**INTRO TUTORIAL 2 (Dinamica molecolare)**Ripartiamo dai file ottenuti con la minimizzazione; vediamo che xmgrace ci restituisce il potenziale decrescente. *Possiamo salvare delle nuove figure a partire dal grafico zoomato (fileàprint setup e possiamo salvare in diversi formati, vettoriali o grafici). Suggerimento, riduciamo il grafico da dentro il programma, ovvero riduciamo gli assi, sennò quando andiamo a salvare l’immagine e ridimensionarla fa casini. Togliamo la legenda potential (display legend), possiamo cambiare i titoli ecc.*

Proviamo a calcolare un altro tipo di potenziale: gmx energy -f .edr -o potential2.xvg

Scegliamo ad es 1 (angoli) poi 2 (diedrali) , 6 (Coulomb) e 7 (Lennar Jones)

Abbiamo calcolate diverse grandezze e per plottarle tutte insieme scriviamo xmgrace -nxy potential2.xvg ; è possibile che alcune curve si sovrappongono ma con la legenda e zoom riesco a vederle bene. Innanzitutto, vedo che la LJ è tutta positiva però possiamo comunque far convergere l’energia *(positive significa che alcune particelle sono a sx del loro minimo ?),* mentre quella di Coulomb è negativa (attrattiva).

**STRUTTURE SECONDARIE (#do\_dssp)**

*Possiamo usare un tool online per studiare le strutture secondarie, ad es Stride. Carico il pdb iniziale della proteina. Ci racconta come è fatta la catena, se è alpha elica,ecc. Calcola tutti i phi e psi e ci dice che struttura secondaria ha ogni residuo.*

Possiamo altrimenti calcolare la struttura secondaria durante la minimizzazione, dell’intera traiettoria à usare gmx do\_dssp, che si installa con sudo apt-get install dssp.

Gmx do\_dssp -f 1em.trr -s 1em.tpr -dt 10

-dt 10 riduce il numero di frames, così poi riusciamo a stampare il eps senza problemi

Ci chiede di chi vogliamo calcolare la struttura secondaria: 1 (proteina)

Abbiamo diversi output:

* **scount.xvg** ci dice come e quali residui sono coinvolti in alpha eliche e coil
* **ss.xpm** è una matrice, ci dice su x gli step, su y il residuo che possiamo visualizzare con less. Possiamo visualizzarla mediante una matrice colorimetrica mediante

gmx xpm2ps -f ss.xpm -o plot.eps

**eps** è un file che ci serve a stampare; lo vediamo dalla cartella con doppio click. Il residuo non cambia struttura secondaria nel tempo perché non siamo ancora in dinamica

**E’ importante dire che le strutture secondarie della proteina senza acqua sono più variabili mentre l’acqua permette alla proteina di stabilizzarsi meglio perché espone delle parti idrofiliche e inoltre l’acqua fa un po’ da scudo permettendo alla proteina di riarrangiarsi meno.**

**DINAMICA MOLECOLARE**: abbiamo ottenuto il file gro che è l’output della minimizzazione ma di per sé non caratterizza la dinamica e non abbiamo fatto un campionamento del punto di minimo.

*Dobbiamo quindi studiare il sistema nel tempo, in quanto è un integratore à abbiamo bisogno delle velocità del sistema à temperatura perché ad una certa velocità corrisponde una certa energia cinetica che si può esprimere in termini di temperatura.*

**Quando accoppiamo in temperatura, visto che non accoppiamo mai in pressione, usiamo un metodo NVT, perché il volume cambia.**

Dobbiamo quindi fare un’altra simu, ci serve:

* file .gro
* file .top
* file .mdp : dobbiamo modificare questi parametri per la simu (cp template.mdp md.mdp ànano md.mdp)

**Prima simulazione (position restrains)**

La prima parte non si tocca teoricamente, ma in realtà su DEFINE metto -DPOSRES \*à **dinamica di position restrain** à significa fare una dinamica molecolare in cui manteniamo vincolati gli atomi del sistema a delle molle perché mentre riscaldiamo non vogliamo che la proteina possa distorcersi troppo (le dinamiche/cinetiche di riscaldamento possono essere molto rapide e potrebbero dare cinetiche non volute perché i vettori velocità non sono ancora orientati come la distribuzione di temperatura)

*\*se ho messo POSRES automaticamente mi crea un file posre.itp che aggiunge alla topologia, file che esclude gli idrogeni prendendo il backbone e lo aggancia a terra con delle molle mediante funzioni armoniche à evita che si muova troppo e che fluttui attorno alla posizione inziale senza distorcersi*

La parte da modificare del mdp è la seguente:

* usiamo come un integratore *md* che è un leap frog, anziché verlet o velocity verlet. (approfondisci teoria)
* time step dt è in picosecondi in questo caso è 0.002 (2 fento secondi, 10^-15), possiamo usare questo valore perché abbiamo le masse aumentate e abbiamo messo come costrains gli all bonds nel template. Vogliamo fare come dinamica una fase di riscaldamento che sarà dell’ordine delle centinaia di pico secondi, quindi nsteps=t\_tot/0.002 picos=100 ps/0.002 ps
* Cancello gli emtol e emstep perché servono per la minimizzazione.
* OUTPUT CONTROL: salvo ogni 100,100,1000,100,100, energy groups niente
* ACCOPPIAMENTO TERMICO (temperature coupling)

Tcoupl = v-rescale (accoppiamento debole)

Tc-groups = Water non-water (evitiamo il problema del soluto freddo e il solvente caldo)

0.1 0.1

300 300 stiamo accoppiando a 300K, nonostante sia poco verosimile rispetto alla T del corpo, ma il campo di forze è stato validato a questa temperatura e quindi è meglio

* L’ultima parte del GENERATE VELOCITY, mettiamo no (senza punto).

*Un’altra opzione per l’accoppiamento in temperatura sarebbe partire direttamente da 300K senza fare il riscaldamento fino a 300, potrei mettere yes generando direttamente una distribuzione di velocità random a 300K mediante una distribuzione di Maxwell-Boltzmann. I position restrains servono lo stesso, nonostante gli assegni le velocità, esse sono però random, quindi, gli atomi si muovono ovunque perché le direzioni sono casuali, ma voglio comunque vincolarle con position restrains per farle orientare in modo concorde. Si riescono ad orientare stabilendo un certo equilibrio per il fatto che interagiscono un po’ tra di loro.*

**Ho quindi tutto pronto per fare la simulazione, dopo che ho modificato il file mdp con le nuove indicazioni per la dinamicaa (md.mdp), poi avrò il file gro derivante dalla minimizzazione e, infine, la topologia.**

* Genero file tpr: Facciamo grompp gmx -f md.mdp -p topol.top -c 1em.gro -o 1em.tpr
* Facciamo la simulazione con gmx mdrun -s 1em.tpr -v -deffnm 1em

Durante la simulazione il sistema sta, per ogni step, integrando le equazioni di Newton e aggiunge anche ciò che serve per l’accoppiamento termico. Man mano la temperatura crescerà, il sistema si muove ma molto poco.

Se ora listiamo vediamo che abbiamo ottenuto:

* Mdout.mdp (mi restituisce i parametri di come è andata la simulazione)
* .log
* .edr
* (.tpr)
* .trr
* .cpt è uno stato del sistema ad un certo punto del sistema cosicchè se la simu crasha, ho un checkpoint da cui partire senza riniziare tutto

Guardiamo la traiettoria con vmd dandogli il .gro e poi il .trr che ci fa vedere la dinamica lungo gli step. Le righe non ci fanno capire nulla, le eliminiamo con il comando gmx trjconv

**NUOVA TRAIETTORIA (#trjconv)**

Ci permette di riscrivere la traiettoria non considerando le condizioni periodiche al contorno. Gli serve il tpr perché ha le masse e al suo interno ha la topologia. In output avrò un’altra traiettoria (.trr) ma che NON tiene conto del fatto che se un atomo salta il box allora anche tutto il resto della molecola lo faccia.

Gmx trjconv -s 1em.tpr -f 1em.trr -o 1em\_nojump.trr -pbc mol -ur compact

-ur compact ha senso solo con box dodecaedrico

Vogliamo come output tutto il sistema, quindi 0

**SALVARE UN ISTANTE DESIDERATO (#trjconv -dump)**

Possiamo estrarre anche un solo istante della traiettoria, in quanto 1em.gro è l’istante finale mentre ne vogliamo un altro. Usiamo sempre trjconv con l’opzione -dump, estrae il frame più vicino al tempo che gli chiedo, ad es al t=0

Gmx trjconv -s 1em.tpr -f 1em\_nojump.trr -o 1em\_0.gro -dump 0

Ci richiede cosa voglio e gli dico il sistema totale (0).

Ora ricarico su vmd per vedere se non ci sono più le righe: vmd 1em\_0.gro 1em\_nojump.trr

Si vede come l’acqua si muova, in quanto il position restrains è applicato solo alla proteina che si muoverà poco ma sempre più della minimizzazione, almeno vibra. Se guardo il backbone a cui ho legato le molle sarà molto fermo.

Vmd ci permette anche di plottare il Ramachandran plot che ci da un’idea di come si muovano gli atomi e come esplorino lo spazio durante la dinamica.

Andiamo a vedere come è andata la dinamica di riscaldamento per studiare la temperatura con

gmx energy -f 1em.edr -o T.xvg

Selezioniamo Temperature (13)

Vediamo che la media di T raggiunta è quella che volevamo intorno ai 300K, con un errore di 0.64

Andiamo a vederlo: xmgrace T.xvg, nei primi 3-5 picosecondi cresce da 0 a 300k, il primo punto a 300 è circa 1.1 ps. Che è il tempo che ci impiega a scaldare il sistema, ha una velocità di riscaldamento altissima quindi ha senso usare i pos res perché sennò indurremmo instabilità. Possiamo diminuire questo tempo aumentando molto la tau, accoppiando meno bene con il bagno termico e abbassando la curva.

Se zoomiamo intorno all’equilibrio, gli ultimi 20 ps intorno a 300 vediamo che oscilla attorno a quel valor medio, possiamo saperlo con precisione guardando il RMSD che ci restituisce facendo xmgrace.

E’ una gaussiana che possiamo plottare con un istogramma.

POTENZIALE, CINETICA E TOTALE

Gmx energy -f 1em.edr -o energies.xvg

Seleziono la potenziale, la cinetica e la totale (9,10,11)

Xmgrace -nxy energies.xvg

La curva dell’energia potenziale, zoommando sui primi ps, cresce perché stiamo scaldando il sistema facendo uscire il sistema dal minimo. Arriva a 300K e nei successivi ps (fino a 20) il sistema evolve fino a ricercare un altro minimo, facendo decrescere il suo valore.

La cinetica, cresce partendo da 0 (velocità iniziali nulle), raggiunge un massimo e poi rimane abbastanza stabile attorno al valore 3000 che non è altro che la somma di 1/2\*m\*v^2

**RMSD: DISTORSIONE DELLA STRUTTURA (#rms)**

E’ una media pesata di tutti gli spostamenti delle particelle rispetto alla posizione iniziale (info sulla deformazione), possiamo anche decidere di farlo rispetto ad un’altra posizione, tolti gli effetti della rototraslazione rigida (li potremmo considerare se guardassi due pezzi separati).

Gmx rms -f 1em\_nojump.trr -s 1em.tpr -fit none

Se non mettiamo l’output ce lo crea automaticamente ed avrà .xvg, -fit none non blocca alcuna rototraslazioni rigide (?)

Blocco il fit dei carboni alpha e voglio vedere l’RMSD della proteina

Ora guardiamo l’andamento del RMSD nel tempo, con xmgrace del file .xvg che ha creato: parte da 0 e sale velocemente durante il riscaldamento ma comunque perché è iniziata la simu, poi va in convergenza e si muove. I valori sono comunque molto bassi (0.05) simili alla minimizzazione (0.03), perché comunque abbiamo messo i position restrains.

**RMSF**

L’altra cosa che possiamo vedere è il grafico delle **fluttuazioni dei residui** (-res) durante tutta la simulazione, si perde l’info sul tempo:

gmx rmsf -s 1em.tpr -f 1em\_nojump.trr -res

scelgo la proteina

e poi plotto con xmgrace, occhio che sull’asse delle x non ho più il tempo ma il numero dell’amminoacido a cui mi riferisco. I bordi fluttuano un po’ di più rispetto al centro.

**Seconda simulazione (NO position restrains, T maggiore)**

Vogliamo guardare come il sistema evolve senza i position restrains à modifichiamo il file .mdp, quello nuovo lo chiamiamo md.mdp

* Togliamo la DEFINE perché non è più un position restrains
* togliamo angular;
* selezioniamo md come integrator,
* nsteps 50000,
* togliamo tutta la minimizzazione,
* output control come prima,
* tcoupl v-rescale
* t più alta, 500 500 K
* tau 1 1
* generiamo le velocità a 300K

Gen-seed è un numero random che inizializza la distribuzione di velocità di Maxwell-Boltzman (-1)

Gmx grompp -f md.mdp -c 1em.gro -p topol.top -o md.tpr -maxwarn 2

Gmx mdrun -v -deffnm md -s md.tpr

Puliamo la traiettoria con trjconv. Eliminando il problema delle condizioni periodiche al contorno

La proteina si muove molto di più perché siamo senza position restrains.