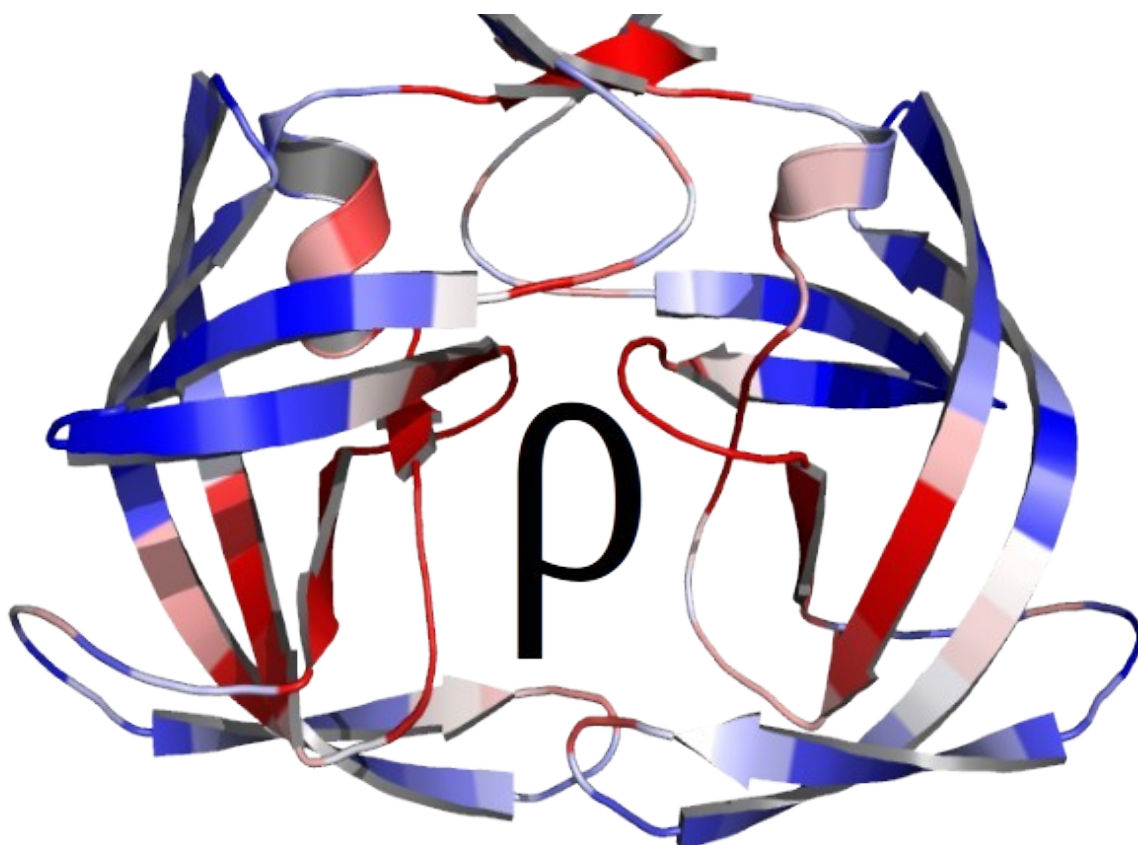

XI EMMSB

Instruções dos Tutoriais do PRIMoRDiA

Igor Barden Grillo
(barden.igor@gmail.com)
github.com/bardenChem

3 de setembro de 2024



PRIMoRDiA
PRIMoRDiA Macromolecular Reactivity Descriptor Access

Esse documento tem como objetivo trazer todos os comandos e instruções diretas para a execução dos tutoriais ministrados no mini-curso na XI edição da Escola de Modelagem Molecular de Sistemas Biológicos (EMMSB) do Laboratório Nacional de Computação Científica (LNCC). Para a descrição completa dos tutoriais, material de referência e ilustração dos resultados esperados, confira o outro documento incluso nos materiais disponíveis desse mini-curso.

1 Tutorial 1: Complexos Proteína-Ligante

Nesse tutorial iremos:

1. Montar o input para os cálculos no PRIMoRDiA para os complexos RTA-Ligante
2. Executar o PRIMoRDIA
3. Analisar os arquivos de saída
4. Visualizar os mapas de reatividade no Pymol e salvar as imagens em alta resolução

Primeiro comando:

```
computer@user:$ path/to/PRIMoRDiA_1.25v -input -op 3 -p mopac
```

Segundo Comando:

```
computer@user:$ path/to/PRIMoRDiA_1.25v -f primordia_editado.input
```

Para a análise dos resultados no Pymol, vamos primeiro abrir o software e executar o comando

```
@19m_pymols_pdb.pym
```

Vamos ativar o objeto "netphilicity" e executar o seguinte comando no Pymol

```
spectrum b, blue_white_red, minimum=-0.1,maximum=0.1
```

Depois vamos ativar o objeto "hardness_fukui_pot_left" e executar o comando

```
spectrum b, white_yellow_orange_red_black, minimum=-0.1,maximum=0.25
```

E depois ajustar o comando para

```
spectrum b, white_yellow_orange_red_black, minimum=-0.15,maximum=0.23
```

Ajustar o tipo de tracejado ao redor da molécula

```
set ray_trace_mode, 1
```

Renderizar a molécula em alta qualidade

```
ray antialias=2
```

E salvar

```
png complex_rta_19m_hardness.png
```

2 Tutorial 2: Gerando Trajetória de Catálise Enzimática

Com todos os programas devidamente instalados, nesse tutorial, vamos executar um script em Python já fornecido no material do mini-curso, a realizar a análise dos resultados. Durante o mini-curso, vamos avaliar como foi montado esse script em Python para que se de uma noção de como pode ser montado para outros sistemas e simulações.

Dentro da pasta do mini-curso, no diretório Reaction, já se encontram os arquivos de campo de força e coordenadas do sistema enzimático de teste, o script em Python e algumas etapas já prontas, pois são demoradas. O único comando a ser executado será:

```
computer@user:$ python3 Reaction_TIM.py
```

Outras atividades serão: abrir os gráficos de energia versus coordenada de reação, e visualizar o caminho de reação no Pymol.

3 Tutorial 3: Análise de Reatividade de Caminho de Reação Enzimático

Nesse tutorial vamos executar o PRIMoRDIA em um modo diferente, que trata arquivos de estrutura eletrônica que estejam relacionados entre si por ordem de Frames, ou seja uma trajetória. Vamos usar os arquivos ".aux" do método RM1 calculado no refinamento de energia no mopac, mas primeiro, é necessário mudar o nome desses arquivos, tirando o "rm1", do nome para que o PRIMoRDIA reconheça a sequência dos frames pela extensão do arquivo. Isso pode ser realizado facilmente com essas linhas de comando no terminal.

```
computer@user:$ for file in *rm1.aux;
computer@user:$ do
computer@user:$ mv "$file" "${file/_rm1.aux/.aux}";
computer@user:$ done
```

```
computer@user:$ for file in *rm1.pdb;
computer@user:$ do
computer@user:$ mv "$file" "${file/_rm1.pdb/.pdb}";
computer@user:$ done
```

O input para esse tutorial pode ser copiado e colado em um editor de texto a partir dos dados fornecidos na próxima caixa de texto

Listing 1: Input editado para execução do tutorial 3

```
#RT reaction
#PR eband 3 Rscript pymols
#Reaction RC1 137 138 73
#Reaction dimX 20
#TRJ residues 3 4 9
3 frame.aux true 30 10 frame.pdb mopac 0 0 0 0 EW
```

Salve esse arquivo na pasta onde os arquivos de estrutura eletrônica ficaram salvos, e rode com o seguinte comando

```
computer@user:$ path/to/PRIMoRDiA_1.25v -f reaction_analysis.input
```

Para gerar os gráficos de variação dos descritores globais e para os átomos da coordenada de reação usando o R, vamos rodar o comando abaixo

```
computer@user:$ Rscript frame0_reaction_analysis.R
```

Depois de analisar esses gráficos, vamos gerar algumas visualizações dos descritores volumétricos no Pymol. Só abrir o programa e executar o seguinte comando

```
@frame0_pymols.pym
```

Vamos explorar as visualizações volumétricas e aprender a renderizar e salvar as imagens.

Para os descritores condensados, vamos abrir a estrutura inicial e a estrutura do estado de transição com os seguintes comando no Pymol

```
@frame0_pymols_pdb.pym
```

```
@frame15_pymols_pdb.pym
```

Para finalizar as análises, vamos utilizar o R novamente para gerar gráficos com as estatísticas dos descritores para os resíduos e substrato durante a trajetória, com o seguinte comando

```
computer@user:$ Rscript frame0_reaction_analysis.R
```

4 Instalação dos Programas

A maioria dos softwares já estarão instalados nos computadores do mini-curso. Para alguns deles, nós veremos durante o tutorial como configurar variável de caminho e outras opções. No entanto, as instruções da instalação dos programas citados nesses tutoriais estão logo abaixo. Informações mais específicas sobre compilação, instalação e uso dos binários já compilados, consulte o guia de usuário fornecido no repositório.

4.1 PRIMoRDiA Software

Para baixar os dados do repositório é possível usar o seguinte comando do git:

```
computer@user:$ git clone https://github.com/igorChem/PRIMoRDiA1.0v.git
```

Por padrão, o PRIMoRDiA utiliza o g++ e a biblioteca Eigen 3. Para assegurar que sua máquina tenha o necessário para compilar o PRIMoRDiA, execute os comandos abaixo

```
computer@user:$ sudo apt install g++  
computer@user:$ sudo apt install cmake  
computer@user:$ sudo apt install libeigen3-dev
```

Depois de descomprimir a pasta baixada com a versão estável, entre no diretório e faça a compilação usando os seguintes comandos:

```
computer@user:$ cd /caminho/da/pasta/PRIMoRDiA1.0v  
computer@user:$ cmake .  
computer@user:$ make
```

O processo de compilação deve ocorrer sem problemas e gerar um arquivo binário chamado "PRIMoRDIA_1.25v_LINUX64 ", caso a versão 1.25 seja a escolhida (mais recomendado). Com o comando descrito abaixo é possível criar um atalho para executar o programa de qualquer diretório do seu sistema

4.2 Pymol

Nos ambientes linux baseados em Debian, isto é, que utilizam o sistema apt-get para a instalação dos pacotes pelo repositório, o Pymol apresenta um versão padrão em condições satisfatórias para uso geral e para as virtualizações requeridas nesses tutoriais. Portanto, com o seguinte comando é possível instalar o programa

```
computer@user:$ sudo apt-get install pymol
```

4.3 pDynamo

Para instalar o pDynamo primeiro baixe ou clone o repositório do Git Hub, que pode ser feito com o seguinte comando

```
computer@user:$ git clone https://github.com/pdynamo/pDynamo3.git
```

Para a instalação é necessário conferir a existência de algumas bibliotecas, compiladores e módulos instalados no Python: Python 3, versão 3.5 ou mais atualizada; Cython, instala o compilador e o módulo pelo pip

```
computer@user:$ sudo apt-get install cython3
```

```
computer@user:$ pip install cython
```

PyMal

```
computer@user:$ sudo apt-get install python3-yaml
```

É um compilador de C, mas para usar o PRIMoRDIA, o compilador do GCC já se instalou quando o g++ foi instalado. A instalação do pDynamo se dá no diretório "installation", com o seguinte comando

```
computer@user:$ python3 Install.py -f
```

Para o programa funcionar ele precisa ter suas biblioteca no caminho do Python e a configurar de outras variáveis. Para isso, abra o arquivo ".bashrc", localizado no seu diretório "/home/usuário" e adicionar as linhas (não esqueça de editar elas antes)

```
PDYNAMO3_HOME=/home/usuário/programs/pDynamo3;export PDYNAMO3_HOME
export PDYNAMO3_PARAMETERS=$PDYNAMO3_HOME/parameters
export PDYNAMO3_PYTHONCOMMAND=python3
export PDYNAMO3_SCRATCH=$PDYNAMO3_HOME/scratch
export PDYNAMO3_STYLE=$PDYNAMO3_PARAMETERS/ccsStyleSheets/defaultStyle.css
export PYTHONPATH=$PDYNAMO3_HOME:$PYTHONPATH
```

4.4 pDynamo_Scripts

A "instalação" do pDynamo_scripts se dá pelo download do repositório do Git Hub, que pode ser feito apenas clonando o repositório com o comando mostrado a baixo

```
computer@user:$ git clone https://github.com/bardenChem/pDynamo3_scripts.git
```

Para ativar o pDynamo3 e pDynamo_scripts é necessário colocar os scripts no caminho de execução de bibliotecas do Python3, o que foi deixado para esse tutorial, pois é algo que as vezes causa confusão. Abra o arquivo ".bashrc" e adicione a seguinte linha.

```
computer@user:$ export PYTHONPATH=$HOME/pDynamo3_scripts:$PYTHONPATH
```

O pDynamo_scripts ainda está em desenvolvimento, mas diversas funcionalidades já estão disponíveis e podem ser conferidas nos arquivos da pasta de teste, que tem um modelo básico para rodar diversos tipos de simulação.