

## 9 Přílohy

### 9.1 Metoda MD simulací prakticky

Následující kapitola bude obsahovat mé přesné postupy, v některých případech i triky z praxe, řešení možných problémů, ukázky mnou používaných souborů a návody na zvládnutí všech mnou prováděných úkonů pro přípravu a spuštění MD simulace v GROMACSu. V MD simulacích a počítačových vědách vůbec existuje vždy vícero možností, jak zadaný problém vyřešit, a záleží proto na kreativitě a především pak lenosti uživatele. Zde uváděné a mnou využívané postupy se z hlediska funkce pro mé potřeby osvědčily, i když je možné, že existují mnohem elegantnější řešení. Dalším úskalím je, že i když mně zde popsané postupy fungovaly pro krátké peptidy, nemusí platit pro jiné biomolekuly. Každá molekula, se kterou se v MD simulaci pracuje, obvykle disponuje jistou „osobitostí“ a není proto výjimečné, že co funguje pro jednu molekulu, nemusí fungovat pro druhou. Následující text je psán tak, aby i čtenář naprosto neznalý světa MD simulací, byl schopen zopakovat veškeré mé postupy. Jenom tak může být tato práce užitečná budoucím studentům biologie, kteří se s MD simulacemi v praxi ještě nesetkali, rádi by je vyzkoušeli a zatím nevědí, jak jako samouci začít. Vystudovaný teoretik fyzikálních a chemických věd necht' promine a nebere následující text jako ironický a groteskní. Inspirace při psaní následujících návodů vycházela z populárních anglických knih z edice „*for Dummies*“.

### 9.2 Základní soubory nutné pro start MD simulace

Pro spuštění jakékoliv MD simulace jsou potřeba tři základní typy souborů:

- Atomové souřadnice ve formátu *.gro*, obvykle soubor *conf.gro*
- Topologie (obvykle soubor *topol.itp*) a silová pole pro atomy ve formátu *.itp*
- Parametry a podmínky pro MD simulaci v souboru s příponou *.mdp*, obvykle *grompp.mdp*

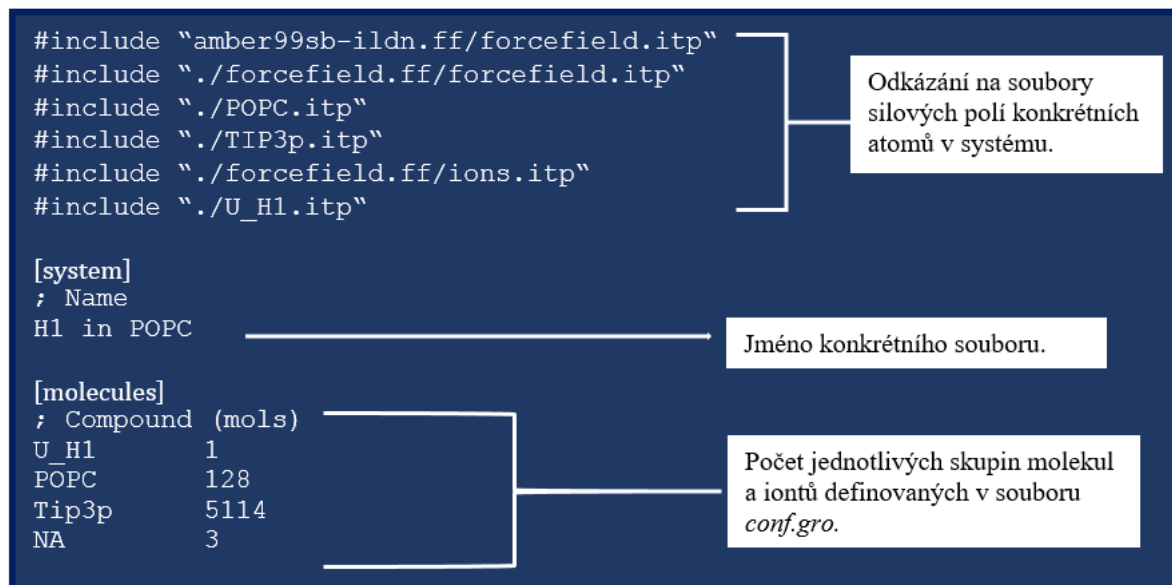
Obsah těchto základních souborů je v této práci demonstrován na mnou používaných souborech pro MD simulaci H1 v POPC membráně. V následujících odstavcích také krátce popsány a některé jejich detaily stroze vysvětleny.

Ukázkový soubor *conf.gro*, obsahující všechny definované atomy a jejich souřadnice v systému, je zobrazen na Obr. 88 níže. V hlavičce souboru je zobrazen obvykle jeho název. Pokud pochází vstupní *conf.gro* soubor již z nějaké trajektorie vygenerované MD simulací, bývá to zde také obvykle zapsáno. Např. nástroj *trjconv* je schopen získat souřadnice atomů v systému v konkrétním čase MD simulace, v tomto případě to bylo v 80 ps.

Generated by trjconv : H1 in POPC t= 80.0000						
32906						Celkový počet atomů.
7ALA	N	1	2.017	4.144	5.637	Číselné označení a jméno konkrétní molekuly.
7ALA	H1	2	2.061	4.084	5.705	
7ALA	H2	3	1.925	4.166	5.671	
7ALA	H3	4	2.065	4.233	5.635	
7ALA	CA	5	2.006	4.071	5.510	
...						Název konkrétního atomu.
35POPC	N	410	4.774	0.524	5.160	Pořadí konkrétního atomu v souboru.
35POPC	C12	411	4.679	0.546	5.270	
...						X Y Z souřadnice daného atomu.
5276SOL	OH232901		2.181	1.728	0.503	
5276SOL	H132902		2.156	1.652	0.556	
5276SOL	H232903		2.099	1.761	0.468	
5277NA	NA32904		6.027	0.080	0.740	
5278NA	NA32905		4.947	0.576	5.670	
5279NA	NA32906		4.046	2.557	1.213	Rozměry pbc boxu.
			6.45843	6.75200	7.26655	

**Obr. 88.:** Části souboru *conf.gro*. V hlavičce souboru je vždy uveden celkový počet atomů odpovídající poslednímu číselnému označení atomu druhého sloupce. První sloupec označuje skupinu molekul, např. 7ALA představuje první v pořadí AMK helixu H1, ... naznačují pokračování popisu molekuly. 35POPC jsou první molekuly POPC membrány po H1, 5276SOL označuje poslední molekulu rozpouštědla v systému. 5277-5279NA představují přidané protiionty k H1. Druhý sloupec obsahuje identifikaci konkrétních atomů, např. N označuje atom dusíku, 3 atomy vodíku za ním (H1-H3) jsou navázané na N-terminálu H1, CA označuje  $\alpha$ -uhlík helixu H1, N a C12 atomy molekuly 35POPC jsou také vypsány. Atomy OH2, H1 a H2 náleží molekulám vody. Třetí sloupek *conf.gro* pak definuje pořadí atomů a za ním následují už jednotlivé souřadnice odpovídající pozicím konkrétních atomů v kartézském souřadnicovém systému. Na konci souboru jsou uvedeny rozměry pbc boxu v nm.

Soubor *topol.top* (Obr. 88) obsahuje topologii celého systému, zde H1 v POPC, použitého pro MD simulaci v Gromacsu. Zahrnuje definice pro silová pole (FF) nutná pro MD výpočet (FF rozpouštědel, a iontů z databáze Gromacsu, vazebné i nevazebné interakce, FF vytvořená uživatelem, aj.) a dále počet jednotlivých molekul, ev. atomů a iontů v definovaném systému.



**Obr. 88:** Ukázkový soubor *topol.top*. Do jeho hlavičky jsou vloženy skrze instrukci *include* FF jednotlivých skupin molekul tvořících systém peptidu H1 v POPC membráně. „; Name“ označuje komentář uživatele vymezený středníkem. Tento komentář je neviditelný pro program. „H1 in POPC“ je název daného systému, U\_H1 představuje molekulu helixu H1 kolicinu U, 128 molekul POPC zastupuje 128 lipidů palmitoyl-oleoyl-glycerofosfocholinu tvořících membránu. Tip3p<sup>4</sup> jsou molekuly rozpouštědla.

Část souboru pro FF helixu H1 (U\_H1.itp), který byl vygenerovaný příkazem *pdb2gmx* a mnou upraven do vhodného použití pro MD simulaci H1 v POPC, je na Obr. 89. Do komentovaného úvodu je vhodné psát původ FF. Soubor pro FF má v části [moleculetype] jméno dané molekuly, které musí být stejné jako v *topol.top*. Oddělení [atoms] definuje jednotlivé atomy, obsahuje jejich číselné označení (nr), přiřazení (type, resnr, residue, atom, cgnr), náboje (*e*) a atomovou hmotnost (*a.u.*). Dále jsou v souboru ve sloupkové podobě délky vazeb, vazebné úhly, torze a distorze a na konci podmínkový cyklus pro soubor *posre.itp*. V [bonds] funct 1 představují vazby popsané harmonickým oscilátorem, v [pairs] jsou atomy navzájem od sebe oddělné třemi vazbami. Nakonec jsou definovány úhly [angles] a torze [dihedrals].

<sup>4</sup>Běžnější je v *topol.top* označení molekul rozpouštědla jako SOL; já používala označení tip3p z praktických důvodů, a při spuštění MD simulace nástrojem *grompp* jsem atomy nechala automaticky přejmenovat, samotné *grompp* bylo spuštěno současně s možností *-maxwarn 4*, 4 chyby totiž odpovídaly přepisu těchto atomů molekul tip3p na SOL a iontu NA či CL na Na či Cl.

```

; File 'topol.top' was generated by user Kamila Riedlova on host Planck at date
4.3.2017
; It was generated using program: pdb2gmx - version 4.6.5
; Command line was: pdb2gmx -f H1-vertical.gro
; Include forcefield parameters #include "amber99sb-ildn.ff/forcefield.itp"

[moleculetype]
; Name          nrexcl
U_H1           3

[atoms]
; nr  type  resnr  residue  atom  cgnr  charge  mass  typeB  chargeB  massB
; residue 7ALA rtp -NALA q +1.0
  1   N3    7     ALA     N     1     0.1414  14.01 ;qtot 0.1414
  2   H     7     ALA    H1    2     0.1997  1.008 ;qtot 0.3411
  3   H     7     ALA    H2    3     0.1997  1.008 ;qtot 0.5408
  4   H     7     ALA    H3    4     0.1997  1.008 ;qtot 0.7405
  5   CT    7     ALA    CA    5     0.0962  12.01 ;qtot 0.8367
  6   HP    7     ALA    HA    6     0.0889  1.008 ;qtot 0.9256
  7   CT    7     ALA    CB    7     -0.0597 12.01 ;qtot 0.8659
  8   HC    7     ALA    HB1   8     0.03    1.008 ;qtot 0.8959
  9   HC    7     ALA    HB2   9     0.03    1.008 ;qtot 0.9259
 10  HC    7     ALA    HB3  10     0.03    1.008 ;qtot 0.9559
 11  C     7     ALA    C     11    0.6163  12.01 ;qtot 1.572
 12  O     7     ALA    O     12    -0.5722  16     ;qtot 1
; residue 8ASN rtp  ASN q 0.0
 13  N     8     ASN    N     13    -0.4157  14.01 ;qtot 0.5843
 14  H     8     ASN    H     14     0.2719  1.008 ;qtot 0.8562
...

[bonds]
; ai  aj  funct  c0  c1  c2  c3
  1   2   1      1
  1   3   1      1
  1   5   1      1
...

[pairs]
; ai  aj  funct  c0  c1  c2  c3
  1   8   1      1
  1   9   1      1
  1  10   1      1
...

[angles]
; ai  aj  ak  funct  c0  c1  c2  c3
  2   1  3   1      1
  2   1  4   1      1
  2   1  5   1      1
...

[dihedrals]
; ai  aj  ak  al  funct  c0  c1  c2  c3  c4  c5
  2   1  5   6   9      1
  2   1  5   7   9      1
  2   1  5  11   9      1
...

; Include Position restraint file
#ifdef POSRES
#include "posre.itp"
#endif

```

**Obr. 89:** část souboru FF pro H1 kolicinu U vygenerovaný pdb2gmx a mnou upravený. Kvůli přehlednosti zde už nejsou uvedeny popisky.

Soubor *grompp.mdp* zobrazený níže na Obr. 90 zahrnuje parametry a podmínky nutné pro spuštění a běh MD simulace.

```

; PARAMETRY PRO BĚH MD SIMULACE
integrator          = md
tinit              = 0
dt                 = 0.002
nsteps             = 250000000
pbc                = xyz
comm-mode          = linear
nstcomm            = 1
comm-grps          = System

; MOŽNOSTI VÝSTUPŮ
nstxout            = 0
nstxtcout          = 10000
nstvout            = 0
nstfout            = 0
nstlog             = 10000
nstlist            = 10
ns_type            = grid
nstenergy          = 1000
energygrps         = System

; PARAMETRY PRO ELEKTROSTATIKU A VDW
coulombtype        = pme
rcoulomb           = 1.0
rlist              = 1.0
rlistlong          = 1.6
rvdw               = 1.4
dispcorr           = EnerPres
table-extensing    = 1
fourierspacing     = 0.12

; PARAMETRY PRO „COUPLING“ ALGORITHMY
tcoupl             = v-rescale
tc-grps            = System
tau-t              = 0.5
ref-t              = 310
pcoupl             = parrinello-rahman
pcoupltype         = semiisotropic
ref-p              = 1.013 1.013
tau-p              = 10.0 10.0
compressibility     = 4.5e-5 4.5e-5

; „CONSTRAINTS“
constraints         = all-bonds
constraint-algorithm = Lincs

```

leap-frog integrátor  
počáteční čas pro běh  
2fs časový krok  
počet kroků odpovídající 500 ns  
periodické okrajové podmínky v x y z  
odstraňuje těžiště translace  
frekvence pro odstranění těžiště pohybu  
skupina pro odstranění těžiště pohybu

neukládat trajektorii s příponou *.trr*  
ukládat trajektorii jako *.xtc* každých 20 ps  
neukládat ve výstupu žádné  $\vec{v}$   
neukládat ve výstupu žádné  $\vec{F}$   
aktualizovat soubor *md.log* každých 20ps  
aktualizovat seznam sousedních částic a dalekodosahových  $\vec{F}$   
metoda ke generování seznamu sousedních částic  
ukládat energie v souboru *.edr* každé 2 ps  
skupiny zapisovat do souboru *.edr* (energie)

model dalekodosahových elektrostatických interakcí  
 $r$  pro Coulombický cut-off (krátkodosahové  $\vec{F}$ )  
 $r$  cut-off pro tvorbu listu sousedních částic (krátkodosahové  $\vec{F}$ )  
 $r$  cut-off pro tvorbu listu sousedních částic (dalekodosahové  $\vec{F}$ )  
 $r$  pro Van der Waals cut-off (krátkodosahové  $\vec{F}$ )  
korekce disperze jak na tlak, tak potenciál PME mřížky  
rozšíření ne vazebných potenciálů nad největší  $r$  cut-off  
rozteč mřížky „Fourier-space“

použitý termostat  
„kaplovat“ celý systém  
časová konstanta pro termostat  
požadovaná teplota  
použitý barostat  
separátní udržování tlaku na box v xy a z  
požadovaný tlak  
časová konstanta pro barostat  
izotermická kompresibilita

omezit všechny vazby  
omezovací algoritmus zajišťující, že vzdálenost mezi  
hmotnými body zůstane zachována (z Newtonovského  
pohybu tuhého tělesa)

Obr. 90. Ukázkový soubor *grompp.mdp* s parametry pro MD simulaci. Vysvětlivky:  $\vec{F}$  = síla [k]/(mol · nm),  $p$  = tlak [bar],  $r$  = délka [nm],  $t$  = čas [ps],  $T$  = teplota [K],  $\vec{v}$  = rychlost

[nm/ps]. Podrobnější vysvětlení jednotlivých zde uvedených parametrů (a dalších) je dostupné v manuálu Gromacsu, např. na adrese <http://www.gromacs.org/> k 27.07.2017.

Přehled mnou používaných souborů pro spuštění základní MD simulaci peptidu H1 v membráně POPC je shrnut v následující Tab. 19. Složení jednotlivých souborů se liší jen nepatrně podle konkrétního složení daného systému pro MS simulaci.

Název souboru	Obsah souboru
<i>./forcefield.itp</i>	složka, obsahující ostatní FF pro výpočty
<i>POPC.itp</i>	FF pro POPC membránu
<i>U_H1_V</i>	skript pro spuštění výpočtu MD simulace na počítačovém klastru
<i>U_H1.itp</i>	FF pro helix H1 kolicinu U
<i>conf.gro</i>	souřadnicový soubor
<i>forcefield.itp</i>	FF (vazebné a nevazebné interakce)
<i>grompp.mdp</i>	soubor s parametry pro MD simulaci
<i>ions.itp</i>	FF pro ionty
<i>tip3p.itp</i>	FF pro rozpouštědlo
<i>topol.top</i>	soubor topologie
<i>topol.tpr</i>	nejdůležitější soubor umožňující samotný výpočet; vygenerován <i>grompp</i>

**Tab. 19:** Přehled mnou používaných souborů pro spuštění základní MD simulaci

### 9.2.1 Spuštění a výpočet MD simulace

Prakticky je spuštění MD simulace velmi snadné. Základem je vygenerovaný soubor *topol.tpr* pomocí příkazu *grompp*. Nejedná se o kompilaci, jde o výstup preprocessingu – dochází zde ke sloučení informací do formátu, který je čitelný programu. *Topol.tpr* v sobě tudíž obsahuje všechny informace z původních souborů *conf.gro*, *topol.top*, *grompp.mdp* a souborů *.itp*. Samotná MD simulace se poté spustí příkazem *mdrun*. Tím ji lze spustit v konkrétním adresáři na běžném stolním počítači a opětovně ji zastavit Ctrl+c. Průběh jejího výpočtu v terminálu lze vypsát např. příkazem *mdrun -v*, který jsem během práce používala standardně. Zpravidla se však k výpočtům trajektorií používají počítačové klastry, ve kterých je *mdrun* definováno ve vytvořeném skriptu, k jehož spuštění (a následnému spuštění výpočtu MD simulace) obvykle slouží příkaz *qsub* (př. *qsub U\_H1\_V*), *qdel* a ID počítané trajektorie MD výpočet naopak zastaví (př. *qdel 11111*).

## 9.3 Používání softwaru pro MD simulace krátkých peptidů v lipických membránách

### 9.3.1 Generování jednotlivých peptidů ze struktury celé CTD a FF

Stažené struktura z PDB, nebo struktura vygenerovaná softwarem Rosetta či iTasser, byla nejprve zobrazena v programu VMD. V hlavním panelu nabídky VMD byla přes možnost „Graphics zvolena možnost „Representations“, nová definice atomů v souboru se vytvoří pomocí tlačítka „Create Rep“ a do okna pro „Selected Atoms“ se vypíší konkrétní atomy – např. „*r 15 to 30*“, kde *r* značí residuum. Výsledkem je zobrazení zvolených residuí s defaultní vykreslovací metodou „lines“. Při tvorbě simulačních systémů je ale velmi užitečné pracovat se zobrazením pomocí metody VDW, kterou lze zvolit z nabídky „Drawing Method“. Pro uložení souřadnic pouze vybraných atomů je potřeba levým tlačítkem myši označit pracovní soubor v hlavním panelu (*ID* např. 0). Potom na něj stačí kliknout pravým tlačítkem myši a zvolit možnost „Save Coordinates“. V novém nabídkovém okně se posléze vybere, které atomy v souboru (dříve definované v reprezentaci), se mají uložit. Tlačítkem *Save* se otevře další okno s předdefinovaným adresářem, kam souřadnice uložit a očekává se definování názvu souboru s příponou *.pdb* (ve VMD lze ale samozřejmě struktury ukládat i v jiných formátech). Gromacs je pak už sám schopný v daném adresáři pomocí nástroje *editconf* přepsat souřadnice formátu s příponou *.pdb* na *.gro*, který je tím výchozím souborem obsahujícím souřadnice pro MD simulace v Gromacsu. Mnou běžně používaný příkaz k tomuto účelu (včetně příkladu) je uvedený níže, přičemž platí, že v Gromacsu obvykle *-f* v definuje výchozí soubor, *-o* pak vytvoří nový soubor pod zadaným názvem.

```
editconf -f X.pdb -o Y.gro
(př.: editconf -f resid-7-34.pdb -o H1-orig.gro)
```

Takto byly získány všechny souřadnice jednotlivých struktur „umělých peptidů“ z původní celé CTD.

V této práci bylo třeba připravit peptidy pro MD simulace v dané orientaci, tudíž byly tyto „umělé peptidy“ nejprve v příkazové řádce srovnány s osou *x*. K tomu slouží opět nástroj *editconf*, který obvykle s kombinací možnosti *-princ* poskytne zvolenou strukturu v horizontální pozici:

```
editconf -f resid-7-34.gro -princ -o H1-horizontally.gro
```

Pro získání peptidu ve vertikální orientaci (tj. rovnoběžně s osou *z*) stačí v programu už jen zadat:

```
editconf -f H1-horizontally.gro -rotate 0 90 0 -o H1-vertically.gro
```

Možnost *-rotate* rotuje strukturou v libovolném směru pod libovolným úhlem. Poté je doporučeno zkontrolovat ve VMD strukturu, zda došlo ke správnému otočení molekuly. Pokud se totiž získávají různé struktury z mnohem větších kompaktních celků a je třeba získat strukturu v přesně daném směru, např. aby každý peptid vždy směřoval N-koncem nahoru v ose z, je samozřejmě potřeba v příkazu změnit úhel rotace (např. z 90 ° na 270 °).

Po získání struktury peptidu v dané orientaci je třeba zavolat příkaz *pdb2gmx*, který pracuje s předdefinovanou databází residuí a na jejím základě vygeneruje soubor *topol.top*, *posre.itp* a *conf.gro*.

```
pdb2gmx -f H1-vertically
```

Po zadání tohoto příkazu je uživatel vyzván, aby si zvolil silové pole. V mém případě bylo pro MD simulace peptidů zvoleno silové pole AMBER99SB-ILDN, tzn. do příkazové řádky bylo zadáno konkrétní číslo silového pole (př. 6) a potvrzeno. Následně byl uživatel vyzván k zadání modelu pro molekuly vod, opět stačilo zadat např. 1 pro model TIP3P a potvrdit. Program následně vypsal svůj postup: kontroloval duplikované atomy, generoval chybějící vodíky a přidával na konec struktury C- a N-terminály, tvořil nové vazby, generoval úhly, dihedrály a torse. V terminálu také vypsal hmotnost struktury v a.m.u. a celkový náboj struktury. Ten je užitečné si zapamatovat pro další krok přípravy peptidu pro simulaci.

Výsledkem je tedy soubor *conf.gro* obsahující nové opravené souřadnice zadané struktury. Vodíky jsou do struktury přidávány velmi běžně, protože velmi často u některých PDB struktur získaných rentgenovou krystalografií chybějí. Touto metodou lze získat detailní strukturu proteinů a prostorové souřadnice těžkých atomů, bohužel však rozlišení této metody není vždy dostatečné k tomu, aby byla schopná také detekovat polohy vodíků. Záleží potom na autorovi konkrétní struktury, jak dobrou provede validaci struktury před samotným nahráním do PDB. Výborná vlastnost tohoto programu je, že se také na N-konec a C-konec „umělého peptidu“ přidají odpovídající residua. Výsledkem je tedy soubor souřadnic kompletního „umělého peptidu“ získaný z původně velké struktury, který už sloužil jako výchozí pro následující simulaci daného peptidu.

Dalším vygenerovaným souborem pro danou strukturu peptidu byl *posre.itp*, který v sobě nese informace o fixaci vazeb pro Ca uhlíky aj., resp. obsahuje „*position restraint*“ pro všechny těžké atomy, vodíky přidané *pdb2gmx* fixovány nejsou. Při definování *posre.itp* v *topol.top* se



struktura peptidu v průběhu simulace nebude měnit, což se hojně využívá např. v přípravných MD simulacích.

Třetí vygenerovaný soubor *topol.top* v sobě obsahoval silové pole pro daný peptid. Je užitečné ihned tento soubor *topol.top* zkopírovat na soubor nový, např. *U\_H1.itp*. V příkazové řádce jednoduše:

```
cp topol.top U_H1.itp
```

Tento soubor pak v hlavičce obsahuje použitou verzi programu, zadanou formulaci do příkazové řádky a název zadaného silové pole. Tyto informace jsou komentovány středníkem a program jako takový je tudíž při výpočtech nevidí. Dále v souboru následuje označení „*moleculetype*“. V ní je definován název molekuly, který se bude následně vyvolávat programem pro simulaci. Je velmi užitečné zvolit krátké a praktické pojmenování, např. *U\_H1*. Dále v souboru následují parametry pro „*atoms*“, kde jsou očíslována jednotlivá residua, resp. atomy tvořící jednotlivé aminokyseliny. Kromě jejich názvů jsou uvedené také jejich atomové hmotnosti a náboje. Poté jsou definovány vazby, atomové páry, úhly, dihedrály a torze. Na konci souboru je definovaný podmínkový cyklus pro soubor *posre.itp*. Ten znamená, že pokud ve vyvolávacím souboru uvedeme a definujeme „*posre.itp*“, v simulaci pak budou fixovány vazby konkrétní struktury. Soubor lze samozřejmě pojmenovat jinak, což je užitečné při práci s vícero různými strukturami a jejich sloučením do jednoho systému, např. pro přidání různých helixů do jedné membrány. A v tomto případě se musí pamatovat na to, že nestačí pouze změnit název souboru „*posre.itp*“ a tento název pak jen uvést v souboru *topol.top* definujícím už pouze topologii. Vždy se musí jeho nový název upravit v této podmínce na konci souboru FF s příponou *.itp*. Vše ostatní, co je za řádkem ukončujícím podmínku pro fixaci vazeb, je vhodné smazat, tzn. smazat vše od spodu souboru po „*Include water topology*“ včetně.

Poté byl vytvořen topologický soubor s názvem „*topol.top*“. V něm je vždy odkázáno na silová pole pro všechny molekuly a atomy, které se v simulaci měly vyskytovat a ev. bylo odkázáno na soubor *posre.itp*. Následně bylo zapsáno konkrétní složení systému - dané molekuly, ionty, a jejich počty. To je potřeba zapsat v pořadí podle souboru *conf.gro*. Pokud jsou na jeho začátku souřadnice pro peptid, pak následují souřadnice membránových lipidů, molekul vod a nakonec iontů, pak musí být dodrženo stejné pořadí i při psaní topologie.

### 9.3.2 Příprava membrány POPE:POPG

Postup přípravy byl následující. V  $\beta$ -verzi MemBuildru (Ghahremanpour et al., 2013), dostupné na adrese <http://bioinf.modares.ac.ir/software/mb2/>, jsem zvolila složení membrány tak, aby celkový počet jejich lipidů byl 128, konkrétně je to 100 molekul POPE a 28 molekul POPG. Pokud by membrána měla simulovat naprosto reálnou vnitřní membránu *E. coli*, mělo by být k tomto poměru přidáno ještě cca 6 molekul kardiolipinu (DPG), nicméně DPG pro zjednodušení modelu, již do této membrány nebyl přidán. Místo něj byly přidány další POPE a POPG, takže jejich celkový poměr byl 78 % : 22 %.

Při tvorbě membrány skrz tuto webovou stránku stačí již nechat předvyplněné membránové konfigurace a zvolit si z nabídky silových polí to vhodné FF. Jelikož v ostatních simulacích používám FF Amber, vybrala jsem „*Slipid/Amber*“. Dále je potřeba označit symetrické vyplnění vrstev, pokud je žádoucí mít složení lipidů v obou monovrstvách stejné, a vyplnit procentuální počet vybraných lipidů pro jednu vrstvu, druhá vrstva se vyplní automaticky. Poté stačí už jen definovat koncentraci soli, zaškrtnout „*Run Minimization*“, a tlačítkem „*Build Membrane*“ se online vygeneruje odpovídající množství lipidů, molekul vod a iontů. Poté je možné stáhnout PDB soubor membrány, soubor se souřadnicemi ve formátu *.gro* pro Gromacs, topologii s příponou *.top* a soubor o průběhu minimalizace (*.log*).

Silové pole tu lze také stáhnout (*Slipid/Amber.ff*). Já jsem tuto možnost nevyužila. Silová pole pro lipidy POPE a POPG jsem stáhla ze stránky <http://www.fos.su.se/~sasha/SLipids/Downloads.html>, kterou váženému čtenáři vřele doporučuji. Kromě silových polí pro různé lipidy je zde možné také stáhnout souřadnice různých již připravených membrán pro různé teploty, vzorkový soubor topologie a ukázkový příklad souboru *.mdp* s parametry potřebnými pro spuštění simulace v Gromacsu.

Takto vygenerovaná membrána se nedá použít ihned k simulaci. Proto byl upraven standardní soubor topologie pro membránu POPC, byl do něj přidán odpovídající počet molekul POPE a POPG na místo POPC a dále bylo stažené silové pole pro lipidy připsáno do tohoto souboru (podrobný popis je v předchozí kapitole). Následně byla spuštěna simulace této membrány při standardních podmínkách simulace pro dobu 10 ns během níž došlo k propojení monovrstev. Souřadnice z trajektorie v 10 ns byly následně upraveny – byly smazány původní ionty soli přidané online programem a pomocí nástroje *genbox* přidáno takové množství molekul vody, aby byla membrána srovnatelná s membránou POPC. Následně pomocí utility *genion* byly přidány ionty

v koncentraci 0,154 M NaCl, a 22 protiiontů  $\text{Na}^+$  k 22 negativně nabitým POPG. Protiiont jsem přidala do membrány tak, že jsem na konci souřadnicového souboru *.gro* smazala veškeré parametry pro oba vodíky navázané na atomu kyslíku molekuly vody a atomová označení kyslíku přepsala na NA. Ostatní souřadnicové parametry pro bývalý kyslík, kromě číselného označení pořadí atomů, jsem zachovala. Takto bylo v souboru *conf.gro* přepsáno 22 molekul vod za 22 atomů  $\text{Na}^+$ . Nesmí se však následně zapomenout na upravení celkového počtu atomů v hlavičce souboru. Poté jsem upravila soubor *topol.top*, ve kterém jsem smazala odpovídající počet molekul vod a připsala odpovídající počet iontů  $\text{Na}^+$ . Takto připravený systém byl zminimalizovaný při 1000 krocích a spuštěn při standardních podmínkách na 200 ns. Souřadnice této membrány v 200 ns MD simulace byly následně využity ke vložení peptidu H1.

### 9.3.3 Příprava solvatovaného peptidu pro MD simulaci

Peptid rozpuštěný ve vodě, ev. CTD ve vodě, byl připraven v Gromacsu pomocí definování velikosti boxu, která byla stejná jako pro simulaci peptidu v membráně, a nástroje *genbox*. Použit byl model spc216 molekul vod a silové pole pro tyto molekuly v simulacích bylo TIP3P. Postup byl následující. Nejprve byl přepsán původní soubor se souřadnicemi peptidu, *conf.gro*, na soubor, který na svém konci obsahuje informaci o velikosti pbc boxu. Tato velikost byla definována nástrojem *editconf*:

```
editconf -f conf.gro -box 6.5 6.5 7.5 -o conf.gro
```

Poté byly pomocí příkazu *genbox* a *genion* přidány molekuly vody a poté požadovaný typ a počet protiiontů ke kompenzaci nábojů nabitých peptidů.

```
genbox -cp conf.gro -cs -p topol.top -o conf.gro
```

Možnosti *-cp* a *-cs* jsou nutné jako input: *-cp* definuje strukturu v boxu, která má být solvatována (např. peptid H1), *-cs* potom vkládá zvolený model molekul vody do boxu, *-p topol.top* není nutné uvádět, nicméně to doporučuji. Výsledkem je pak nový soubor *conf.gro* se solvatovaným proteinem. Jako doplněk výsledku tohoto příkazu se také automaticky dopíše počet přidávaných molekul rozpouštědla do souboru s topologií, pokud je tak přikázáno *-p topol.top*. Pro přidání protiiontů jako kompenzace nábojů na peptidů za účelem získání elektroneutrálního systému, se může použít buď nástroj *genion*, nebo ruční přepsání molekul vody, resp. atomů kyslíku z molekul vody za ionty  $\text{Na}^+$  v *conf.gro*. Nástroj *genion* udělá tuto práci prakticky za uživatele a to následovně:

```
genion -s topol.tpr -n index.ndx -np 3 -pname NA -pq 1 -p  
topol.top -o conf.gro
```

Jako input jsou nutné *-s topol.tpr* a *-n index.ndx*. přičemž *-s topol.tpr* je soubor shrnující všechny potřebné informace pro spuštění výpočtu simulace, a *-n* volá indexový soubor, např. *index.ndx*. Tento soubor je potřeba vytvořit příkazem *make\_ndx -f conf.gro* a definovat v něm molekuly rozpouštědla, *-p topol.top* opět zajistí, že se do topologie vloží název a počet přidávaných iontů. *-np 3* definuje počet vkládaných kationtů (pro anionty se používá *-nn*), *-pname NA* označuje jméno kationtu (*-nnname* aniontu), a *-pq 1* určuje náboj kationtu (*-nq* náboj aniontu), nicméně *-pq* není třeba zadávat, pokud jsou vkládány monovalentní ionty, hodnota 1 (ev. -1) je generována automaticky. Po zadání tohoto příkazu je uživatel vyzván, aby označil, které molekuly v systému mají být nahrazeny zvolenými ionty - je třeba zvolit číslo odpovídající molekulám SOL (molekulám vody). Jak je patrné, uživatel musí nejdříve vygenerovat soubor *topol.tpr*, pro jehož generaci příkazem *grompp.mdp* musí být definována topologie, soubor se souřadnicemi, a soubor obsahující parametry simulace. Jednoduše se dá obsah *topol.tpr* představit jako „3 v 1“ získaný přepisem různých informací od uživatele do jedné velké informace v jazyce, kterému program bezpečně rozumí. Z hlediska času, pokud jde o přidání několika málo iontů do systému, je leckdy výhodnější atomy přepsat ručně v souboru *conf.gro* a následně v *topol.top*. Pokud je však potřeba do systému přidat větší množství iontů různého typu, je výhodnější použít tento nástroj.

Molekuly vody pro membránu odpovídající *E.coli* byly přidány předchozím způsobem v takovém poměru, aby odpovídaly počtu molekul vody v membráně POPC a poté byla přidána v koncentraci 154mM sůl NaCl. Použit byl příkaz:

```
genion -s topol.tpr -n index.ndx -conc 0.154 -p topol.top -o  
conf.gro
```

Pokud si uživatel nemůže vzpomenout, jaké možnosti má který nástroj, vždy je tu eventualita zavolat nápovědu pomocí *-h*, (např. *genion -h*). Na příkazové řádce se pak vypíše manuál použití konkrétního příkazu.

### 9.3.4 Návod na vložení jednoho peptidu (ev. dvojic a trojic peptidů) nebo čtyř samostatných peptidů současně, či šesti samostatných peptidů po jednom) do lipidové membrány v Gromacsu

#### 1. Přidat peptid k membráně

To lze udělat několika způsoby, např. předem připraveným skriptem. v Pythonu, aj.

- Např. já používám Pythonovský skript, který automaticky vezme 1. zadaný soubor *conf.gro* a přidá na jeho konec za souřadnice pro atomy 2. zadaný souřadnicový soubor a vygeneruje 3. soubor s příponou *.gro* obsahující souřadnice obou inputů. Vše je v příkazové řádce vykonáno jedním příkazem. Výhodou je rychlost a to, že se získá jeden soubor s už specificky číselně označenými souřadnicemi. Při dalších úpravách ve VMD tedy stačí jen označit např. příslušná residua proteinu a lze s nimi dále pracovat. Běžný uživatel Pythonu si takový skript může napsat mnoha různými způsoby. Doporučuji mít v konečném souboru jako první souřadnice peptidu, pak až membránu s vodou. V opačném případě by nešlo použít snadného přepisování molekul vody za ionty při přidávání protiiontů do souboru a bylo by nutné využít nástroj *genion*.

```
př.: ./groadd.py protein.gro membrane.gro >new.gro
```

- V nejhorším případě, když si člověk neví rady při skriptování, lze souřadnice peptidu dopsat za souřadnice membrány v jejím *conf.gro* souboru ručně a v jeho hlavičce upravit počet atomů v souboru. Což se pro velmi malé molekuly zvládnout dá, ovšem při pracování s delšími molekulami doporučuji raději využít skripty nebo různé grafické programy schopné slučování struktur do jednoho formátu (*.pdb*, nebo *.gro*).

## 2. Ve VMD posunout peptid vedle membrány z boku a v orientaci, která je požadovaná.

- Následné zobrazení nově vygenerovaného souboru s oběma, či vícero strukturami ve VMD obsahuje jejich nesprávné pozice, je tedy třeba protein přesunout z membrány pryč na konkrétní pozici, která je požadovaná pro výslednou orientaci, z boku membrány.

```
vmd new.gro
```

To lze udělat opět mnoha způsoby. Nejsnazší je si před zobrazením souboru ve VMD zapamatovat počet aminokyselin, nebo resp. jejich číselných označení, a pak vypsát jejich počáteční a konečné číslo (př. *r 1 to 28*) v okně pro definování atomů k reprezentaci. Vepsání „*protein*“ do definice atomů doporučuji pouze pro sledování grafického vývoje systému v průběhu simulace, ne pro ukládání souřadnic. Program VMD s touto definicí atomů pomocí „*protein*“ často nebere v potaz první a poslední aminokyselinu v sekvenci.

- Jiná možnost ve VMD je např. přes kartu grafické reprezentace v hlavním panelu a v kartě „*Selections*“ využít tabulky „*Keyword*“ a zvolit „*resname*“. Pokud na to uživatel 2x klikne myší a stejně označí i příslušné hodnoty, tedy jednotlivé aminokyseliny v souboru, vše se

automaticky vypíše na řádek pro zvolené atomy. Potvrzením se vytvoří nová definice s atomy, v tomto případě celými aminokyselinami. Tím je definován protein.

- Před vlastním přesunutím proteinu nejprve ale doporučuji zvolit pro peptid vykreslovací metodu *VDW* (nabídka „*Drawing Method*“) a skrýt zobrazení membrány dvojitým kliknutím myši v okně s vypsanými definicemi „*all*“ či „*membrane*“. Pak lze použít nabídku „*Mouse*“ hlavního panelu, zvolit „*Move*“ a „*Rep*“, a následně v okně zobrazení struktur označit myši molekulu k přesunutí. Tahem myši ji přesunout za hranici membrány, která by měla být v zobrazení skryta. Aby to šlo provést, je ale třeba mít v okně grafické reprezentace označenou definici přetahovaných molekul, nebo atomů!
- Následně lze zkontrolovat umístění opětovným zobrazením membrány a pozici proteinu vůči ní doladit. Vzdálenost peptidu od membrány musí být dostatečná, nesmí dojít k překryvu povrchů nebo van der Waalsovských poloměrů.
- Pro uložení souřadnic pouze vybraných atomů (nebo i celého systému) je třeba levým tlačítkem myši označit pracovní soubor v hlavním panelu VMD a přes dvojitě kliknutí pravým tlačítkem myši zvolit možnost „*Save Coordinates*“. Poté stačí vybrat, které atomy v souboru (nebo celý systém), se mají uložit. Je třeba uložit souřadnice nového souboru s příponou *.pdb*.
- Následně se tento soubor v příkazové řádce přepíše pomocí nástroje *editconf* na *.gro* soubor.

```
př.: editconf -f H7-toMem.pdb -o H7-toMem.gro
```

```
př. editconf -f all.pdb -o all.gro
```

- Ve svém postupu jsem vždy uložila celý protein v nově vygenerovaných souřadnicích vůči membráně, a pak jej opětovně přidala k souboru pouze s membránou pomocí skriptu *./groadd.py*.

```
př.: ./groadd.py H7-toMem.gro membrane.gro >new.gro
```

### 3. Definovat nový pbc box

- Nyní je třeba definovat rozměry nového pbc boxu. Já např. zkopírovala hodnoty původního pbc boxu ze souboru pro samostatnou membránu a zvětšila je obvykle v každé straně o 0.4 nm. K tomuto účelu se používá nástroj *editconf* a možnost *-box*.

```
př.: editconf -f new.gro -box 6.6 6.9 7.6 -o conf.gro
```

### 4. Provést minimalizaci alespoň při 1000 krocích:

- Takto připravený *conf.gro* se musí minimalizovat. K minimalizaci lze použít jednoduše úpravný standardní soubor *grompp.mdp*, ve kterém se změní MD integrátor z „*md*“ na „*steep*“; a upraví se počet kroků („*nsteps*“). Ve svém příkazu používám u *grompp* příponu *-maxwarn*, která ignoruje počet zadaných chyb. Je to z toho důvodu, že v souborech mám molekuly vody označené jako *tip3p* (místo obvyklého *sol*) a při načítání se přepisují názvy těchto atomů. Další chyby a poznámky jako např. použití souboru *.posre* a nevložení protiiontů do systému, může být v tomto kroku ignorováno.

```
př.: grompp -f grompp_min.mdp -maxwarn 5
```

- Po spuštění minimalizace souboru příkazem výše může dojít ke dvěma možnostem:

#### 4.1. Minimalizace se zastaví:

- V tom případě je potřeba přesunout peptid do jiné pozice vedle membrány, např. posunout v některém ze směrů, nebo ho úplně přesunout na jinou stranu membrány; někdy je potřeba postup opakovat i 3x; a pokračovat dále od 3. bodu.

#### 4.2. Minimalizace se vypočítá a vygeneruje se soubor *confout.gro*, přidání protiiontů

4.2.1. Úspěšnou minimalizací je velká naděje na to, že systém bude brzy připraven, jelikož došlo k vložení peptidu do membrány v požadované pozici. V tomto kroku je vhodné vzít minimalizovaný soubor a přidat do něj protiionty.

4.2.2. Následně **zkusit spustit MD simulaci při 150 K (ev. 50 K)**, *parrinello-rahman* barostatu a *v-rescale* termostatu

```
př.: editconf -f confout.gro -o conf.gro
      grompp -maxwarn 5
      mdrun -v
```

4.2.2.1. Pokud simulace poběží alespoň na 20 ps, pak použít snímek z jeho 20ps souřadnic a membránu s peptidem ohřát na 230 K opět při 20 ps. Následně je třeba vzít snímek se souřadnicemi v 20 ps této MD simulace a opětovně spustit při 310 K na 80 ps. Souřadnice v 80 ps obvykle tvořily můj připravený simulační systém. Je však potřeba zkontrolovat, zda do membrány již nenatekla nějaká voda. V souboru *topol.top* se pak už smaže definice pro fixaci vazeb těžkých atomů tím, že se smaže definice pro *posre.itp*.

- 4.2.2.2. Pokud se MD simulaci spustit nepodaří, osvědčilo se mi snížení teploty na 50 K (ev. až na 10 K) a membránu s peptidem pomalu rozhýbat a ohřívat. Někdy bylo třeba změnit termostat a barostat na Berendsenův.
- 4.2.2.3. Pokud se ale simulace při výše uvedených parametrech a ani opětovné minimalizaci přesto po pár krocích zastaví nebo se vůbec nespustí, je nutné se vrátit ke kroku č.2), vložit peptid k membráně v jiné pozici a postup opakovat. Ve většině případů se ale obvykle při 50 K MD simulaci podařilo spustit.