

GROMACS: um método da Física Computacional

GROMACS: a method of computational physics

RESUMO

O GROMACS é uma importante ferramenta no campo da física molecular, sendo extremamente efetiva para modelagem e simulações de dinâmica molecular. Este trabalho tem como objetivo demonstrar uma aplicação prática do GROMACS em uma molécula simples, a proteína branca do ovo. Será feita a remoção das moléculas residuais de água, inserção de íons, minimização de energia, equilíbrio e a simulação da proteína solvatada. Analisamos os resultados por meio de uma série de gráficos gerados com as ferramentas de análise molecular incorporadas no pacote. Concluímos que o GROMACS é uma poderosa ferramenta da física-computacional, tendo grande valia para a área da biofísica molecular.

PALAVRAS-CHAVE: GROMACS. Química computacional. Dinâmica molecular.

ABSTRACT

GROMACS is an important tool in the field of molecular physics, being extremely effective for modeling and simulating molecular dynamics. This work aims to demonstrate a practical application of GROMACS in a simple molecule, the white egg protein. We will remove residual water molecules, insert ions, minimize energy, balance and simulate the solvated protein. We analyze the results through a series of graphs generated with the molecular analysis tools embedded in the software package. We conclude that GROMACS is a powerful tool of computational physics, having great value for the area of molecular biophysics.

KEYWORDS: GROMACS. Computational Chemistry. Molecular Dynamics.

Marcus Vinicius Soerensen

Spanhol

Vini0703@gmail.com

Universidade Tecnológica Federal
do Paraná, Toledo, Paraná, Brasil

Rafael Bertolini Frigori

frigori@utfpr.edu.br

Universidade Tecnológica Federal
do Paraná, Toledo, Paraná, Brasil

Recebido: 19 ago. 2020.

Aprovado: 01 out. 2020.

Direito autorial: Este trabalho está
licenciado sob os termos da Licença
Creative Commons-Atribuição 4.0
Internacional.



INTRODUÇÃO

O GROMACS é um pacote computacional de código aberto que permite efetuar simulações de dinâmica molecular e minimização de energia. Ele permite o estudo *in silico* da química computacional e da modelagem molecular. A modelagem molecular, via de regra, é utilizada para analisar a estrutura tridimensional das moléculas, ao passo que a química computacional permite compreender as interações intra e inter-moleculares desses sistemas.

A dinâmica molecular é um procedimento computacional que analisa, por meio da solução numérica das leis de Newton, os movimentos físicos de átomos e moléculas. Para tal, é preciso que seja fornecido um “campo de forças” (i.e. um potencial molecular “V”) capaz de descrever as interações entre as mais diversas espécies atômicas ao longo de um período temporal determinado. Podemos escrever essas interações como:

$$m_i \frac{\partial^2 r_i}{\partial t^2} = F_i, i = 1 \dots N \quad (1)$$

$$F_i = -\frac{\partial V}{\partial r_i} \quad (2)$$

Enquanto a Eq. (1) expressa o princípio fundamental da dinâmica, agindo sobre cada i-ésimo átomo de massa m_i , as forças F_i responsáveis por essa dinâmica são obtidas a partir do gradiente do potencial molecular V , como explicitado na Eq. (2). O GROMACS incorpora uma série de algoritmos paralelos capazes de resolver numericamente, e com enorme eficiência, esse sistema de equações para N partículas. Essas equações são resolvidas simultaneamente, de modo iterativo, em pequenos intervalos de tempo. Ainda, os algoritmos conhecidos como termostato e barostato permitem controlar a temperatura e a pressão do sistema. Para tal, o sistema é acompanhado e suas coordenadas são ajustadas e escritas em um arquivo em intervalos regulares. Essas coordenadas escritas como função do tempo são chamadas de trajetórias do sistema. Assim, conforme o tempo passa o sistema evolui a partir de uma condição inicial, em conformidade com as condições de contorno impostas pelo termostato e barostato, e caminha para encontrar seu equilíbrio termodinâmico. Nesse regime, as análises estruturais são feitas.

A título de ilustração, vale explicitar como a temperatura pode ser calculada através da energia cinética total do sistema de N partículas. A dedução dessas relações envolve um estudo aprofundado da chamada “teoria cinética dos gases”. Mas em síntese as relações mais importantes são resumidas nas seguintes equações:

$$E_k = \frac{1}{2} \sum_{i=1}^N m_i v_i^2 \quad (3)$$

$$\frac{1}{2} N_{df} kT = E_k \quad (4)$$

A Eq. (3) explicita a energia cinética E_k de todos os i -ésimos átomos, componentes do sistema molecular, movendo-se a velocidades v_i . Partindo dela podemos computar a temperatura T isolando-a a partir da Eq. (4), que relaciona termodinamicamente a temperatura do sistema e a sua energia cinética. Aqui, N_{df} é o grau de liberdade (em 3 dimensões N_{df} vale 3, por exemplo) e k é a constante

de Boltzmann. A temperatura e a energia cinética são medidas importantes para a minimização da energia configuracional do sistema. Resta dizer que, uma vez fixada a temperatura média do ambiente T , o termostato do GROMACS é capaz de inicializar as diversas velocidades iniciais v_i obedecendo uma “distribuição Maxwelliana” de velocidades, ou seja, de forma a respeitar o comportamento termodinâmico de uma proteína imersa em um fluido aquoso termalizado à temperatura fixada.

Assim, com o uso do Linux e do GROMACS foram realizadas simulações de interações de moléculas (proteína e seu solvente aquoso) descritas por um campo de forças. Neste relatório em específico foi utilizado a proteína branca do ovo em um ambiente com água e íons de Cloro para neutralização de cargas.

MATERIAL E MÉTODOS

Inicialmente, foi executada a instalação do Linux e dos pacotes necessários para o uso do GROMACS como os compiladores de C e C++ e o FFTW (Fastest Fourier Transform in the West). Em seguida, para o processo de simulação a estrutura tridimensional inicial da proteína que usaríamos foi obtida a partir do banco de dados de proteínas RCSB. Utilizando-nos de programas para visualizar as estruturas como o VMD ou o Chimera podemos observar a molécula em 3D, como ilustrado na Figura 1.

Figura 1 – Proteína branca do ovo



Fonte: <<https://www.rcsb.org/3d-view/1AKI/1>> Acesso em: 08 ago. 2020.

Uma vez feito o download dessa estrutura da molécula foi necessário remover os cristais de água da estrutura, para esse caso foi usado o comando `grep -v HOH 1aki.pdb > 1AKI_clean.pdb`, que irá limpar a parte HOH do arquivo 1aki.pdb e esse arquivo limpo será nomeado 1AKI_clean.pdb.

Após limparmos os cristais de água usaremos o comando `gmx pdb2gmx -f 1AKI_clean.pdb -o 1AKI_processed.gro -water spce`. Este comando lê os arquivos .pdb e adiciona hidrogênio as moléculas e gera as coordenadas no GROMACS. O comando espera que o usuário escolha um campo de força para a molécula ser submetida. Foi usado o campo OPLS-AA/L all-atom force field para esse experimento.

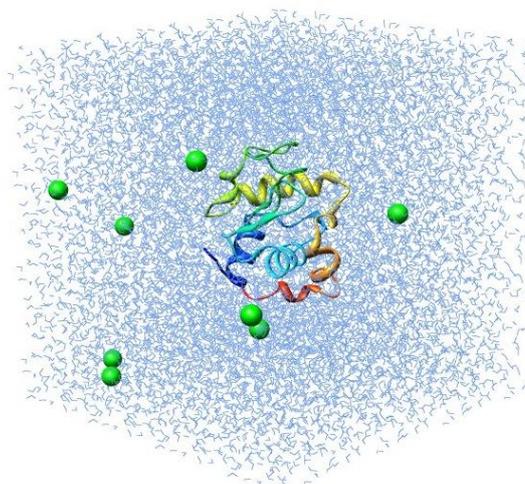
Com escolha do campo de força feita vamos definir uma caixa para ser colocada a proteína utilizando o comando `gmx editconf -f 1AKI_processed.gro -o 1AKI_newbox.gro -c -d 1.0 -bt cubic` que vai criar uma caixa com a proteína no centro e deixar pelo menos 1 nanômetro de distância das bordas da caixa.

Podemos agora adicionar o solvente da nossa proteína, água. Usando `gmx solvate -cp 1AKI_newbox.gro -cs spc216.gro -o 1AKI_solv.gro -p topol.top` estaremos aplicando o solvente na nossa proteína.

Usando o comando `gmx grompp -f ions.mdp -c 1AKI_solv.gro -p topol.top -o ions.tpr` iremos verificar a validade da nossa proteína agora com solvente e expandir a topologia dela em uma descrição atômica da mesma para que possamos adicionar íons.

Feito a verificação podemos agora usar `gmx genion -s ions.tpr -o 1AKI_solv_ions.gro -p topol.top -pname NA -nname CL -neutral` para adicionar os íons na nossa solução. O comando remove as moléculas de água e adiciona íons e define nomes positivos e negativos para os íons. Os íons de CL e NA são adicionados para tentar neutralizar a carga na proteína. São adicionados 8 íons de CL- para neutralizar a diferença de 8+ e ficamos com a seguinte proteína após a adição de íons

Figura 2 – Proteína com íons



Fonte: <http://www.mdtutorials.com/gmx/lysozyme/Images/system_solv_ions.jpg>

Acesso em 08 de out 2020

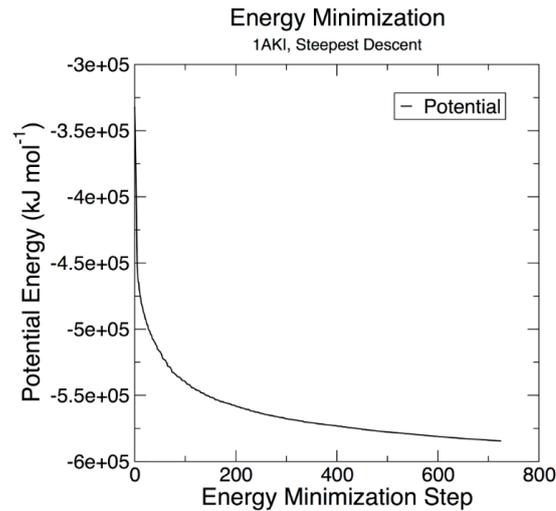
Em seguida, se faz necessário minimizar a energia do sistema garantindo que ele esteja relaxado e livre de forças espúrias que pudessem induzir “estouros” numéricos. Esse processo é muito parecido com o anterior, vamos usar o mesmo comando para montar a estrutura, sua topologia e simular os parâmetros que estamos buscando. Utilizando `gmx grompp -f minim.mdp -c 1AKI_solv_ions.gro -p topol.top -o em.tpr` estaremos atualizando nossa estrutura e topologia após a adição de íons e garantindo que nossa molécula foi atualizada.

Através do comando `gmx mdrun -v -deffnm em` faremos a minimização da energia do sistema. Este comando é a principal *engine* computacional de química

do GROMACS. Ele lê o arquivo de entrada e distribui a topologia conforme foi ordenado.

Agora com o uso de *gmx energy -f em.edr -o potencial.svg* teremos a extração da energia dos componentes ou distância dos laços do nosso arquivo que foi gerado anteriormente. Utilizando o Xmgrace teremos um gráfico da minimização de energia que vai convergir para a mínima energia potencial.

Figura 3 – Gráfico de minimização de energia potencial.

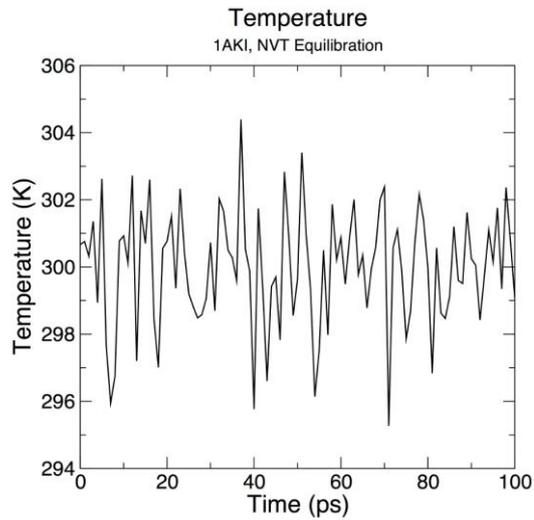


Fonte: Autoria própria.

Agora que garantimos a minimização de energia podemos começar a dinâmica molecular. Vamos equilibrar os solventes e os íons na proteína. Para isso vamos ajustar a temperatura para o sistema não colapsar. Vamos chamar o comando *grompp* e o *mdrun* novamente como fizemos na minimização de energia. Usando o *gmx grompp -f nvt.mdp -c em.gro -r em.gro -p topol.top -o nvt.tpr* e depois *gmx mdrun -deffnm nvt* para atualizarmos nossa estrutura e rodarmos nosso teste.

Após concluído vamos analisar a temperatura usando o comando *energy*. Teremos o seguinte gráfico depois de usar o comando *gmx energy -f nvt.edr -o temperature.svg*

Figura 4 – Gráfico da temperatura.



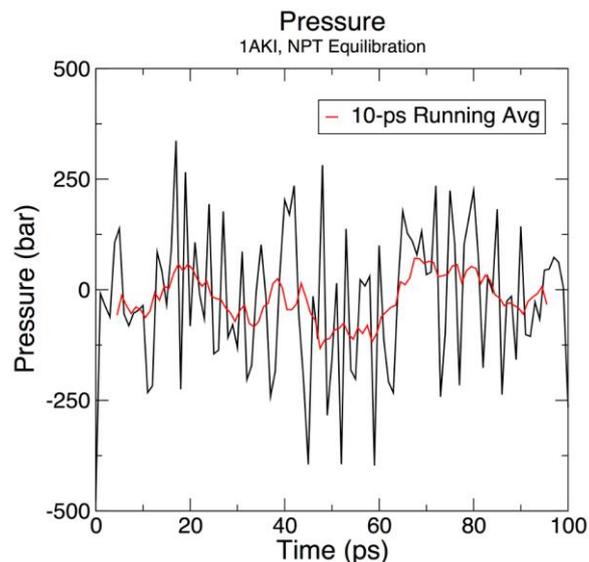
Fonte: A autoria própria.

Vemos que a temperatura gira em torno dos 300 Kelvin.

Com a temperatura estabelecida ajustaremos a pressão com os mesmos comandos *grompp* e *mdrun*, *gmx grompp -f npt.mdp -c nvt.gro -r nvt.gro -t nvt.cpt -p topol.top -o npt.tpr* e *gmx mdrun -deffnm npt*. Depois faremos o gráfico usando o comando *energy* como foi feito anteriormente *gmx energy -f npt.edr -o pressure.xvg*.

Teremos o seguinte gráfico:

Figura 5 – Gráfico da pressão.



Fonte: A autoria própria.

Após completar os dois equilíbrios no sistema iremos agora rodar a dinâmica molecular na nossa proteína. Com o comando *gmx grompp -f md.mdp -c npt.gro -t npt.cpt -p topol.top -o md_0_1.tpr* faremos uma estimativa de quantos núcleos usaremos para a PME(Particle-mesh Ewald), que é um algoritmo para tratar

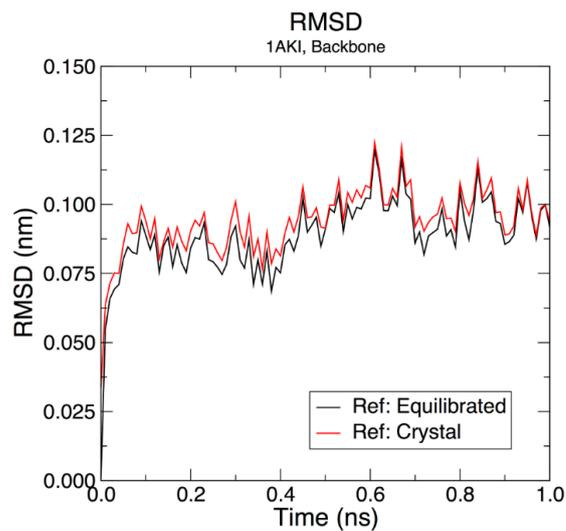
componentes que não tem ligações, e para a PP(*Particle-particle*) que lida com as interações entre as ligações. Agora iremos rodar o comando `gmx mdrun -deffnm md_0_1` para executar nossa dinâmica molecular.

Agora que já fizemos a simulação vamos converter a trajetória para corrigir periodicidade ou pulos da molécula para fora da nossa caixa com o seguinte comando `gmx trjconv -s md_0_1.tpr -f md_0_1.xtc -o md_0_1_noPBC.xtc -pbc mol -center`. Agora vamos analisar a dinâmica.

RESULTADOS E DISCUSSÃO

Para concluir graficaremos o desvio quadrático médio (RMSD) entre a estrutura simulada da molécula e sua estrutura cristalina de referência em função do tempo com os seguintes comandos `gmx rms -s md_0_1.tpr -f md_0_1_noPBC.xtc -o rmsd.svg -tu ns`. Aqui, a opção `-tu` é para converter o tempo para nanossegundos para termos uma maior clareza na saída. Agora para a estrutura cristalina fazemos `gmx rms -s em.tpr -f md_0_1_noPBC.xtc -o rmsd_xtal.svg -tu ns`. Plotando as duas juntas obtemos o gráfico:

Figura 6 – Gráfico da estrutura cristalina e molécula equilibrada.

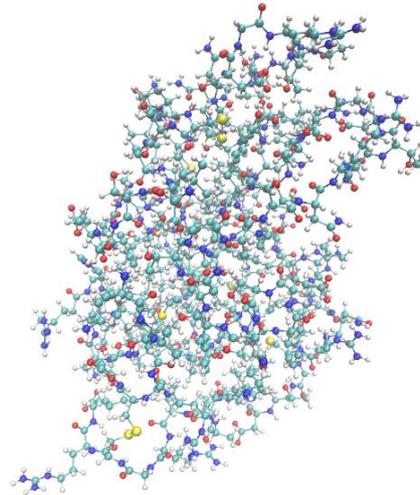


Fonte: Autoria própria

Como pode ser observado elas apresentam uma pequena diferença mostrando que a estrutura é estável. A diferença no momento inicial nos diz que a estrutura é diferente da estrutura cristalina e isso é esperado devido a minimização de energia que fizemos anteriormente.

E a molécula após todo o processo fica assim:

Figura 7 – Proteína após o processamento



Fonte: Autoria própria

CONCLUSÃO

Apesar de ser um procedimento simples, as tarefas executadas são extremamente importantes para o entendimento do funcionamento do GROMACS e da dinâmica molecular computacional como um dos mais importantes métodos da Física Computacional. O poder de processamento das máquinas está inversamente ligado ao tempo necessário para rodar os experimentos, ou seja, quão mais poderoso o PC mais veloz a execução do GROMACS.

AGRADECIMENTOS

Gostaria de agradecer o professor Rafael Bertolini Frigori por todo o apoio durante todo o projeto, pela preocupação com cada detalhe desde o trabalho em si assim como a preocupação com o psicológico nesse tempo de pandemia. Obrigado por tudo.

REFERÊNCIAS

BRANDEN, C. I.; TOOZE, J. Introduction to protein structure. New York: Garland Science, 1998.

BIGGIN, C. P.; BOND, J. P. Molecular Modeling of Proteins, New York: Andreas Kukol, 2015.

RAPAPORT, D. C. The Art of Molecular Dynamics Simulation, Cambridge University, 2013.

Manual do GROMACS 5.1.4. Disponível em: <
<http://manual.gromacs.org/documentation/5.1.4/index.html> >. Acessado em
23/08/2020.

FRIGORI et. al.. PathMolD-AB: Spatiotemporal pathways of protein folding using parallel molecular dynamics with a coarse-grained model, 2020; Disponível em: <
<https://doi.org/10.1016/j.combiolchem.2020.107301> >, Acessado em
08/10/2020.

GROMACS Tutorial Lysozyme in Water; J. A. Lemkul. Disponível em: <
<http://www.mdtutorials.com/gmx/lysozyme/index.html> >. Acessado em
23/08/2020.