

Virtual Screening Workshop: Parte I

Isabella Guedes¹ e Alessandro S. Nascimento²

¹ *Laboratório Nacional de Computação Científica - LNCC/MCT. Av. Getúlio Vargas 333 - 1A11 - Quitandinha, Petrópolis, RJ, Brazil. 25651-075.*

² *Instituto de Física de São Carlos, Universidade de São Paulo. Av. Trabalhador SãoCarlense, 400. Parque Arnold Schimidit. São Carlos, SP, Brazil. 13566-590.*

Resumo

Nesta série de documentos, faremos um tutorial sobre a triagem virtual, passando pela preparação de um alvo, calibração do modelo e avaliação dos resultados. Na sequência, faremos a seleção de moléculas para a triagem, a triagem e a avaliação dos resultados. Estes documentos foram preparados como parte do material da [IX EMMSB 2018](#). Os arquivos gerados nesta prática também estão disponíveis no [GitHub](#).

1 Introdução

Nesta primeira parte, faremos a preparação de um alvo e iniciaremos a calibração do modelo de *doc-king*. Para esta etapa, estamos assumindo que o aluno tenha um acesso a um computador com ambiente Linux. Os seguintes softwares serão usados: [UCSF Chimera](#) [1], [OpenBabel](#) [2], além de acesso a internet através de qualquer navegador. O programa OpenBabel está disponível nos repositórios de diversas distribuições linux (Ubuntu, por exemplo).

2 Organizando os Arquivos

Projetos de Triagem Virtual tem o potencial de gerar centenas a milhares de arquivos facilmente e é fácil se perder com tantos arquivos. Vamos começar criando uma estrutura de organização de arquivos que facilite nossa vida.

- Vamos criar uma pasta onde serão colocados todos os arquivos do projeto: `$ mkdir EMMSB_VS`. Em seguida vamos entrar nesta pasta: `$ cd EMMSB_VS`;
- Nesta pasta, vamos criar pastas para guardar os arquivos MOL2, para a calibração, e para a triagem: `$ mkdir calibrate mol2 docking`.

3 Preparação do Receptor e Ligante de Referência

Nesta etapa, vamos fazer a preparação de um arquivo do receptor e um arquivo de um ligante de referência. O objetivo desta etapa é termos, ao final, um arquivo MOL2 para cada uma destas moléculas, com as devidas cargas e tipos atômicos atribuídos. Para isto, vamos usar o programa [UCSF Chimera](#).

- Vamos começar inspecionando a estrutura [1L2S do PDB](#). A estrutura parece OK. Vamos prosseguir.
- No programa Chimera, vá até *File* e *Fetch by ID*. No campo *PDB*, digite 1L2S.
- A estrutura tem duas cadeias, A e B. Vamos usar apenas uma destas. Selecione a cadeia A: *Select -> Chain -> A*. Depois vá até *Actions -> Atoms/Bonds -> delete*. Há um ligante na interface entre as cadeias A e B. Vamos apagá-la também. Clique em qualquer átomo deste ligante com o botão *Ctrl* pressionado. Em seguida aperte a seta para cima do teclado para expandir a seleção para todo o ligante. Sem seguida, vá em *delete* novamente.
- Nossa estrutura agora contém apenas a proteína, o ligante e moléculas do solvente. Vamos analisá-la. Vá em *Favorites -> Command Line*. Na linha de comando, digite `focus :STC`.
- Vamos analisar os aminoácidos no entorno do ligante: `sel :STC za < 5`. Em seguida expanda a seleção apertando uma vez a seta para cima do teclado. Finalmente, digite `disp sel`.
- Há três moléculas de água próximas ao grupo carboxilato do ligante que podem ser importantes para as interações ligante-receptor. Analise estas moléculas.
- A molécula HOH481 em particular parece ser parte de um conjunto de interações envolvendo a THR316 (cadeia principal e lateral). Vamos mantê-la: `sel :HOH`. Agora todas as moléculas de água foram selecionadas (ainda que não estejamos vendo). Mantenha as teclas *Ctrl+Shift* pressionadas e clique sobre a molécula de água HOH481. Isto fará com que

ela fique desselecionada. Agora, vamos apagar as demais: **del sel**. Por hora, temos receptor e ligante. Vamos salvar este complexo: *File -> Save PDB*. Salve como *complex.pdb* na pasta EMMSB_VS.

- Agora vamos adicionar os hidrogênios às moléculas: *Tools -> Structure Editing -> AddH*. Para a escolha dos estados de protonação, marque a opção *Unspecified (Determined by method)*. Desmarque a opção *Consider each model in isolations of all others*. Inspeção o modelo gerado e veja se este modelo faz sentido. Vamos salvar novamente este modelo como *complexH.pdb*.
- Agora vamos focar no receptor: **sel :STC** e depois **del sel**. Apagamos o ligante e ficamos apenas com o receptor agora. Vá em *Tools -> Structure Editing -> Dock Prep*. Na janela que se abre, desmarque a opção *delete solvent* (já fizemos isto). Clique em OK.
- Confirma a nova adição de hidrogênios da mesma forma (algumas cadeias laterais podem ter sido alteradas e, por isto, pode ser necessário re-adicionar hidrogênios. Aqueles que já foram adicionados serão mantidos). Na janela de adição de cargas, confirme o campo de força *AMBER ff14SB* e clique em OK. Finalmente, salve o modelo como *ampc_rec.mol2*.
- Agora vamos fazer algo similar para o ligante de referência: *File -> Close Session*. Em seguida: *File -> Open* e selecione o arquivo *complexH.pdb*. Vamos selecionar o ligante novamente e inverter a seleção: **sel :STC**. Em seguida, *Select -> Invert (all models)* e, depois, **del sel**. Agora sobrou apenas o ligante.
- Vamos visualizar o ligante com hidrogênios: **disp**. Agora vá em *Tools -> Structure Editing -> Add Charge*. Na janela que se abre, confirme que a opção AM1-BCC (método de carga) está selecionado e clique em OK. O Chimera já percebe que o ligante deve ter carga total de -1 elétron. Clique OK. As cargas serão reajustadas com o método AM1-BCC através dos programas ANTECHAMBER e SQM, que são parte do [AMBERTOOLS](#) e que são distribuídos juntamente com o UCSF Chimera. Esta etapa pode levar alguns instantes.
- Finalmente, vá em *File -> Save Mol2* e salve como *ampc_xtallig.mol2*.
- Finalmente, abra os dois arquivos conjuntamente e verifique mais uma vez.

Outras formas de preparação do receptor de forma mais automatizada existem. Alguns servidores como o [H++](#) ou o [PDB2PQR](#) podem fazer isto. Vale lembrar também que, no caso do servidor [DockThor](#), muitas das etapas de preparação do receptor e do ligante também são feitas automaticamente.

4 Seleção de Compostos Ativos - Opção 1

Para a seleção de compostos ativos, usaremos o banco de dados [ChEMBL](#). Em particular, recomendando usar a [versão beta do banco](#).

- No campo de busca, digite *AMPC*;
- No campo *Search Results*, clique em *Targets*. Encontre a AMPC de *Escherichia coli* K-12 (ChEMBL2026). Na linha deste alvo, há um link para os 61963 compostos ativos no alvo. Clique no link.
- Vamos aplicar alguns filtros. No campo esquerdo, vamos filtrar por *Targets = [1-3]* e *Bioactivities = 1*. Dos 449 compostos remanescentes, vamos navegar pelas atividades, clicando em *Browse Activities*.
- Finalmente, vamos filtrar por compostos que tem K_i . Agora temos 23 compostos. Agora voltemos à busca de compostos clicando em *Browse Compounds*.
- Nesta janela, vamos selecionar todos os compostos e fazer o download em formato SDF. Salve o arquivo no diretório EMMSB_VS com o nome *actives_chembl.sdf*.
- Vamos converter o arquivo SDF em arquivo SMILES. Para isto, o programa OpenBabel é bastante útil. O comando a ser usado é `$ babel -isdf actives_chembl.sdf -osmi actives_chembl.smi`.
- Vamos usar o banco de dados [ZINC](#) para obter nossos compostos em formato MOL2 e devidamente preparados. Para isto, clique em *Substances* e vamos fazer a busca por SMILES fazendo o upload do arquivo *actives_chembl.smi*. Selecione o arquivo e clique em *Search Many*. Na janela que se abre, clique na seta apontada para baixo e selecione *MOL2*.

5 Seleção de Compostos Ativos - Opção 2

Outra opção para a seleção de compostos ativos é através do banco de dados ZINC [3]. O ZINC é um banco de dados de moléculas voltados para o *docking* e quimioinformática em geral.

- Vá para a página do ZINC, clique em *Biological* e, depois, em *Genes*.
- No campo de busca, busque por *AMPC*. Clique no link para resultado do gene *AMPC* encontrado. Na página que se abre, clique no botão *Browse All* que aparece na linha *Highest Affinity Substances*.
- Neste momento, temos algo em torno de 304 compostos. Vamos filtrar para selecionar os compostos de mais alta afinidade. No botão que tem uma etiqueta desenhada (ao lado de *Filters*), selecione a opção *10 nM*, ou seja, ligantes de afinidade de pelo menos 10 nM. Nossa lista fica reduzida a 31 compostos ativos iniciais.
- Clique no botão que tem uma seta apontada para baixo e selecione a opção SMI para fazer o download do arquivo com as moléculas em formato SMILES. Salve este arquivo como *actives.smi* na pasta *EMMSB_VS*.
- Vamos separar os códigos ZINC deste arquivo: `$ more actives.smi | grep -v affinity | awk '{print $2}'`.
- Finalmente, no site do ZINC, em *Substances*, copie e cole no campo *Search Using Many* os códigos ZINC que extraímos no passo anterior. No botão *Output format*, selecione MOL2 para fazer o download do arquivo em formato MOL2. Salve este arquivo como *actives.mol2*.
- Alguns dos ativos podem não estar disponíveis na versão atual do ZINC em formato MOL2. Vamos checar quantos ativos realmente temos: `$ more actives.mol2 | grep ZINC | grep -v '\.' | grep -v RESIDUE > actives_found.zinc` e, em seguida, `wc -l actives_found.zinc`. Com isto vemos que temos 13 ativos encontrados. É suficiente para o nosso propósito aqui. Numa campanha de VS, poderíamos expandir o número de ativos aplicando filtros menos restritivos, i.e., afinidades menores que 10 nM.

6 Seleção de Compostos *Decoys*

Para a seleção de compostos decoys, vamos usar o banco de dados DUDE.

- Vamos extrair os SMILES dos 13 compostos encontrados pelo ZINC: `$ for i in `more actives_found.zinc`; do grep $i actives.smi | awk '{print $1}'; done`
- Vá ao site do DUDE e clique em *Generate*. No campo *Paste Smiles*, cole o resultado obtido no comando do item anterior. No campo *send results to*, adicione seu endereço de e-mail e responda ao *human test*. A geração de decoys é relativamente rápida e deve ficar pronta em alguns minutos.
- Ao receber o e-mail de alerta do final deste job no DUDE, faça o download do arquivo *dude-decoys.tar.gz*. Decompile o arquivo usando o comando `$ tar -xzf dude-decoys.tar.gz` na pasta *EMMSB_VS*. Entre na pasta descompactada e vamos gerar uma lista dos decoys usando os códigos ZINC:
 - `$ cd dude-decoys;`
 - `$ for i in `ls decoys`; do more decoys/${i} | grep -v ligand | awk '{print $2}' > decoys.zinc ; done`
 - `$ wc -l decoys.zinc`
- O último comando deve mostrar a quantidade de moléculas que foram geradas como decoys.
- Finalmente, vamos à página do ZINC, copiar e colar novamente os códigos ZINC e fazer o download do arquivo em formato MOL2, como no caso da geração dos ativos. Renomeie este arquivo para *decoys.mol2* e salve no diretório *EMMSB_VS*.

7 Preparando as moléculas para o Docking

Nesta etapa, já temos os arquivos MOL2 com ativos e decoys. Vamos fazer a preparação final para usá-los no docking. Vamos partir do pressuposto que os arquivos *actives.mol2* e *decoys.mol2* estão na pasta *EMMSB_VS*.

- Vamos separar os arquivos em arquivos individuais na pasta *mol2*:
 - `$ cd mol2;`
 - `$ babel -imol2 ../actives.mol2 -omol2 actives..mol2 -m`

- `$ babel -imol2 ../decoys.mol2 -omol2 decoys..mol2 -m`
- Estes comandos usam o OpenBabel para gerar um conjunto de arquivos `actives.i.mol2`, com *i* indo de 1 a N. O mesmo para os decoys. Podemos ainda reduzir o espaço em disco compactando os arquivos: `$ gzip *.mol2`.

8 Docking

8.1 Docking com o LiBELA

Vamos preparar um arquivo de input para o docking. Vá para a pasta *calibrate*. Nesta pasta, crie um arquivo de input no seu editor de texto favorito (vim, nano, gedit, notepad++, etc). Salve o arquivo como *libela.inp*. O modelo está no final deste documento.

Agora, vamos dizer ao LiBELA [4] quais moléculas ele deve docar. Para isto, vamos utilizar 13 ligantes e 390 decoys (relação de 1 para 30): `$ for i in `seq 1 13`; do echo ../mol2/actives.$i.mol2.gz » multimol.dat; done`

Em seguida, vamos fazer o mesmo para os decoys: `$ for i in `seq 1 390`; do echo ../mol2/decoys.$i.mol2.gz » multimol.dat; done`

Finalmente, estamos prontos para fazer esta etapa de docking: `$ time McLiBELa.openmp libela.inp`. Na primeira etapa da execução, o programa fará o cálculo dos potenciais em *grids*. Na sequência, fará o docking de ligantes e decoys.

Referências

- [1] E F Pettersen, T D Goddard, C C Huang, G S Couch, D M Greenblatt, E C Meng, and T E Ferrin. UCSF chimera - A visualization system for exploratory research and analysis. *Journal of Computational Chemistry*, 25:1605–1612, 2004.
- [2] N M O’Boyle, M Banck, C A James, C Morley, T Vandermeersch, and G R Hutchison. Open Babel: An open chemical toolbox. *J Cheminform*, 3:33, 2011.
- [3] J J Irwin, T Sterling, M M Mysinger, E S Bolstad, and R G Coleman. ZINC: A Free Tool to Discover Chemistry for Biology. *J Chem Inf Model*, 52(7):1757–1768, 2012.
- [4] Heloisa dos Santos Muniz and Alessandro S. Nascimento. Ligand- and receptor-based docking with LiBELA. *Journal of Computer-Aided Molecular Design*, 29(8):713–723, 2015.

```

### LiBELa input file for docking

# mode
mode                dock
dock_parallel       yes
parallel_jobs        4

# input files

rec_mol2             ../ampc_rec.mol2.gz
lig_mol2             ../ampc_xtallig.mol2.gz
reflig_mol2          ../ampc_xtallig.mol2.gz
mol2_aa              no
multifile            multimol.dat

# force field parameters

scoring_function     3
dielectric_model      r
diel                  1.0
deltaij              0.0
deltaij_es           0.0
use_grids             yes
use_pbsa              no
grid_spacing          0.5
grid_box              30.0    30.0    30.0
write_grids           McGrid

# Optimization

search_box            12.0    12.0    12.0
minimization_tolerance 0.000001
minimization_delta    0.000001
dock_min_tol          0.000001
minimization_timeout  30
overlay_optimizer      ln_auglag
energy_optimizer       direct
ignore_h              no
deal                  no
elec_scale            1.0
vdw_scale              1.0
sort_by_energy        no

# output

output_prefix         ampc
write_mol2            no

# flexible ligands

generate_conformers   yes
number_of_conformers  10
conformers_to_rank    1
conformer_generator    GA
conformer_min_steps   1000

```