ICHAINS<=NCHAINS;ICHAINS++));

do

cp CHAINS\_$ICHAINS.xvg CHAINS\_$ICHAINS.dat

awk '{print $1}' CHAINS\_3.dat > CHAINS\_3\_dist.dat

awk '{f=""; for (i=2; i<=NF; i++) f=f" "$i;} f!=last{print f;} {last=f;}' CHAINS\_$ICHAINS.dat > CHAINS\_d$ICHAINS.dat

awk '{print $2}' CHAINS\_25.dat > CHAINS\_d25.dat

wc -l CHAINS\_d\*.dat

paste CHAINS\_3\_dist.dat CHAINS\_d\*.dat > CHAINS\_ALL2.xvg

awk '{for (i=2;i<=NF; i++) print $1, $i}' CHAINS\_ALL2.xvg > CHAINS\_ALL\_dist2.xvg

done

rm \*dat

exit

**J. Solvent accessible surface area (SASA)**

In order to study the drug release of polymer prodrugs, we compare the Solvent accessible surface of active points (Amide) in aggregate (contains 24 chains) and 1 chain (in octanol and in same calculate condition (water, WF and ions, same concentration of GemPi1kda)

For 24 chains aggregate

gmx sasa -f traj1\_fin.gro -s cluster.tpr -n traj1.ndx -o SAS\_all.xvg -or RES\_SAS\_all.xvg -pbc -surface -output -ndots 25

PtxPi1kda as surface

Ptx, diglycolate, bead\_25, hydrophilic group as output

SAS\_all.xvg is the All SAS distribute as function of times

RES\_SAS\_all.xvg is the average SAS distribution as function of chains

For 1 chain calculation

1. 1chain in octanol (jarre, 6h)

:~/GemPi1kDa/byrd03/octanol/Take6\_ok/test\_sasa

~/ProDrug/GemPi1kDa/byrd03/solution/Gem24/trajectory/1chain\_octanol

#increase the calculate time until 1 us#

*gmx convert-tpr -s cg\_md.tpr -o cg\_md\_1.tpr -until 1000000*

*gmx mdrun -s cg\_md\_1.tpr -v -deffnm cg\_md*

Analysis

#SASA#

gmx sasa -f cg\_md\_1.xtc -s cg\_md\_1.tpr -n cg\_md.ndx -o 1chain\_oct\_sasa.xvg -or 1chain\_oct\_res\_sas.xvg -pbc -surface -output -ndots 25

#Average#

*awk '{sum+=sprintf("%f",$5)}END{printf "%.6f\n%.6f\n",sum,sum/NR}' 1chain\_W\_sasa.xvg*

2. 1chain in water ( byrd02, 3h)

~/ProDrug/GemPi1kDa/byrd03/solution/Gem24/trajectory/1chain\_water

conditions:

* 150mM NaCl + water
* 1mM ProDrug (1kda)
* 1microsecond expected
* same cg\_md.itp file in octanol

Copy cg\_md.xtc, cg\_md.tpr, 1kda\_mapped.itp from ~/ProDrug/Octanol/Take6\_ok

*#creat prodrug gro file#*

*gmx trjconv -f cg\_md.xtc -s cg\_md.tpr -o prodrug.gro*

# Place the solute in a simulation box

*gmx editconf -f prodrug.gro -bt cubic -d 1 -o prodrug\_inbox.gro*

# define the size of box

*gmx editconf -f prodrug\_inbox.gro -o system\_temp.gro -d 1.0 -bt cubic -box 11.84 11.84 11.84*

# Minimization in vaccum following by minimization in solution

*gmx grompp -f minimization-vac.mdp -c system\_temp.gro -p system.top -o minimization-vac.tpr*

*gmx mdrun -deffnm minimization-vac -v*

# Solvatation

*gmx solvate -cp minimization-vac.gro -cs water.gro -radius 0.21 -o system\_W.gro*

*gmx grompp -f ions.mdp -c system\_W.gro -p system.top -o ions.tpr*

# Add ions into the box

*gmx genion -s ions.tpr -p system.top -neutral -conc 0.15 -pname NA+ -nname CL- -o system\_WI.gro*

Then select the water group, here it’s the #3.

# Add anti-freeze water into the box

Gempi1kda 1

;W 13115

W 11533

WF 1282

NA+ 150

CL - 150

change the last 1282 W to WF in *system\_WI.gro*

# minimization

*gmx grompp -f minimization.mdp -c system\_WI.gro -p system\_2.top -o minimization.tpr -maxwarn 1*

*gmx mdrun -deffnm minimization -v*

*#define the set in equilibration.mdp#*

*vi equilibration.mdp, check the set as below;*

tc-grps = ProDrug Solvent

*gmx make\_ndx -f minimization.gro -o system.ndx*

*name 2(1kda) ProDrug*

*3(W) | 4(WF) | 5(ION)*

*name 6 Solvent*

# Equil

*gmx grompp -f equilibration.mdp -c minimization.gro -p system.top -o equilibration.tpr -n system.ndx*

gmx mdrun -deffnm equilibration -v

# Prod

*gmx grompp -f dynamic.mdp -c equilibration.gro -p system.top -o dynamic.tpr -n system.ndx*

*gmx mdrun -deffnm dynamic -v*

*Analysis*

#SASA#

*gmx sasa -f dynamic.xtc -s dynamic.tpr -n 1chain.ndx -o 1chain\_W\_sasa.xvg -or 1chain\_W\_res\_sas.xvg -pbc -surface -output -ndots 25*

#Average#

*awk '{sum+=sprintf("%f",$5)}END{printf "%.6f\n%.6f\n",sum,sum/NR}' 1chain\_W\_sasa.xvg*

***SASA of Ptx\_digly (bead1-25)***

[*ping@jarre*](mailto:ping@jarre)*:~/ProDrug/Ptx\_PI\_SG1/PtxPi1kda\_redo/Solution/Ptx1/Ptx1\_water\_redo/Ptx\_digly*

*ln -s ../../Ptx1\_water\_redo/dynamic.gro dynamic.xtc dynamic.tpr*

*gmx make\_ndx -f dynamic.gro -o Ptx\_digly.ndx*

*a 1-25 Ptx\_digly*

*3 | 4 | 5 Solvent*

*6 | 7 new\_system*

*gmx trjconv -f dynamic.xtc -s dynamic.tpr -o Ptx\_digly.gro -pbc whole -n Ptx\_digly.ndx*

*cp ../../../../Ptx1\_water\_redo/\*.mdp system.top mart\**

# minimization

*gmx grompp -f minimization.mdp -c Ptx\_digly.gro -p system.top -o minimization.tpr -maxwarn 1*

*gmx mdrun -deffnm minimization -v*

*#define the set in equilibration.mdp#*

*vi equilibration.mdp, check the set as below;*

tc-grps = ProDrug Solvent

*gmx make\_ndx -f minimization.gro -o system.ndx*

*name 2 ProDrug*

*3(W) | 4(WF) | 5(ION)*

*name 6 Solvent*

# Equil

*gmx grompp -f equilibration.mdp -c minimization.gro -p system.top -o equilibration.tpr -n system.ndx*

gmx mdrun -deffnm equilibration -v

# Prod

*gmx grompp -f dynamic.mdp -c equilibration.gro -p system.top -o dynamic\_new.tpr -n system.ndx*

*gmx mdrun -deffnm dynamic\_new -v*

*Analysis*

#SASA#

*gmx sasa -f dynamic\_new.xtc -s dynamic\_new.tpr -n Ptx\_digly.ndx -or 1chain\_W\_res\_sas.xvg -pbc -surface -output -ndots 25*

***Biphase\_PtxPi1kda***

*[ Bead1\_7 ]*

*1 2 3 4 5 6 7 42 43 44 45 46 47 48 83 84 85 86 87 88 89 124 125 126 127 128 129 130 165 166 167 168 169 170 171 206 207 208 209 210 211 212 247 248 249 250 251 252 253 288 289 290 291 292 293 294 329 330 331 332 333 334 335 370 371 372 373 374 375 376 411 412 413 414 415 416 417 452 453 454 455 456 457 458 493 494 495 496 497 498 499 534 535 536 537 538 539 540 575 576 577 578 579 580 581 616 617 618 619 620 621 622 657 658 659 660 661 662 663 698 699 700 701 702 703 704 739 740 741 742 743 744 745 780 781 782 783 784 785 786 821 822 823 824 825 826 827 862 863 864 865 866 867 868 903 904 905 906 907 908 909 944 945 946 947 948 949 950*

*[ Bead\_8 ]*

*8 49 90 131 172 213 254 295 336 377 418 459 500 541 582*

*623 664 705 746 787 828 869 910 951*

*[ Bead9\_11 ]*

*9 10 11 50 51 52 91 92 93 132 133 134 173 174 175*

*214 215 216 255 256 257 296 297 298 337 338 339 378 379 380*

*419 420 421 460 461 462 501 502 503 542 543 544 583 584 585*

*624 625 626 665 666 667 706 707 708 747 748 749 788 789 790*

*829 830 831 870 871 872 911 912 913 952 953 954*

*[ Bead12\_15 ]*

*12 13 14 15 53 54 55 56 94 95 96 97 135 136 137*

*138 176 177 178 179 217 218 219 220 258 259 260 261 299 300*

*301 302 340 341 342 343 381 382 383 384 422 423 424 425 463*

*464 465 466 504 505 506 507 545 546 547 548 586 587 588 589*

*627 628 629 630 668 669 670 671 709 710 711 712 750 751 752*

*753 791 792 793 794 832 833 834 835 873 874 875 876 914 915*

*916 917 955 956 957 958*

*[ Bead16\_18 ]*

*16 17 18 57 58 59 98 99 100 139 140 141 180 181 182*

*221 222 223 262 263 264 303 304 305 344 345 346 385 386 387*

*426 427 428 467 468 469 508 509 510 549 550 551 590 591 592*

*631 632 633 672 673 674 713 714 715 754 755 756 795 796 797*

*836 837 838 877 878 879 918 919 920 959 960 961*

*[ Bead19\_21 ]*

*19 20 21 60 61 62 101 102 103 142 143 144 183 184 185 224 225 226 265 266 267 306 307 308 347 348 349 388 389 390 429 430 431 470 471 472 511 512 513 552 553 554 593 594 595 634 635 636 675 676 677 716 717 718 757 758 759 798 799 800 839 840 841 880 881 882 921 922 923 962 963 964*

**k. Analyze the average rdf distribution (Tap said is not a good method, but script can use for another things)**

#cp part4\_rdf\_Gem\_mol.xvg part4\_rdf\_Gem\_mol.dat

#vi part4\_rdf\_Gem\_mol.dat

Read the values of columns from 2nd to 17th in rdf\_Gem.dat

awk '{print ($2+$3+$4+$5+$6+$7+$8+$9+$10+$11+$12+$13+$14+$15+$16+$17+$18+$19+$20+$21+$22+$23+$24+$25)}' rdf\_Gem.dat

Calculate the average value of columns from 2nd to 17th , then output into rdf\_gem\_average.dat

awk '{print ($2+$3+$4+$5+$6+$7+$8+$9+$10+$11+$12+$13+$14+$15+$16+$17+$18+$19+$20+$21+$22+$23+$24+$25)/24}' rdf\_Gem.dat > rdf\_Gem\_average.dat

output the value of 1st columns to rdf\_gem\_dist.dat

awk '{print $1}' part4\_rdf\_Gem\_mol.dat > rdf\_gem\_dist.dat

check the columns of two files

wc -l rdf\_End\_average.dat rdf\_End\_dist.dat

combine two .dat files into one .dat file

paste rdf\_End\_dist.dat rdf\_End\_average.dat > rdf\_End\_dist\_average.dat

open the file, check the results

xmgrace rdf\_gem\_dist\_average.dat