

md-sim-gromacs-ntb

March 25, 2023

Contenido

0 Preliminares

0.1 Introducción a las simulaciones de Dinámica Molecular

0.2 Librerías para manejo del notebook

0.3 Descarga del archivo PDB

0.4 Limpieza de la Proteína

1 Generación de la topología

2 Generación una caja de simulación y adición de solvente

3 Adición de iones y creación de un sistema de carga neutra

4 Minimización de energía

4.1 Archivo de configuración de la simulación (minim.mdp)

5 Equilibramiento

5.1 Simulación de Equilibración NVT

5.2 Simulación de Equilibración NPT

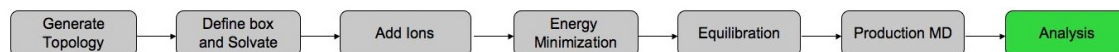
6 Simulación de Producción

7 Análisis de la Simulación

7.1 Preprocesamiento de la trayectoria

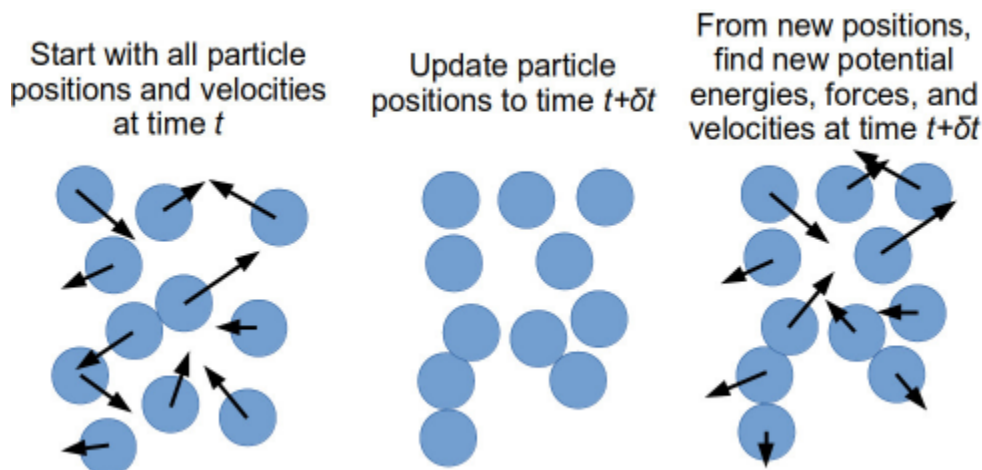
7.2 Desviación cuadrática media raíz (RMSD)

1 Simulaciones de dinámica molecular con GROMACS



1.1 Preliminares

1.1.1 Introducción a las simulaciones de Dinámica Molecular



1.1.2 Librerías para manejo del notebook

```
[47]: # Librerías de Python
import nglview          # For visualizing 3D structures
import ipywidgets       # For organizing 3D structures in panels
from IPython.display import Image
```

1.1.3 Descarga del archivo PDB

```
[48]: ## Descarga del archivo PDB
!wget https://files.rcsb.org/download/5pep.pdb &> log00a.log
!cp 5pep.pdb proteina.pdb
```

```
--2023-03-22 11:36:57-- https://files.rcsb.org/download/5pep.pdbn
Resolving files.rcsb.org (files.rcsb.org)... 128.6.158.70
Connecting to files.rcsb.org (files.rcsb.org)|128.6.158.70|:443... connected.
HTTP request sent, awaiting response... 404 Not Found
2023-03-22 11:36:58 ERROR 404: Not Found.
```

```
[49]: ## Visualización resultados
!ls -tl|head -3
view = nglview.show_structure_file ("proteina.pdb")
view.add_representation (repr_type="ball+stick", selection="HOH")
view
```

```
total 84252
-rw-rw-r-- 1 lg lg 263250 mar 22 11:36 proteina.pdb
-rw-rw-r-- 1 lg lg 776624 mar 22 11:35 md-sim-gromacs-ntb.html
NGLWidget()
```

1.1.4 Limpieza de la Proteína

```
[50]: ## Limpieza de la Proteína
!grep -v 'HOH' proteina.pdb > proteina_limpia.pdb
```

```
[51]: ## Visualización resultados
!ls -tl|head -3
view = nglview.show_structure_file ("proteina_limpia.pdb")
view.add_representation (repr_type="ball+stick", selection="HOH")
view
```

```
total 84252
-rw-rw-r-- 1 lg lg    232794 mar 22 11:36 proteina_limpia.pdb
-rw-rw-r-- 1 lg lg    263250 mar 22 11:36 proteina.pdb

NGLWidget()
```

1.2 Generación de la topología

```
[52]: ## Creación de una topología Gromacs (PDB2GMX)
!{ echo 1;echo 1; }|gmxd b2gmxd -f proteina_limpia.pdb -o proteina_gromacs &> log01.log
```

```
[53]: ## Visualización resultados
!ls -tl|head -5
view = nglview.show_structure_file ("proteina_gromacs.gro")
view.add_representation (repr_type="ball+stick", selection="HOH")
view
```

```
total 85636
-rw-rw-r-- 1 lg lg      8142 mar 22 11:37 log01.log
-rw-rw-r-- 1 lg lg    210794 mar 22 11:37 proteina_gromacs.gro
-rw-rw-r-- 1 lg lg   1290677 mar 22 11:37 topol.top
-rw-rw-r-- 1 lg lg     75479 mar 22 11:37 posre.itp

NGLWidget()
```

1.3 Generación una caja de simulación y adición de solvente

```
[54]: ## Generación una caja de simulación
!gmxd editconf -f conf.gro -c -d 1 -bt cubic -o proteina_caja.gro &> log02a.log
```

```
[55]: ## Visualización resultados
!ls -tl|head -5
```

```
total 85640
-rw-rw-r-- 1 lg lg      2959 mar 22 11:37 log02a.log
-rw-rw-r-- 1 lg lg      8142 mar 22 11:37 log01.log
```

```
-rw-rw-r-- 1 lg lg    210794 mar 22 11:37 proteina_gromacs.gro
-rw-rw-r-- 1 lg lg   1290677 mar 22 11:37 topol.top
```

```
[56]: ## Adición de solvente al sistema
!gmx solvate -cp proteina_caja.gro -cs spc216.gro -o proteina_solvente.gro -p
↪topol.top &> log02b.log
```

```
[57]: ## Visualización resultados
!ls -tl|head -5
view = nglview.show_structure_file ("proteina_solvente.gro")
view.add_representation (repr_type="cartoon", color="green")
view.add_representation (repr_type="point", selection="SOL")
view
```

```
total 89648
-rw-rw-r-- 1 lg lg      4116 mar 22 11:37 log02b.log
-rw-rw-r-- 1 lg lg   1290708 mar 22 11:37 topol.top
-rw-rw-r-- 1 lg lg  2800169 mar 22 11:37 proteina_solvente.gro
-rw-rw-r-- 1 lg lg     2959 mar 22 11:37 log02a.log
```

NGLWidget()

1.4 Adición de iones y creación de un sistema de carga neutra

Agregar iones a su sistema solvatado puede servir para dos propósitos: puede ayudar a neutralizar cualquier carga en su sistema; y le permite simular sistemas con concentraciones de sal similares a sus equivalentes del mundo real. Es posible que haya notado anteriormente que nuestro sistema está cargado. Aquí lo neutralizaremos agregando iones.

La adición de iones se realiza en dos partes: primero, debe usar la herramienta grompp para generar un archivo .tpr que se usará al agregar iones, y luego debe reemplazar algunos de las moléculas solventes agregados recientemente con los contraiones necesarios usando genion.

La herramienta de preprocesador GROMACS grompp lee archivos de coordenadas y topología para generar un archivo de entrada de nivel atómico (con una extensión .tpr). Este archivo .tpr contiene todos los parámetros necesarios para todos los átomos del sistema. Para generar un archivo .tpr de entrada de ejecución, grompp necesita un archivo de estructura (.gro), un archivo de topología (.top) y un archivo que define las instrucciones para la ejecución de la simulación (esto se guarda en un archivo .mdp). Este archivo .mdp se puede mantener vacío cuando se ioniza el sistema, ya que no se ejecutará ninguna simulación real.

```
[58]: # Crea un archivo .mdp vacío
!touch mdrun.mdp
!gmx grompp -f parametros_iones.mdp -c proteina_solvente.gro -p topol.top -o
↪parametros_iones.tpr &> log03a.log
```

Ahora que se ha generado el .tpr, se puede usar genion para neutralizar la carga del sistema. La carga del sistema disminuye reemplazando varias partes del sistema con aniones y cationes. Esto se hace ejecutando lo siguiente:

```
[59]: !echo 13 |gmj genion -s parametros_iones.tpr -p topol.top -neutral -o┐  
      ↪proteina_iones.gro &> log03b.log
```

```
[60]: ## Visualización resultados  
      !ls -tl|head -7  
      view = nglview.show_structure_file ("proteina_iones.gro")  
      view.add_representation (repr_type="cartoon", color="green")  
      #view.add_representation (repr_type="point", selection="SOL", color="gray")  
      view.add_representation (repr_type="point", selection="NA", color="yellow")  
      view
```

```
total 93656  
-rw-rw-r-- 1 lg lg      5787 mar 22 11:37 log03b.log  
-rw-rw-r-- 1 lg lg 2796758 mar 22 11:37 proteina_iones.gro  
-rw-rw-r-- 1 lg lg 1290724 mar 22 11:37 topol.top  
-rw-rw-r-- 1 lg lg      2999 mar 22 11:37 log03a.log  
-rw-rw-r-- 1 lg lg          0 mar 22 11:37 mdrun.mdp  
-rw-rw-r-- 1 lg lg      4116 mar 22 11:37 log02b.log  
  
NGLWidget()
```

1.5 Minimización de energía

1.5.1 Archivo de configuración de la simulación (minim.mdp)

```
[61]: !cat simulacion_minim.mdp
```

```
; MINIMIZATION RUN (minim.mdp)  
;(used as input into grompp to generate em.tpr)  
;  
; Parameters describing what to do, when to stop, and what to save  
integrator = steep          ; A steepest descent algorithm for energy  
minimization  
emtol      = 1000.0         ; Stop minimization when the maximum force < 1000.0  
kJ/mol/nm  
emstep     = 0.01          ; Minimization step size  
nsteps     = 500           ; 5000 Maximum number of (minimization) steps to  
perform  
  
;ELECTROSTATIC AND VDWAALS  
; Parameters describing how to find the neighbors of each atom and how to  
calculate the interactions  
nstlist    = 1             ; Frequency to update the neighbor list and long  
range forces  
cutoff-scheme = Verlet     ; Buffered neighbor searching  
ns_type     = grid         ; Method to determine neighbor list (simple, grid)  
coulombtype  = PME         ; Treatment of long range electrostatic interactions  
rcoulomb    = 1.0          ; Short-range electrostatic cut-off (nm)
```

```
rvdw          = 1.0          ; Short-range Van der Waals cut-off (nm)
pbc           = xyz          ; Periodic Boundary Conditions in all 3 dimensions
```

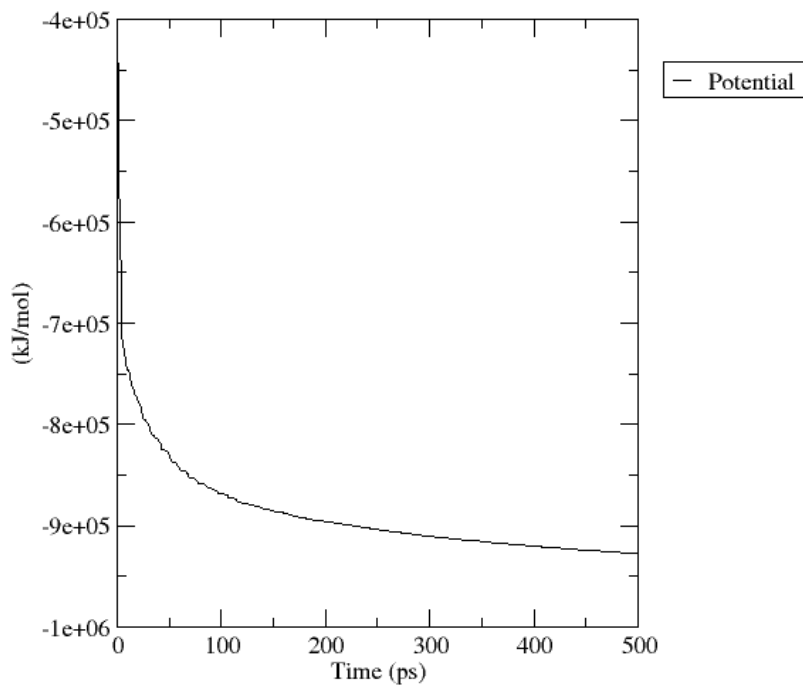
```
[62]: ## Preparación archivo de simulación
!gmx grompp -f simulacion_minim.mdp -c proteina_iones.gro -p topol.top -o
↪simulacion_minim.tpr &> log04a.log
```

```
[63]: ## Ejecución de la simulación
!gmx mdrun -v -deffnm simulacion_minim &> log04b.log
```

```
[64]: ## Analysis de la simulación: Energía potencial
!echo 10 | gmx energy -f simulacion_minim.edr -o energia_potencial.xvg &>
↪log04c.log
!grace -nxy energia_potencial.xvg -hdevice PNG -hardcopy -printfile
↪energia_potencial.png
Image(filename='energia_potencial.png', width = 400, height = 400)
```

[64]:

GROMACS Energies



1.6 Equilibramiento

1.6.1 Simulación de Equilibración NVT

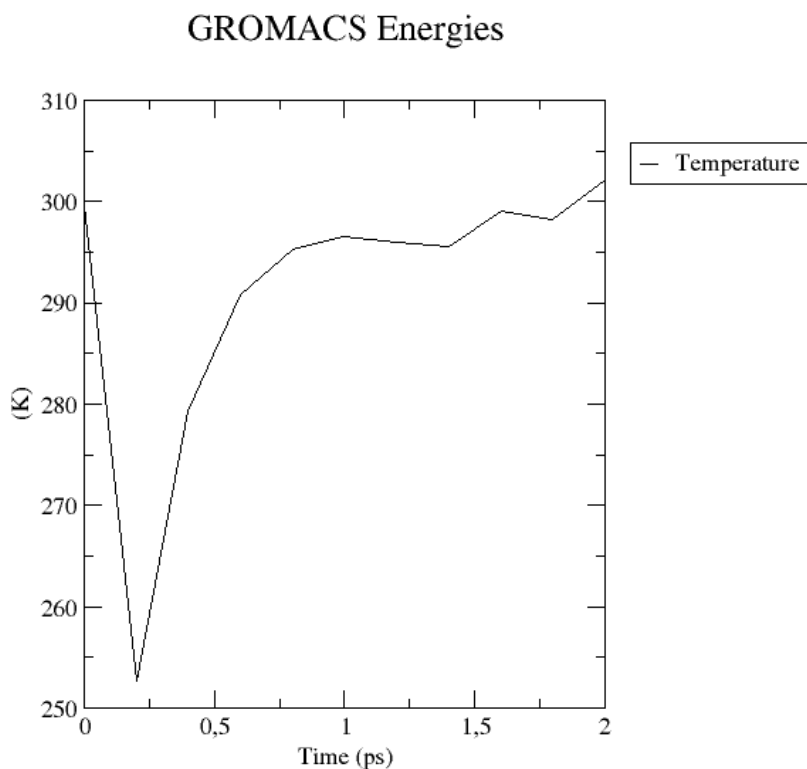
```
[65]: !gmx grompp -f simulacion_nvt.mdp -c simulacion_minim.gro -r simulacion_minim.  
      ↪gro -p topol.top -o simulacion_nvt.tpr &> log05a.log  
      !gmx mdrun -v -deffnm simulacion_nvt &> log05b.log
```

```
[66]: ## Resultados simulación NVT  
      !ls -lt|head -5  
      !echo 16 | gmx energy -f simulacion_nvt.edr -o temperatura.xvg &> log05c.log  
      !grace -nxy temperatura.xvg -hdevice PNG -hardcopy -printfile temperatura.png  
      Image(filename='temperatura.png', width = 400, height = 400)
```

total 124580

```
-rw-rw-r-- 1 lg lg      3689 mar 22 11:38 log05b.log  
-rw-rw-r-- 1 lg lg    28942 mar 22 11:38 simulacion_nvt.log  
-rw-rw-r-- 1 lg lg     3676 mar 22 11:38 simulacion_nvt.edr  
-rw-rw-r-- 1 lg lg  4288334 mar 22 11:38 simulacion_nvt.gro  
ls: write error: Broken pipe
```

[66]:



1.6.2 Simulación de Equilibración NPT

```
[67]: !gmx grompp -f simulacion_npt.mdp -c simulacion_nvt.gro -r simulacion_nvt.gro
      ↪ -t simulacion_nvt.cpt -p topol.top -o simulacion_npt.tpr &> log05d.log
      !gmx mdrun -v -deffnm simulacion_npt &> log05dd.log
```

:-) GROMACS - gmx mdrun, 2021.4-Ubuntu-2021.4-2 (-:

GROMACS is written by:

Andrey Alekseenko	Emile Apol	Rossen Apostolov
Paul Bauer	Herman J.C. Berendsen	Par Bjelkmar
Christian Blau	Viacheslav Bolnykh	Kevin Boyd
Aldert van Buuren	Rudi van Drunen	Anton Feenstra
Gilles Gouaillardet	Alan Gray	Gerrit Groenhof
Anca Hamuraru	Vincent Hindriksen	M. Eric Irrgang
Aleksei Lupinov	Christoph Junghans	Joe Jordan
Dimitrios Karkoulis	Peter Kasson	Jiri Kraus
Carsten Kutzner	Per Larsson	Justin A. Lemkul
Viveca Lindahl	Magnus Lundborg	Erik Marklund
Pascal Merz	Pieter Meulenhoff	Teemu Murtola
Szilard Pall	Sander Pronk	Roland Schulz
Michael Shirts	Alexey Shvetsov	Alfons Sijbers
Peter Tieleman	Jon Vincent	Teemu Virolainen
Christian Wennberg	Maarten Wolf	Artem Zhmurov

and the project leaders:

Mark Abraham, Berk Hess, Erik Lindahl, and David van der Spoel

Copyright (c) 1991-2000, University of Groningen, The Netherlands.

Copyright (c) 2001-2019, The GROMACS development team at

Uppsala University, Stockholm University and

the Royal Institute of Technology, Sweden.

check out <http://www.gromacs.org> for more information.

GROMACS is free software; you can redistribute it and/or modify it

under the terms of the GNU Lesser General Public License

as published by the Free Software Foundation; either version 2.1

of the License, or (at your option) any later version.

GROMACS: gmx mdrun, version 2021.4-Ubuntu-2021.4-2

Executable: /usr/bin/gmx

Data prefix: /usr

Working dir: /home/lg/repos/lgcursos/Docking-MDSims-

UnBosque2023/practicas/gromacs-mdsim-notebook

Command line:

gmx mdrun -v -deffnm simulacion_npt

Back Off! I just backed up simulacion_npt.log to ./#simulacion_npt.log.1#

Compiled SIMD: SSE4.1, but for this host/run AVX2_256 might be better (see log).

Reading file simulacion_npt.tpr, VERSION 2021.4-Ubuntu-2021.4-2 (single precision)

Changing nstlist from 10 to 50, rlist from 1 to 1.105

Using 1 MPI thread

Using 4 OpenMP threads

Back Off! I just backed up simulacion_npt.trr to ./#simulacion_npt.trr.1#

Back Off! I just backed up simulacion_npt.edr to ./#simulacion_npt.edr.1#
starting mdrun 'PEPSIN in water'

1000 steps, 2.0 ps.

step 900, remaining wall clock time: 4 s

Writing final coordinates.

Back Off! I just backed up simulacion_npt.gro to ./#simulacion_npt.gro.1#

step 1000, remaining wall clock time: 0 s

	Core t (s)	Wall t (s)	(%)
Time:	186.416	46.604	400.0
	(ns/day)	(hour/ns)	
Performance:	3.712	6.466	

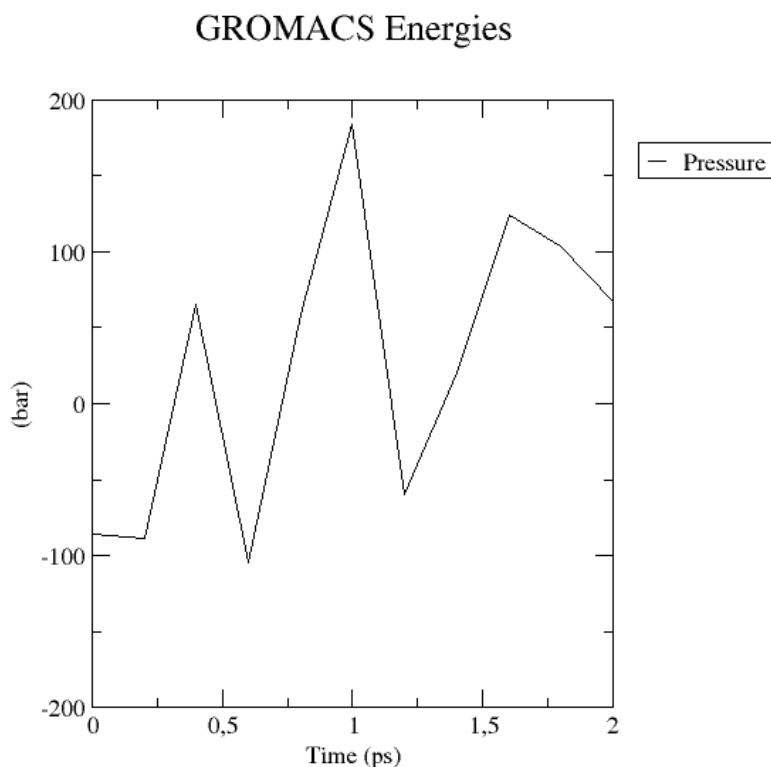
GROMACS reminds you: "It Was My Pleasure" (Pulp Fiction)

```
[68]: ## Resultados simulación NPT
!ls -lt|head -5
!echo 18 | gmx energy -f simulacion_npt.edr -o presion.xvg &> log05e.log
!grace -nxy presion.xvg -hdevice PNG -hardcopy -printfile presion.png
Image(filename='presion.png', width = 400, height = 400)
```

total 149200

```
-rw-rw-r-- 1 lg lg 28834 mar 22 11:39 simulacion_npt.log
-rw-rw-r-- 1 lg lg 4348 mar 22 11:39 simulacion_npt.edr
-rw-rw-r-- 1 lg lg 4288334 mar 22 11:39 simulacion_npt.gro
-rw-rw-r-- 1 lg lg 16408656 mar 22 11:39 simulacion_npt.trr
ls: write error: Broken pipe
```

[68]:



1.7 Simulación de Producción

```
[69]: !gmx grompp -f simulacion_prod.mdp -c simulacion_npt.gro -t simulacion_npt.cpt
      ↪ -p topol.top -o simulacion_prod.tpr &> log06a.log
      !gmx mdrun -v -deffnm simulacion_prod &> log06b.log
```

1.8 Análisis de la Simulación

1.8.1 Preprocesamiento de la trayectoria

```
[70]: !{ echo 1; echo 0; } | gmx trjconv -s simulacion_prod.tpr -f simulacion_prod.
      ↪ xtc -o simulacion_prod_noPBC.xtc -pbc mol -center &> log07a.log
```

1.8.2 Desviación cuadrática media raíz (RMSD)

```
[71]: !{ echo 4; echo 4; } | gmx rms -s simulacion_prod.tpr -f simulacion_prod_noPBC.xtc
      ↪ -o rmsd.xvg -tu ns &> log07b.log
```

[72]: *## Resultados simulación Producción*

```
!ls -lt|head -5
```

```
!grace -nxy rmsd.svg -hdevice PNG -hardcopy -printfile rmsd.png
```

```
Image(filename='rmsd.png', width = 400, height = 400)
```

total 162436

```
-rw-rw-r-- 1 lg lg      5170 mar 22 11:40 log07b.log
```

```
-rw-rw-r-- 1 lg lg       897 mar 22 11:40 rmsd.svg
```

```
-rw-rw-r-- 1 lg lg     5865 mar 22 11:40 log07a.log
```

```
-rw-rw-r-- 1 lg lg  2494240 mar 22 11:40 simulacion_prod_noPBC.xtc
```

[72]:

