

Taller Dinámicas Moleculares

Sebastián Franco Ulloa y Gian Pietro Miscione

Módulo de Mecánica Molecular

Octubre 10, 2017

Este taller tiene como objetivo familiarizar al estudiante con la técnica computacional de dinámicas moleculares (MD). Para esto haremos una simulación del complejo entre una ADN girasa y su ligando nativo presente en uno de los cristales ya reportados. Las dinámicas se llevarán a cabo usando los programas gratuitos Gromacs y AmberTools. El orden de trabajo se ilustra en la Figura 1. Los resultados de estas simulaciones se visualizarán con el programa, también gratuito, VMD (*Visual Molecular Dynamics*).

1. Vi

Antes de comenzar con el taller es útil tener a la mano algunos de los comandos básicos del editor de texto llamado Vim (o Vi).

Accion	Comando
Abrir archivo	vi archivo.txt → Enter
Salir y salvar	ESC → :x → Enter
Salir y no salvar	ESC → :q! → Enter
Entrar al modo de edición	i
Salir del modo de edición	ESC

2. Parametrización del ligando

La simulación la haremos con el campo de fuerza AMBER99SB-ILDN, un campo de fuerza típico en la simulación de proteínas. Aunque Gromacs puede asignar los *atom types* necesarios a los átomos de una proteína, debemos crear *atom types* para nuestro ligando.

- Descargue del repositorio de Github del taller (**click aquí**) el archivo *4plb_wild_fixed.pdb*. Este es el sistema del cual haremos la dinámica molecular (código PDB: 4PLB). Para hacer esto (y, en general descargar cualquier archivo de la página del curso) debe ir al sitio web indicado, dar click en el archivo de interés, seleccionar la opción *RAW*, copiar el URL de la nueva página y ejecutar el siguiente comando:

wget *ElURLQueCopió*

- Abra el archivo descargado en Maestro y familiarícese con la proteína, el ADN y el ligando que vamos a simular.
- Vamos a empezar por calcular las cargas parciales de los átomos del ligando usando un método QM. Para esto se debe salvar un archivo *.pdb* de únicamente el ligando y convertirlo a formato *.com* con programas como GAUSSVIEW. En este caso puede descargar el archivo *31N.com* de la página del curso.
- Abra el archivo *.com* en un editor de texto (i.e. *vi 31N.com*) y edítelo para que el encabezado se vea de la siguiente manera. OJO: Las líneas en blanco se deben dejar a propósito. Asegúrese de dejar una línea en blanco al final del documento.

```
%chk=31N.chk
%mem=5000MB
%nproc=8
#P HF/6-31G* SCF=Tight Pop=(MK) IOP(6/33=2)

31N_SCF

1 1
C 42.51000 32.42800 28.66000
Acá siguen las coordenadas del resto de los átomos
```

- Lo anterior es un método compatible con los campos de fuerza AMBER para calcular cargas parciales de átomos.
- Para correr este cálculo necesitamos una máquina que tenga instalado *Gaussian*, el programa por excelencia para hacer cálculos QM.
- Conéctese a la dirección “estudiantes@157.253.60.12” desde una terminal con el siguiente comando.

```
ssh estudiantes@157.253.60.12
```

La contraseña es “computacional2017”

- Cree una carpeta con su nombre usando el comando *mkdir* (i.e. *mkdir NuevoNombre*).
- Vaya al directorio donde tenga el archivo *31N.com* y, usando el comando *scp*, llévelo a la carpeta que acaba de crear.

```
scp 31N.com estudiantes@157.253.60.12:~/SuNombre/
```

- Vaya a la carpeta que creó y ejecute el siguiente comando:

```
g16 < 31N.com > 31N.log
```

- El archivo *31N.log* tiene, en algún lugar, las cargas parciales de los átomos de nuestro ligando. Para extraerlas usaremos herramientas de AMBER (la empresa, no el campo de fuerza). Estas herramientas (conocidas como AmberTools) están instaladas en otra máquina.
- Conéctese usando el comando *ssh* al usuario “guest@157.253.72.13”. La contraseña es “12345”.
- Cree una carpeta con su nombre (si todavía no la hay) y lleve el archivo *31N.log* a esa carpeta. De ahora en adelante trabajaremos en esta carpeta.
- Para extraer las cargas y la geometría necesaria usamos la herramienta *antechamber* que hace parte de AmberTools:

```
antechamber -i 31N.log -fi gout -o 31N.mol2 -fo mol2 -c resp -nc 1 -m 1
```

- Las opciones *-nc* y *-m* piden la carga y la multiplicidad del sistema respectivamente, en este caso ambas son 1.
- El resultado de este cálculo es un archivo *31N.mol2* que contiene las coordenadas y las cargas parciales de todos los átomos.
- Por defecto, *antechamber* nombra al residuo “MOL”. Es preferible ponerle a nuestro ligando un nombre que lo distinga. Para esto, usamos el comando *sed*, el cual reemplaza un patrón de caracteres por otro:

```
sed s'/MOL/31N/' 31N.mol2 > 31N_2.mol2
```

- El resultado de este comando es un archivo *31N_2.mol2* que dice “31N” donde antes decía “MOL”. Revise este archivo con un editor de texto y arréglole. ¿Qué toca arreglar? ¿El patrón “MOL” se reemplazó en algún lugar indebido?
- Cambie el archivo de nombre a “31N.mol2” con el comando *mv* (i.e. *mv 31N_2.mol2 31N.mol2*)
- Ahora debemos calcular los demás parámetros necesarios para el campo de fuerza con el comando *parmchk*.

```
parmchk -i 31N.mol2 -f mol2 -o 31N.frcmod -a y
```

- El resultado del comando anterior es un archivo *31N.frcmod* que muestra todos los parámetros que puede asignar. Si no logra asignar algún parámetro lo va a poner en 0,000 y al lado derecho dirá “ATTN: Needs Revision”.
- Ya que calculamos los parámetros para el ligando, vamos a usar *tleap* para crear un archivo *.lib* con la declaración de los *atom types* presentes. Para esto vaya al repositorio del curso y descargue los archivos “*leapLig.in*” y “*leapComplex.in*”.
- Los archivos *.in* contienen la información necesaria para que *tleap* haga lo que queremos que haga. Entienda cada línea del archivo *leapLig.in* y ejecútelo de la siguiente manera:

```
tleap -sf leapLig.in
```

- El archivo *leapLig.in* lo que hace es importar los parámetros y asignarlos a una geometría. El resultado de ese cálculo es un archivo *leap.log* que tiene la información de lo que hizo *tleap* y otros 3 archivos: *31N.inpcrd* que tiene las coordenadas del ligando en formato AMBER (el programa, no la empresa), *31N.prmtop* que tiene la conectividad del ligando en formato AMBER y *31N.lib* que tiene la declaración de los *atom types* presentes. Note que si alguno de sus archivos se llama diferente a como se ha establecido en el taller, deberá modificar los archivos *leapLig.in* y *leapComplex.in*.
- Ahora debemos unir los parámetros del receptor y del ligando. Para esto vamos a usar otra vez *tleap*, esta vez con el archivo *leapComplex.in*.
- Además necesita descargar del repositorio del taller este archivo *complex.pdb*.
- El archivo *complex.pdb* es el mismo archivo *4plb_wild_fixed.pdb* pero alineando el compuesto de *31N.mol2* con el ligando nativo y eliminando el ligando nativo. Esto se puede hacer con la herramienta *Superposition* de Maestro. Esto es necesario ya que en los pasos anteriores generamos *atom types* a átomos con nombres específicos que salen del archivo *31N.mol2*. Si se usara *4plb_wild_fixed.pdb* se obtendría un error diciendo que no se encontraron *atom types* para todos los átomos. Por tiempo, no vamos a generar el archivo *complex.pdb* en el taller.
- Además del ligando, existe otro átomo para el cual no se tienen parámetros en el campo de fuerza predeterminado de AMBER99SB-ILDN: el magnesio que tiene la ADN girasa como cofactor. Estos parámetros se pueden encontrar en internet. Una copia de los archivos con los parámetros se encuentran en el repositorio del taller: “MG.frcmod” y “MG.prep”. Descarguelos, como todo lo demás, en su carpeta de trabajo.
- Ahora puede abrir el archivo *leapComplex.in* y entender qué hace cada línea.
- Corra el archivo similar a como se hizo la vez pasada:

```
tleap -sf leapComplex.in
```

- Este cálculo debe arrojar dos archivos: *sys.inpcrd* y *sys.prmtop*. El primero tiene las coordenadas del sistema en formato AMBER y el segundo tiene la topología (conectividad y parámetros) de todos los átomos también en formato de AMBER.
- Ya casi estamos listos para empezar la simulación, solo falta convertir los archivos de coordenadas y topología a un formato reconocible por Gromacs. Para esto se usa un *script* disponible en internet llamado *acpype* el cual puede descargar del repositorio del taller. Cópelo a la carpeta de trabajo. Para hacer la conversión solo se debe correr el siguiente comando:

```
python acpype.py -p sys.prmtop -x sys.inpcrd -r
```

- El resultado del cálculo anterior son dos archivos con la misma información anterior pero en formato de Gromacs: *sys_GMX.gro* (coordenadas) y *sys_GMX.top* (topología).

3. Creación de la caja

Una vez con la topología y las coordenadas se puede proceder a definir la región del espacio que se simulará. Note que entre más grande sea esta región más se le va a exigir a los procesadores durante la simulación pues habrán más moléculas de disolvente.

- Primero debemos conectarnos a la máquina en la cual trabajaremos. Para esto use el comando *ssh* y el IP.

```
ssh guest@157.253.72.13
```

La contraseña a esta cuenta es “12345”

- Nuevamente con el comando *mkdir* cree una carpeta con su nombre si todavía no existe.
- Ahora vamos a copiar los 2 archivos finales que generamos en la sección anterior. Para esto podemos usar el comando *scp*:

```
scp estudiantes@157.253.60.12:~/SuNombre/sys_GMX.gro .
```

Use el mismo comando para trasladar el archivo *sys_GMX.top*

- Para definir la región en el espacio a simular (la caja), se usa el siguiente comando:

```
gmx editconf -f sys_GMX.gro -o sys_BOX.gro -d 1.5
```

- El prefijo *gmx* es la palabra clave para que el computador entienda que estamos llamando a Gromacs. La segunda palabra hace referencia al comando de Gromacs que se desea usar. Después del comando se especifican las opciones del comando.
- La opción *-f* establece el archivo de coordenadas con el que se trabajará. La opción *-o* especifica el nombre del archivo con la información de la caja. La opción *-d* forza a los átomos del sistema a encontrarse a al menos $1.5nm$ (15\AA) de la caja. Por defecto se usa una caja ortorómica (ángulos iguales, lados distintos) pero se puede especificar otro tipo de caja.
- Al final del archivo *.gro* que se generó se deberían encontrar 3 números correspondientes a las dimensiones de la caja.

4. Adición de agua

Ahora podemos agregar moléculas de agua al espacio en el que no se encuentra la proteína. Las moléculas de agua se agregan dependiendo del modelo de solvatación que se desee usar.

- Descargar del repositorio del taller los archivos *tip3p.itp* y *ions.itp*.
- Ejecutar el siguiente comando para solvatar el sistema:

```
gmx solvate -cp sys_BOX.gro -cs spc216.gro -p sys_GMX.top -o sys_SOLV.gro
```

- La opción *-cp* especifica el archivo de coordenadas a usar (el que se generó en el paso anterior). La opción *-cs* especifica el modelo de solvatación a utilizar, “*spc216.gro*” significa el modelo *TIP3P*. La opción *-p* especifica el archivo de topología del sistema y *-o* el nombre del archivo de salida. El archivo de coordenadas de salida ahora debería tener las coordenadas de todas las moléculas de agua que se agregaron además de la información que tenía anteriormente.
- Note que al haber agregado átomos (oxígenos e hidrógenos), tenemos una topología incompatible con el archivo de coordenadas pues la topología no tiene parámetros para estos nuevos átomos. Para hacerlos compatibles debemos abrir el archivo de topología con un editor de texto (i.e. *vi sys_GMX.top*) y agregar los parámetros pertinentes. Dos modificaciones deben ser hechas para esto:
 1. Al comienzo del archivo hay una sección llamada *[atomtypes]*. Al final de la sección se deben agregar las cargas y parámetros de van der Waals para los nuevos *atom types* (OW y HW). Agregue las siguientes líneas:

```
OW OW 16.0000 -0.8476 A 3.15061e-01 6.36272e-01
HW HW 1.0000 0.4238 A 0.00000e-01 0.00000e-01
```

2. Al final del archivo hay una sección llamada *[system]*. Antes de que esta sección inicie se debe agregar una línea que llame al archivo con los parámetros de nuestro modelo de solvatación. Escriba entonces la siguiente línea:

```
#include “./tip3p.itp”
```

5. Adición de iones

Aunque el sistema ya está solvatado, aún debemos agregar iones para simular correctamente las condiciones fisiológicas y neutralizar el sistema, el cual, como ya debe saber en este momento, tiene una carga de casi $-60e$. Es importante mencionar que a partir de este momento son necesarios dos pasos para hacer prácticamente cualquier cosa: Unir las coordenadas, la topología y las especificaciones en un único archivo, y luego ejecutar ese archivo. Por esta razón debe proceder descargando todos los archivos con extensión *.mdp* de la página del curso.

- Primero uniremos las coordenadas, topología y parámetros del cálculo con el comando *grompp*:

```
gmx grompp -f minim.mdp -c sys_SOLV.gro -p sys_GMX.top -o sys_ION.tpr
```

- El archivo *.mdp* es un archivo de texto que contiene todas las especificaciones técnicas de lo que se va a hacer, en este caso, como solo se van a agregar iones, el archivo no es importante pero se le debe dar. Ese archivo se puede descargar del repositorio del taller. Al igual que los demás comandos de Gromacs, éste requiere de un archivo de coordenadas y de topología.
- El resultado de este cálculo es un archivo *.tpr* fácil de manipular pero que no se puede abrir como un archivo de texto (es un binario).

- Para agregar los átomos solo hace falta ejecutar el archivo que acabamos de generar. Para esto se debe correr el siguiente comando:

```
gmx genion -s sys_ION.tpr -o sys_ION.gro -p sys_GMX.top -pname So -neutral
```

- Después de la ejecución preguntará qué moléculas se desean reemplazar por iones. Seleccione el solvente.
- El resultado de este cálculo es un archivo *.gro* en el que algunas moléculas de agua fueron reemplazadas por átomos de tipo “So”. “So” quiere decir sodio, la razón por la que no se usa “Na” es porque puede haber un *atom type* (algún nitrógeno) que ya tenga este nombre (puede ocurrir lo mismo con “So” pero es menos probable, de todas maneras siempre es mejor asegurarse).
- Al igual que en la sección anterior, ahora es necesario agregar los parámetros de los átomos que se agregaron al sistema, nuevamente se deben hacer dos cosas:

1. En la sección de *[atomtypes]* agregar la siguiente línea.

```
So So 23.0000 1.0000 A 3.32840e-01 1.15897e-02
```

2. Antes de la sección *[system]* agregar la siguiente línea.

```
#include “./ions.itp”
```

6. Minimización

Ahora sí tenemos el sistema que deseamos simular. Lo primero que debemos hacer es una minimización, para asegurarnos de que la simulación no empiece en un punto energéticamente desfavorable.

- Nuevamente usamos *grompp* para pegar los archivos pertinentes de la siguiente manera:

```
gmx grompp -f minim-solvated.mdp -c sys_ION.gro -p sys_GMX.top -o sys_MIN.tpr
```

- Esta vez el archivo *.mdp* sí es importante pues nos especifica aspectos como el número máximo de iteraciones, el método con el que se minimizará el sistema, etc.
- Ahora usamos el archivo *.tpr* que acabamos de generar para minimizar el sistema:

```
gmx mdrun -v -gpu_id 0 -deffnm sys_MIN
```

- La herramienta *mdrun* es la encargada de hacer minimizaciones, calentamientos, equilibraciones y producciones. La opción *gpu_id* especifica la *gpu* en la que se correrá el cálculo. Para asegurarse de que nadie más esté usando esa *gpu* puede correr el comando *nvidia-smi* y ver el porcentaje de *gpu* usado actualmente. Si está en uso, puede cambiar el número 0 por 1, 2 o 3 (la máquina de trabajo solo tiene 4 *gpu*’s). La opción *-deffnm* busca un archivo *.tpr* con el nombre que se dió a continuación y todos los archivos de *output* se escriben con ese nombre pero con diferentes extensiones.
- El resultado de este cálculo es un archivo *.trr* que es un formato de trayectoria, tiene algunas de las geometrías que se evaluaron durante la minimización (es un binario). También se obtiene un archivo *.gro* que tiene el último marco de la minimización, es decir, el sistema ya minimizado. Se obtiene un archivo *.log* con la información de lo que hizo Gromacs durante el cálculo y un archivo *.edr* que tiene información como energías y propiedades termodinámicas durante la minimización.
- Si se desea extraer información sobre la energía (o cualquier otra propiedad), se debe correr el siguiente comando:

```
gmx energy -s sys_MIN.tpr -f sys_MIN.edr -o sys_MIN.xvg
```

Este comando necesita de un binario *.tpr* y un archivo de energías *.edr*. El resultado del cálculo es un archivo de texto de tipo *.xvg* donde cada columna corresponde a una de las propiedades que se le pidió al comando imprimir tras su ejecución. Estos datos se pueden graficar con programas como *xmgrace* y *gnuplot*, o se puede escribir un *script* que lo haga en el lenguaje de programación de su elección.

- Se recomienda imprimir y graficar la energía potencial del sistema para verificar que efectivamente la energía convergió a un mínimo.

7. Generación de restricciones

Una buena práctica durante el calentamiento y la equilibración es fijar en el espacio algunos átomos. Esto lo hacemos para no permitir que el sistema cambie demasiado durante estas etapas y poder extraer información útil durante la producción. La idea detrás de una restricción es que se le debe decir a Gromacs que un cierto conjunto de átomos les aplique una constante de fuerza considerablemente alta en todas las direcciones, para que no se muevan demasiado.

- Primero creamos un grupo de átomos. Para esto usamos el siguiente comando:

```
gmx make_ndx -f sys_MIN.gro -o index.ndx
```

- Al ejecutar el comando, se va a solicitar al usuario que defina el grupo de interés. Suponiendo que el “backbone” y el “DNA” corresponden a los grupos “4” y “11”, respectivamente, para crear un grupo con ambos conjuntos de átomos se necesita indicar “4 | 11”. Cree un grupo que contenga el “backbone”, “DNA” y “31N”. Cree otro grupo con todo nuestro sistema de interés: “Protein”, “DNA”, “31N” y “MG” y otro con la parte del sistema que no es de nuestro interés: “Water” y “So”. Asegúrese que el número de átomos en los últimos dos grupos sumen los átomos de todo el sistema.
- El resultado de este comando es un archivo de índices de tipo *.ndx*. Este archivo es de texto y está dividido en diferentes secciones. Cada sección corresponde a un grupo de átomos, entre ellos, los que definimos.
- Ahora, para crear las restricciones sobre el primer grupo que definimos se usa el siguiente comando:

```
gmx genrestr -f sys_MIN.gro -n index.ndx -o restraints.itp
```

- Al ejecutar este comando se va a solicitar al usuario que especifique el grupo al que se le desean generar las restricciones. Seleccione el primero que definimos.
- El resultado de este cálculo es un archivo de texto *.itp* que contiene una lista con los átomos a restringir y tres columnas indicando las constantes de fuerza en cada dirección para cada átomo. Note que estas son restricciones flexibles pues no se fija la posición de los átomos, solo se les pone una fuerza muy grande en todas las direcciones.
- Para hacer vigentes las restricciones que acabamos de crear es necesario agregarlas a la topología. Para esto, abra con un editor de texto el archivo de topología y agregue la siguiente línea al final del archivo justo antes de las 2 líneas que ya habíamos escrito:

```
#include "restraints.itp"
```

8. Calentamiento

Ya que tenemos el sistema minimizado y parte de él restringido, podemos empezar a calentar el sistema.

- Descargue del repositorio del curso el archivo *nvt.mdp*, el cual contienen las especificaciones técnicas del calentamiento (termostato a usar, tiempo de calentamiento, constante de acople con el termostato, etc.).
- Para asegurarnos de que el calentamiento ocurra lo más uniformemente posible y que la energía que se escriba sea representativa de nuestro sistema de interés, debemos dividir nuestro sistema en dos: interés y no-interés.
- Abra con un editor de texto el archivo *nvt.mdp* y busque las opciones *energygrps* y *tc-grps*. Cambie los nombres que están después de estas palabras claves a los nombres de los 2 últimos grupos que se definieron en la primera parte de la sección 7. Deberían ser nombres parecidos a “Protein_DNA_31N” y “SOL_So”.
- Ahora unimos todos los archivos pertinentes para hacer el calentamiento:

```
gmx grompp -f nvt.mdp -c sys_MIN.gro -p sys_GMX.top -n index.ndx -o sys_NVT.tpr
```

- Al final de este cálculo se debe obtener un único archivo *.tpr*.
- Solo falta ejecutar el archivo que acabamos de generar:

```
gmx mdrun -v -gpu_id 0 -deffnm sys_NVT
```

- El resultado de este cálculo, al igual que en la minimización son archivos *.trr*, *.gro*, *.edr* y *.log*. En este caso, el primero de los archivos tiene la trayectoria de los átomos durante el calentamiento y el segundo archivo tiene la geometría del último paso del calentamiento. Además, esta vez se debió escribir un archivo de tipo *.cpt*, el cual tiene la misma función que los *checkpoints* de Gaussian. Almacena toda la información del cálculo periódicamente para que, si cae por alguna razón, se pueda retomar el cálculo donde se había quedado.

9. Equilibración

Ya que el sistema está caliente, podríamos empezar a tomar datos a partir de este momento, pero debido a que el sistema puede seguir buscando una conformación estable a esa temperatura, es necesario darle un poco de tiempo para que se equilibre.

- Descargue del repositorio del taller el archivo *npt.mdp* y hágale las mismas modificaciones que se le hizo a *nvt.mdp* en la sección 8.
- Nuevamente empezamos por unir todos los archivos pertinentes:

```
gmx grompp -f npt.mdp -c sys_NVT.gro -t sys_NVT.cpt -p sys_GMX.top -n index.ndx -o sys_NPT.tpr
```

- Note que ahora se agregó una opción nueva (*-t*) al comando. Esta opción llama al archivo *.cpt*. Esto se debe hacer porque este archivo tiene el último cálculo de velocidades y fuerzas del calentamiento, lo cual se usará en el primer paso de la equilibración.
- Procedemos a correrlo:

```
gmx mdrun -v -gpu_id 0 -deffnm sys_NPT
```

- Los archivos que se deben obtener tras este cálculo deben ser los mismos formatos a los que se obtuvieron en la sección 8.

10. Producción

Al fin vamos a llevar a cabo la producción de la dinámica molecular, de donde realmente sacaremos la información de nuestro sistema. Lo primero que se debe hacer es QUITAR LAS RESTRICCIONES!!! Para esto elimine la línea que las incluía que se escribió en la topología.

- Descargue del repositorio del taller el archivo *production.mdp* y haga las mismas modificaciones que se les hizo a los archivos *.mdp* en las secciones anteriores.
- Unimos todos los archivos de interés:

```
gmx grompp -f production.mdp -c sys_NPT.gro -t sys_NPT.cpt -p sys_GMX.top -n index.ndx -o sys_MD.tpr
```

- Ya que este es el paso más demorado, no tiene sentido esperar a que termine frente al computador. En cambio, es preferible dejar el cálculo corriendo en el fondo de la máquina. Para esto se usa el siguiente comando:

```
nohup gmx mdrun -v -gpu_id 0 -deffnm sys_MD &
```

- El símbolo *et* (&) al final del comando hace que se corra en el fondo de la máquina. El comando *nohup* hace que todo lo que debió decirnos el comando en la terminal, se escriba en un archivo (*nohup.out*).
- Los resultados de esta simulación son los siguientes archivos: *.xtc*, un archivo con la trayectoria de la simulación, equivalente al *.trr* de las secciones pasadas. *.gro* un archivo con las coordenadas del último marco de la simulación. *.edr*, *.log* y *.cpt*, archivos con la información de la energía, avances de Gromacs y *checkpoints*, respectivamente.

11. Visualización

Una vez terminamos la producción, estamos listos para observar la trayectoria de la producción y analizar lo que más se pueda de ella. Desafortunadamente, por la forma en la que opera Gromacs, los archivos de trayectorias no muestran lo que queremos ver, se ve muy mal debido a la forma en la que trata las condiciones de frontera periódicas. Antes de la visualización debemos arreglar los archivos de trayectoria.

- Supongamos que se va a ver la trayectoria de la producción (aunque esto aplica para cualquier trayectoria en cualquier formato de Gromacs (*.xtc* o *.trr*)):

```
gmx trjconv -f sys_MD.xtc -o sys_MD_FIX.xtc -s sys_MD.tpr -pbc mol -ur compact
```

- Al ejecutar el comando, la terminal va a solicitar al usuario el grupo cuya trayectoria se desea arreglar. Lo mejor es seleccionar todo el sistema (grupo 0).
- El resultado del cálculo anterior es un único archivo llamado *sys_MD_FIX.xtc* que tiene la misma información que la trayectoria original pero ahora sí se puede analizar más fácilmente.
- Para abrir una trayectoria debe copiar la carpeta en la que se trabajó a un computador que tenga instalado *VMD* y que tenga suficiente memoria RAM como para abrir las trayectorias (cuánta depende de la extensión de la trayectoria).
- Abra *VMD*.
- Abra la última estructura (*.gro*) del paso anterior a la dinámica que se desea estudiar. Por ejemplo, si desea ver la producción (*sys_MD_FIX.xtc*), entonces debe abrir el archivo *sys_NPT.gro*, el cual tiene la geometría del último paso de la equilibración. Para abrir esta geometría:

File → *New Molecule...* → *Browse* → **Seleccione el archivo .gro** → *Load*

- Ya debería poder ver todo el sistema en la interfaz gráfica de VMD.
- Para abrir la trayectoria:

*File → New Molecule... → Load files for → *Nombre del archivo .gro que se acaba de importar* →
Filename → *Selecione el archivo .xtc* → Load*

- Debería poder ver cómo se carga, un marco a la vez, la trayectoria de interés.
- Usted debería poder ver a la proteína que se simuló dentro de un cubo de agua. Para visualizar lo que deseemos es necesario seguir la siguiente ruta:

Graphics → Representations...

- En el cuadro de texto de la ventana que se acaba de abrir, escribimos “protein”. Si también queremos ver el ADN podemos hacer click en “Create Rep” y escribir “nucleic”. Similarmente podemos visualizar el ligando escribiendo “resname 31N”.
- Abajo del cuadro de texto hay tres casillas llamadas “Coloring Method”, “Material” y “Drawing Method”. En estas casillas podemos escoger el color de lo que estamos mostrando, su textura y su forma de representación. Usted es libre de jugar con estas opciones para llegar a una vista con la que se sienta cómodo.
- Volviendo a la ventana principal de VMD, si le damos *play* a la simulación (abajo a la derecha de la ventana) podemos notar que la proteína se empieza a desplazar del centro de la pantalla. Esto puede dificultar su visualización pero solo se debe a los ajustes predeterminados de VMD. Para arreglar esto, desde VMD, siga el siguiente procedimiento:

Extensions → Analysis → RMSD Trajectory Tool

- Desde esta ventana se puede alinear una selección en todos los marcos de la simulación. Por ejemplo, para que los carbonos alfa ($C\alpha$) de la proteína se alineen debemos escribir en el cuadro de texto “alpha”. Si quisiéramos alinear el ADN sería necesario escribir “nucleic”, y si quisiéramos alinear el ligando sería necesario escribir “resname 31N”.
- Para hacer el alineamiento escriba lo que desea alinear en el cuadro de texto, asegúrese que la opción “Top” esté seleccionada en el recuadro llamado “Reference mol” y haga click en “Align”. Asegúrese de que solo haya un sistema en VMD, si hay mas de un archivo abierto puede que este paso no funcione.

12. Análisis de trayectorias

12.1. RMSD

El RMSD (*Root Mean Square Deviation*) es una de las métricas más comunes para estudiar trayectorias de simulaciones MD. Esta métrica compara dos geometrías y da cuentas de qué tanto se diferencian las posiciones de los átomos en promedio. Note que un aumento en RMSD puede implicar dos cosas: 1) Que todos los átomos se encuentren en posiciones ligeramente diferentes a las de la geometría de referencia, o 2) que pocos átomos se hayan desplazado mucho con respecto a la referencia.

El RMSD es común calcularlo entre cada uno de los marcos de la trayectoria y la geometría inicial, de esta manera se puede hacer una gráfica de RMSD vs tiempo. Esta gráfica es una forma preliminar de determinar si el sistema se equilibró o no. El RMSD se puede calcular con VMD pero, por consistencia, lo calcularemos con Gromacs. Para esto debe ir a su carpeta de trabajo, es decir, en la que corrió la dinámica y en la que se encuentra la trayectoria *sys_MD_FIX.xtc*.

- Para calcular el RMSD de cada marco con respecto al primer marco de la producción:

```
gmx rms -s sys_MD.tpr -f sys_MD_FIX.xtc -o RMSD.txt
```

- Al ejecutar este comando se va a solicitar al usuario que seleccione el grupo que se quiere alinear, el cual por lo general coincide con el grupo cuyo RMSD se quiere calcular. En este ejemplo vamos a alinear los carbonos alfa del sistema. Escriba el número del grupo y ponga Enter. A continuación Gromacs pide al usuario especificar el grupo para el cálculo del RMSD, seleccione el mismo grupo y ponga Enter.
- El resultado de este comando es un archivo llamado “RMSD.txt” que si lo abre con un editor de texto como *vi* podrá notar que tiene unos renglones de encabezados y luego dos columnas, una con el tiempo de la simulación y otra con el RMSD. Recuerde las unidades de Gromacs: la columna de tiempo está en picosegundos y la de RMSD en nanómetros.

12.2. RMSF

El RMSF (*Root Mean Square Fluctuation*) es una medida de qué tanto se mueve en promedio un grupo de átomos a lo largo de una simulación y es una métrica complementaria al RMSD. Mientras el RMSD nos da información sobre cuándo ocurre un cambio drástico en una geometría, el RMSF da información sobre dónde se dan los cambios más drásticos en el sistema. Uno puede calcular el RMSF para cualquier grupo de átomos, por ejemplo, si se calculara para todos los átomos que conforman un residuo se obtiene el RMSF del residuo, este se puede relacionar a la plasticidad del sistema.

A diferencia del RMSD, el RMSF no se puede calcular con VMD a menos de que se tenga una simulación muy corta o un sistema muy pequeño, por esto debemos calcularlo también con Gromacs.

Nuestra meta en esta sección es calcular el RMSF de los residuos que tengan un átomo a 5Å o menos de nuestro ligando. Como siempre lo primero que debemos hacer es un grupo con los residuos de interés que entienda Gromacs.

- Vaya a VMD y abra la simulación como se explicó en la sección 11.
- Abra la ventana de representaciones como se explicó en la sección 11.
- Asegúrese de que haya una única representación y escriba “(same residue as protein within 5 of resname 31N) or (same residue as nucleic within 5 of resname 31N) or resname 31N”. Esta selección debe mostrar al ligando y los residuos con un átomo a 5Å de él.
- Los residuos que se muestran son aquellos cuyo RMSF queremos calcular.
- Ahora debemos ir a la ventana principal de VMD y seleccionar:

File → Save Coordinates...

- En la sección “Save data from” seleccionar la simulación que se cargó. En la sección “Selected atoms” seleccionar el texto que se escribió hace 3 puntos. En la sección “Last” poner “0” y haga click en “Save”. Sálvela bajo el nombre “bsite.pdb”.
- El archivo *.pdb* que generamos tiene los números de los residuos que queremos darle a Gromacs para que calcule el RMSF.
- Ahora debemos extraer los números de los residuos y escribirlos en un formato que Gromacs entienda. Para esto descargue del repositorio del curso el script “bindsiteselec.py” y póngalo en la misma carpeta en la que está el archivo *.pdb* recién generado. Ejecute el script de la siguiente manera:

```
python bindsiteselec.py bsite.pdb
```

- Este comando no va a escribir ningún archivo pero va a mostrar un texto en la terminal que nos va a ser útil más adelante.

- Vaya a la carpeta en la que está trabajando con Gromacs y ejecute el siguiente comando:

```
gmx make_ndx -f sys_MD.gro -o rmsf_index.ndx
```

- Cuando se ejecute el comando, se va a solicitar al usuario que defina los grupos que desee. Copie y pegue el texto obtenido con el script “bindsiteelec.py” y ponga Enter. Esto debió generar un nuevo grupo con los residuos que seleccionamos en VMD.
- Ahora que tenemos un archivo de índices (.ndx) con los números de los átomos de interés podemos proceder a calcular su RMSF con el siguiente comando:

```
gmx rmsf -f sys_MD_FIX.xtc -s sys_MD.tpr -n rmsf_index.ndx
```

- Al ejecutar el comando se debe seleccionar el grupo cuyo RMSF se desea calcular. Escriba el nombre del grupo que acabamos de definir y ponga Enter.
- El resultado de este comando es un archivo de texto llamado “rmsf.xvg” que tiene varios renglones de encabezado seguidos de dos columnas. En la primera columna está el número del residuo y en la segunda el RMSF (en nanómetros).
- Nota: Si usted quisiera ver cuál residuo es, por ejemplo, el 1379, puede hacer una representación en VMD y escribir “resid 1379”.

12.3. Ocupación de puentes de hidrógeno

Unas de las interacciones más importantes al momento de estudiar la inhibición de las proteínas son los puentes de hidrógeno entre el receptor y el ligando. Información que nos puede ser muy útil para explicar la potencia de un inhibidor es la cantidad de puentes H y el porcentaje del tiempo de simulación que estos se encuentran. Aunque los puentes H sean interacciones cuánticas, cuando usamos métodos de mecánica molecular la presencia de un puente H se puede determinar a partir de distancias y ángulos entre el aceptor, el donador y el protón involucrados.

Nuestro objetivo en esta sección es determinar qué puentes H hay entre el ligando y el receptor, y por cuánto tiempo se encuentran presentes. Esto lo haremos con Gromacs y un script de acceso libre disponible en línea.

- Vamos a la carpeta en la que estamos utilizando Gromacs y usamos el siguiente comando:

```
gmx hbond -s sys_MD.tpr -num numhbond -hbm matrixhb -hbn indexhb
```

- Al ejecutar este comando se le pide al usuario seleccionar los dos grupos cuyos puentes H desean ser evaluados. En este ejemplo vamos a poner el número del grupo “Protein” (y damos Enter) y el número del grupo que tiene al ligando “31N” (y volvemos a dar Enter).
- El resultado de este cálculo son 3 archivos. El primero de ellos es un archivo de índices que se llama “indexhb.ndx” que al final tiene una lista de grupos con los átomos donadores, aceptores e hidrógenos tanto de la proteína como del ligando. El segundo archivo es un archivo de texto llamado “numhbond.xvg” que tiene 3 columnas: Tiempo de la simulación, número de puentes H presentes en ese momento y número de parejas aceptor-donador que están a una distancia menor a 0.35nm (estos no son considerados puentes H porque la distancia no es el único criterio). El tercero es un archivo de texto llamado “matrixhb.xpm” que aunque se puede abrir con un editor, no nos proporciona información.
- Para poder obtener el factor de ocupación de los puentes de hidrógeno usamos un script llamado “readHBmap.py” que se puede descargar gratuitamente desde la página de Gromacs o desde el repositorio del taller. Descárguelo y póngalo en la misma carpeta que los archivos que se generaron en el punto anterior.

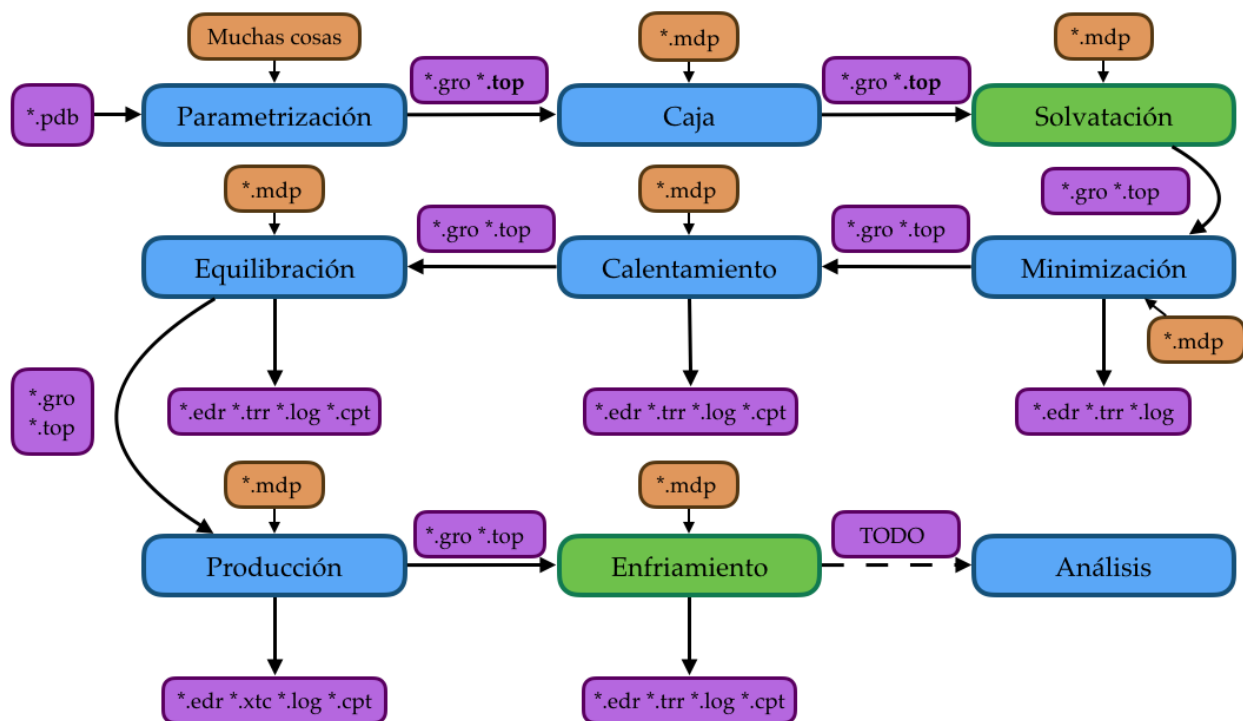


Figura 1: Orden lógico de una simulación de dinámica molecular.

- Ejecute el siguiente comando:

```
python readHBmap.py -hbm matrixhb.xpm -hbn indexhb.ndx -f sys_MD.gro -o hbmap.xvg
```

- El resultado de este comando son dos archivos. El primero es un archivo de texto llamado “pairs.dat” que contiene una lista de todos los puentes H entre la proteína y el ligando y su respectivo factor de ocupación. El segundo archivo también es de texto y se llama “hbmap.xvg”. Este archivo tiene una columna con el número del marco de la simulación y una columna adicional por cada puente H entre la proteína y el ligando. En este archivo también hay factores de ocupación pero no calculados para la simulación total (como en “pairs.dat”) sino que se calcula por fragmentos (cada 100 marcos).