HTMD Startup

In [ ]:
```
from htmd import *
from htmd.protocols.equilibration_v1 import Equilibration
from htmd.protocols.production_v1 import Production
from htmd.molecule.util import uniformRandomRotation
mdx=AcemdLocal()
```

System Preparation

In [ ]:
```
prot=Molecule('complex.pdb')  #needs CYS HG->HG1
prot.set('name','HG1',sel='resname CYS and name HG')
prot.set('segid', 'P', sel='protein')
prot.set('segid', 'W', sel='water')
prot.set('segid', 'CA', sel='resname CA')
prot.set('segid','L', sel='resname MOL')
memb=Molecule('mem80x80_POPC.pdb')
memb.filter('resname POPC')
prot.center()
memb.center()
mol=embed(prot,memb)
```

System Solvation

In [ ]:
```
smol = solvate(mol, minmax=[[-40, -40, -45], [40, 40, 45]])
smol.write('mem-prot_sys.pdb')
```

System Build

In [ ]:
```
ds = [ DisulfideBridge('P',42,'P',296), DisulfideBridge('P',124,'P',202) ]
topos  = ['ligand.rtf','top/top_all36_prot.rtf','top/top_all36_cgenff.rtf','top/top_all36_lipid.rtf','top/top_water_ions.rtf']
params = ['ligand.prm','par/par_all36_prot.prm','par/par_all36_cgenff.prm','par/par_all36_lipid.prm','par/par_water_ions.prm']

plm = charmm.build(smol, topo=topos, param=params,outdir='build',ionize=True,caps=['first ACE','last CT3'], saltconc=0.154,saltanion='
CLA',saltcation='SOD',disulfide=ds)
```

Equilibration

In [ ]:
```
md = Equilibration()
md.numsteps= 20000000
md.temperature = 310
md.useconstantratio = True
md.constraints = {'water': 0.0,'resname MOL': 0.8, 'protein and not name CA': 0.4,'protein and noh and not name CA': 0.1 ,'protein and
name CA': 0.85}
md.acemd.timestep='2'
md.acemd.minimize='5000'
md.k = 0
md.write('build', 'equil')
mdx.submit('equil')
```

Production (Setup for 3 Replicas)

In [ ]:
```
md = Production()
mdx.wait()   # wait for equilibration to complete
md.acemd.run='200ns'
md.temperature = 310
md.acemd.timestep='2'
md.acemd.xtcfreq='10000'
md.acemd.bincoordinates = 'output.coor'   #start
md.acemd.extendedsystem  = 'output.xsc'   #start
md.acemd.binvelocities=None
md.acemd.binindex=None
md.write('equil','run1')

### Run 2 ###
md = Production()
mdx.wait()   # wait for equilibration to complete
md.acemd.run='200ns'
md.temperature = 310
md.acemd.timestep='2'
md.acemd.xtcfreq='10000'
md.acemd.bincoordinates = 'output.coor'   #start
md.acemd.extendedsystem  = 'output.xsc'   #start
md.acemd.binvelocities=None
md.acemd.binindex=None
md.write('equil','run2')

### Run 3 ###
md = Production()
mdx.wait()   # wait for equilibration to complete
md.acemd.wrap='on'
md.acemd.run='200ns'
md.temperature = 310
md.acemd.timestep='2'
md.acemd.xtcfreq='10000'
md.acemd.bincoordinates = 'output.coor'   #start
md.acemd.extendedsystem  = 'output.xsc'   #start
md.acemd.binvelocities=None
md.acemd.binindex=None
md.write('equil','run3')
```

Production (Run 3 Replicas in Series)

In [ ]:
```
mdx.submit('run1')
mdx.wait()
mdx.submit('run2')
mdx.wait()
mdx.submit('run3')
```