

DESENVOLVIMENTO DO SOFTWARE DE VIRTUAL SCREENING VINA2NETWORK

RAFAEL DANELON DOS SANTOS WOLOSKI¹; FREDERICO SCHMITT KREMER²;
LUCIANO DA SILVA PINTO³

¹Universidade Federal de Pelotas – rafael.woloski@ufpel.edu.br

²Universidade Federal de Pelotas – fred.s.kremer@gmail.com

³Universidade Federal de Pelotas – ls_pinto@hotmail.com

1. INTRODUÇÃO

A procura por estruturas químicas para a utilização destas como possíveis medicamentos, normalmente, envolvia screening de alto rendimento (HTS) de todos os possíveis espaços químicos dentro dos limites dos compostos de alguma biblioteca de ligantes (MCINNES, 2007). Este era um processo custoso e trabalhoso, que envolvia testar todas as moléculas válidas *in vitro*, tendo algumas limitações como a incapacidade de usar estruturas sintéticas (BELL et al., 2013).

Virtual Screening surgiu como um método barato para identificar moléculas chave ou moléculas ativas usando abordagens *in silico*, ou seja, utilizando técnicas computacionais para prever virtualmente a capacidade de uma estrutura, como uma molécula ou proteína, de se ligar a um alvo específico (REDDY et al., 2007). Geralmente quando falamos em *virtual screening* nos referimos a análise de um grande número de compostos contra um alvo, mas nem sempre é esse o caso. Pode ser executado tanto como screening baseado na conformação estrutural tridimensional da proteína e do ligante (Docking) ou abordagem baseada em ligantes, considerando similaridades na estrutura química de pequenas moléculas (POLGÁR; KESERU, 2011).

Após o desenvolvimento dos primeiros algoritmos como UCSF Dock (KUNTZ et al., 1982), as técnicas de *Virtual Screening* continuaram a aumentar em credibilidade como fonte de moléculas para o início do processo da descoberta de novos fármacos. Métodos baseado em estrutura (como Docking de Alto Rendimento) avançaram devido a grande quantidade de estruturas tri-dimensionais depositadas em bancos de dados de proteínas, como o PDB, que já contem mais de 100 mil estruturas arquivadas, número que aumenta exponencialmente (MCINNES, 2007).

O processo de *docking* se inicia com a aplicação dos algoritmos do software utilizado, que posiciona as moléculas no sítio ativo do alvo. Esse é um processo que se torna complicado devido ao grande número de conformações diferentes que até mesmo moléculas simples podem ter. Além disso, deve ser executado com precisão suficiente para identificar a conformação que melhor se adequa ao receptor, assim como ser rápido o suficiente para ser possível fazer a análise de diversos compostos em um tempo hábil (KITCHEN et al., 2004). Na Tabela 1 podem ser vistos alguns dos *softwares* de *docking* mais utilizados, assim como seus algoritmos.

Programa	Referência	Algoritmo de Docking
Autodock 4	(MORRIS et al., 2009)	Algoritmo Genético
DOCK	(KUNTZ et al., 1982)	Algoritmo Geométrico

FlexX	(RAREY; KRAMER; LENGAUER, 1995)	Algoritmo de Construção Incremental
FRED	(MCGANN, 2012)	Algoritmo de Docking de Estrutura Rígida
Glide	(FRIESNER et al., 2004)	Algoritmo de Filtros Hierárquicos
Gold	(JONES; WILLETT; GLEN, 1995)	Algoritmo Genético
Surflex	(JAIN, 2003)	Algoritmo Baseado em Similaridade Molecular
Autodock Vina	(TROTT; OLSON, 2010)	Algoritmo Genético

O presente trabalho apresenta o desenvolvimento de uma ferramenta que utiliza vários computadores conectados em rede ou pela internet para a análise de *virtual screening* utilizando o software Autodock Vina.

2. METODOLOGIA

O programa foi desenvolvido utilizando a linguagem de programação Python e foi dividido em duas partes distintas: o modo cliente e o modo servidor. O modo servidor é utilizado na máquina que vai servir de servidor, ou seja, o computador que vai hospedar os arquivos utilizados no docking, assim como distribuir estes para as máquinas rodando no modo cliente. Além disso, também é responsável por receber os resultados do docking.

O modo cliente é utilizado nas máquinas que vão executar o docking usando o Autodock Vina. A sua função é acessar os arquivos de ligantes e receptor do servidor e realizar o docking com estes, enviando os resultados ao servidor para análise. A entrada de dados no programa funciona por linha de comandos, sendo a maioria dos argumentos utilizados aqueles também usados pelo Autodock Vina, adicionados alguns próprios para o Vina2Cluster, como definição de modo (Cliente ou servidor), ip, porta e senha para a conexão. Para conferir se o usuário fez a entrada correta dos dados, foi utilizada a biblioteca Argparse, que só permite que o programa siga o funcionamento se todos os argumentos principais forem corretamente utilizados. Os argumentos então são armazenados dentro de um arquivo JSON, para posteriormente ser transferido para os clientes. Para a transmissão de dados foram utilizadas duas bibliotecas: Ftplib, já nativa da versão 2.7 do Python e utilizada para se conectar a servidores FTP, e Pyftplib, uma biblioteca não-nativa, que é utilizada para criar um servidor FTP (*File Transfer Protocol*) temporário no servidor, de acordo com as especificações necessárias, sem fazer modificações no sistema. A transmissão de dados funciona em um ciclo, onde o servidor transmite o arquivo JSON, o receptor e um dos ligantes para cada um dos clientes, que fazem a análise pelo Autodock Vina – utilizando a biblioteca Subprocess também nativa do Python, e então mandam os resultados de volta para o servidor. Quando o servidor recebe os arquivos de resultado, ele manda novos ligantes para os computadores que já terminaram a tarefa, até todos os ligantes terem sido devidamente processados.

3. RESULTADOS E DISCUSSÃO

Após construído o programa, foi testada a sua execução em função do objetivo inicial, que era a emulação de um cluster. Foram utilizados dez ligantes escolhidos aleatoriamente do banco de dados PDB (RCSB), utilizando a Proteína Tirosina Quinase ABL1 como receptor. O Autodock Vina utiliza 6 entradas de dados diferentes para definir os possíveis locais de ligação dos ligantes na proteína: *Center X*, *Y* e *Z* e *Size X*, *Y* e *Z*. Estes 6 dados definem o local e o tamanho de um cubo imaginário na área total da proteína, onde o programa vai tentar fazer o docking, ou seja, o sítio ativo da proteína. Os valores utilizados na análise de teste do Vina2Cluster foram 11 (*Center X*) 90.5 (*Center Y*), 57.5 (*Center Z*), 22 (*Size X*), 24 (*Size Y*) e 28 (*Size Z*). Tanto a proteína quanto os valores foram obtidos no arquivo de amostra do Autodock Vina (Autodock Vina, 2015).

O programa teve sucesso em utilizar o modo cliente e modo servidor para a transferência de dados e análise via Autodock Vina. Foi testado até cinco computadores conectados executando a análise simultaneamente, sem nenhum problema na transferência, levando cerca de 52 minutos pra fazer o docking de todos os ligantes. Em comparação, um só computador levou 170 minutos para a análise da mesma biblioteca com os mesmos dados. Isso mostra que o Vina2Network processou os dados, nesta análise, em 30% o tempo do Autodock Vina rodando em uma só máquina. Apesar de ter-se usado somente cinco máquinas para a análise, teoricamente não há limites para o número de computadores que possam se conectar. O único limitador é o número de ligantes, pois como cada computador só analisa um ligante por vez, o número máximo de computadores que poderiam ser utilizados de forma não-redundante seria igual ao número total de ligantes.

Comparando o Vina2Network com o MOLA (ABREU et al., 2010), outra ferramenta que utiliza o Autodock Vina para *Virtual Screening*, que também usa computadores conectados em rede para a formação de uma estrutura que emula um *cluster*, podem ser notadas algumas diferenças de desempenho. Utilizando 5 computadores com 2 núcleos cada, o MOLA obteve uma velocidade de 4,3x a do Autodock Vina de um total possível de 5x, ou seja, tendo 14% de perda de velocidade no processo. Esta é uma perda pequena se comparada com os resultados do Vina2Network, que mostrou uma perda de 35% de velocidade comparado ao total possível.

4. CONCLUSÕES

O programa Vina2Network é uma ferramenta rápida e eficiente para fazer análises de *Virtual Screening* a partir do Autodock Vina. Sua capacidade de utilizar qualquer computador conectado pela rede ou pela internet que utilize o sistema operacional Linux torna o software uma ferramenta versátil na criação de clusters computacionais. Apesar de não ter a mesma eficiência de outras ferramentas como o MOLA, o Vina2Network possui vantagens como a facilidade de sua aplicação e a versatilidade de poder utilizar computadores conectados via internet.

O próximo passo é o desenvolvimento de uma *pipeline* melhorada para o software, adicionando funções como o preparo dos ligantes e do receptor, assim como implementar um modo gráfico, visando facilitar ainda mais o uso do Vina2Network.

5. REFERÊNCIAS BIBLIOGRÁFICAS

ABREU, R. M. et al. MOLA: a bootable, self-configuring system for virtual screening using AutoDock4/Vina on computer clusters. **Journal of cheminformatics**, v. 2, n. 1, p. 10, jan. 2010.

Autodock Vina. Disponível em: <<http://vina.scripps.edu/tutorial.html>>. Acesso em: 23 maio. 2015.

BELL, A. S. et al. Plate-based diversity subset screening: an efficient paradigm for high throughput screening of a large screening file. **Molecular diversity**, v. 17, n. 2, p. 319–35, maio 2013.

FRIESNER, R. A. et al. Glide: a new approach for rapid, accurate docking and scoring. 1. Method and assessment of docking accuracy. **Journal of medicinal chemistry**, v. 47, n. 7, p. 1739–49, 25 mar. 2004.

JAIN, A. N. Surflex: fully automatic flexible molecular docking using a molecular similarity-based search engine. **Journal of medicinal chemistry**, v. 46, n. 4, p. 499–511, 13 fev. 2003.

JONES, G.; WILLETT, P.; GLEN, R. C. Molecular recognition of receptor sites using a genetic algorithm with a description of desolvation. **Journal of Molecular Biology**, v. 245, n. 1, p. 43–53, jan. 1995.

KITCHEN, D. B. et al. Docking and scoring in virtual screening for drug discovery: methods and applications. **Nature reviews. Drug discovery**, v. 3, n. 11, p. 935–49, nov. 2004.

KUNTZ, I. D. et al. A geometric approach to macromolecule-ligand interactions. **Journal of Molecular Biology**, v. 161, n. 2, p. 269–288, out. 1982.

MCGANN, M. FRED and HYBRID docking performance on standardized datasets. **Journal of computer-aided molecular design**, v. 26, n. 8, p. 897–906, ago. 2012.

MCINNIS, C. Virtual screening strategies in drug discovery. **Current opinion in chemical biology**, v. 11, n. 5, p. 494–502, out. 2007.

MORRIS, G. M. et al. AutoDock4 and AutoDockTools4: Automated docking with selective receptor flexibility. **Journal of computational chemistry**, v. 30, n. 16, p. 2785–91, dez. 2009.

POLGÁR, T.; KESERU, G. M. Integration of virtual and high throughput screening in lead discovery settings. **Combinatorial chemistry & high throughput screening**, v. 14, n. 10, p. 889–97, dez. 2011.

RAREY, M.; KRAMER, B.; LENGAUER, T. Time-efficient docking of flexible ligands into active sites of proteins. **Proceedings / ... International Conference on Intelligent Systems for Molecular Biology ; ISMB. International Conference on Intelligent Systems for Molecular Biology**, v. 3, p. 300–8, jan. 1995.

REDDY, A. S. et al. Virtual screening in drug discovery -- a computational perspective. **Current protein & peptide science**, v. 8, n. 4, p. 329–51, ago. 2007.

TROTT, O.; OLSON, A. J. AutoDock Vina: improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading. **Journal of computational chemistry**, v. 31, n. 2, p. 455–61, 30 jan. 2010.