

MDock: An Ensemble Docking Suite for Molecular Docking, Scoring and In Silico Screening

Chengfei Yan and Xiaoqin Zou

Abstract

Molecular docking refers to computational methods for the prediction of the binding mode and binding affinity between two molecules. Over decades of development, protein–ligand docking methods have been widely used for in silico screening of molecular libraries for drug candidates, serving as a valuable tool in structure-based drug design. MDock is a protein–ligand docking suite originally released from our laboratory in 2007, which incorporates the iteratively derived knowledge-based scoring function and the ensemble docking method. In this chapter, we describe the methodology and usage of MDock for molecular docking and in silico screening. The MDock suite is freely available to academic users through applications at <http://zoulab.dalton.missouri.edu/mdock.htm>.

Keywords: Molecular docking, Scoring function, In silico screening, Binding affinity

1 Introduction

Molecular docking refers to an approach that predicts the binding mode and affinity between two interacting molecules. This approach has been widely applied to protein–ligand binding, protein–protein binding, and protein–nucleic acid binding. Molecular docking is also an important tool for structure-based drug design [1–6]. Given a potential drug target with a known three-dimensional atomic structure, a key step for drug design is to find small molecules that can bind tightly to a specific site on the target and enhance (or inhibit) the function of the target. Due to its high efficiency and low cost, molecular docking is often used for the screening of large chemical libraries for drug candidates. The top-ranked compounds from in silico screen are normally evaluated in biological assays; the confirmed active compounds are advanced for further lead optimization.

One of the examples of molecular docking tools is MDock, a protein–ligand docking suite released by our laboratory in 2007 [7]. MDock docks a rigid ligand to the protein by matching a subgroup of the ligand atomic centers to the sphere points that represent the negative image of the binding pocket, a strategy that was proposed by Kuntz and co-workers [8, 9]. Each docked pose is

scored in combination with local optimization by using ITScore, an iteratively derived knowledge-based scoring function [10–12]. The pose with the lowest score is considered as the predicted binding mode, and the corresponding score is considered as the predicted binding energy score. Specifically, to account for ligand flexibility, multiple low-energy ligand conformers are pre-generated, and each conformer is docked to the protein independently. The docked conformer with the lowest score among all the conformers being docked is set as the predicted binding mode for the ligand, and the corresponding score is the predicted binding energy score.

To account for protein structural variations during ligand binding, MDock also allows users to dock a ligand simultaneously to multiple protein structures (up to 99 structures), a procedure referred to as ensemble docking. The ensemble docking algorithm in MDock is computationally efficient, with a computational time comparable to single protein docking [7, 13].

In this chapter, we will describe the methodology and the usage of MDock for molecular docking and in silico screening in detail. To further illustrate the usage of MDock for in silico screening, we will use the designed ligands for spleen tyrosine kinase (SYK), which were donated by GlaxoSmithKline (GSK) to the Community Structure-Activity Resource (CSAR) 2014 benchmark (<http://www.csardock.org/>) [14–16], for a case study.

2 Materials

2.1 The MDock Package

The MDock suite is freely available to academic users through application at <http://zoulab.dalton.missouri.edu/mdock.htm>. MDock is an open source software written in Fortran. For users who prefer to modify MDock's source code, the Intel Fortran compiler is required for compiling the executables. For users who apply MDock directly to docking studies, the pre-compiled executable files (*MDock*, *clu_sph*, and *get_sph*) are provided in the *bin* directory of the MDock package and are ready for use. Here, *MDock* is the command for docking, *get_sph* is for selecting the sphere points that cover the binding region, and *clu_sph* is for clustering the sphere points from multiple protein structures for ensemble docking. The source codes are placed under the *Source_codes* directory. The documentation and the demo parameter files are under the *Manual* directory. The tutorial files are under the *Tutorial* directory. Linux users can run the programs directly after *MDock* is installed and the *bin* directory is added (e.g., by typing *source Install_MDock* or by adding the path manually). For windows users, a Windows Linux emulator Cygwin (<https://www.cygwin.com/>) is recommended.

2.2 The Overview of the MDock Methodology

MDock implements a similar method to UCSF DOCK [17] to orient the rigid ligand to the binding pocket by exhaustively matching [7] the ligand atomic centers to the sphere points that represent the negative image of the binding pocket. The uniqueness of MDock lies in its iteratively derived scoring function (referred to as ITScore) and its ensemble docking method. MDock is convenient for in silico screening.

2.2.1 The Iterative Scoring Function

ITScore is a knowledge-based scoring function that was originally developed in our laboratory in 2006 [10, 11]. The main idea for the extraction of ITScore is to iteratively adjust the pairwise potentials by comparing the experimentally observed pair distribution functions ($\mathcal{G}_{ij}^{obs}(r)$) derived from the native structures and the predicted pair distribution functions ($\mathcal{G}_{ij}^k(r)$) derived from the sampled decoys (including the native structures), with each decoy carrying a Boltzmann weight calculated from the interaction potentials of the current step. Finally, ($\mathcal{G}_{ij}^k(r)$) converge to ($\mathcal{G}_{ij}^{obs}(r)$) with all the native structures having the lowest energies in comparison with their decoys. The idea is described by the following equations:

$$u_{ij}^{k+1}(r) = u_{ij}^k(r) + \Delta u_{ij}^k(r), \quad \Delta u_{ij}^k(r) = \frac{1}{2} K_B T [\mathcal{G}_{ij}^k(r) - \mathcal{G}_{ij}^{obs}(r)], \quad (1)$$

where i and j represent the atom types of an atom pair, respectively, and r is the distance between the atom pair. K_B is the Boltzmann constant, and T is the temperature. $\{u_{ij}^k(r)\}$ are the pairwise potentials in the k th step, and $\{u_{ij}^{k+1}(r)\}$ are the updated potentials for the next step. Given a set of initial potentials $\{u_{ij}^0(r)\}$, the potentials are updated using the above iterative equation, until $\{\mathcal{G}_{ij}^k(r)\}$ converge to $\{\mathcal{G}_{ij}^{obs}(r)\}$ and all the native structures are associated with the lowest energies compared to their corresponding decoys. The detailed description of the iterative method is provided in [10].

It should be noted that we recently improved the scoring function by using the refined set of PDBbind 2012 [18, 19] as the new training set, which is much larger than the original training set [20]. To reproduce the results of the case study in this chapter, one should use the latest version of MDock (Ver. 2.0).

2.2.2 The Ensemble Docking

MDock implements the ensemble docking method to account for protein structural variations during ligand binding. Specifically, multiple protein structures are superimposed with the protein conformational state treated as an additional dimension for parameter optimization. The energy function for parameter optimization is defined as

$$E = E(x, y, z, \phi, \theta, \psi, n), \quad (2)$$

where x , y , and z stand for the coordinates of the center of mass of the ligand, and ϕ , θ , and ψ stand for the three Euler angles,

respectively. n represents the n th protein structure in the protein ensemble. MDock simultaneously optimizes the ligand coordinates and the protein conformational variable n to automatically select the optimal protein structure that best fits the ligand.

2.2.3 The In Silico Screening

MDock can be easily applied to in silico screening. For a given chemical database, the user is required to prepare a mol2 format file that provides the coordinates and Sybyl atom types of the chemical compounds. The charge and hydrogen information is not needed for MDock. The effect of charges is implicitly considered in the pairwise interaction potentials of MDock through atom types. MDock will then serially dock all the compounds onto the given target protein, predict the binding modes, and rank these compounds according to the predicted binding affinities. The top candidates can be assayed for experimental verification.

2.3 Software Dependencies

Several additional tools are also needed for file preparation for MDock. All these tools are free for academic users.

1. UCSF Chimera [21]:
Chimera is used for preparing the protein and ligand files and for analyzing the docking results. The software can be downloaded directly from the website <http://www.cgl.ucsf.edu/chimera>
2. DMS (optional):
DMS is used for building the molecular surface. DMS can be obtained from the website <http://www.cgl.ucsf.edu/Overview/software.html#dms>. An alternative option is to use the *Write DMS* tool in Chimera.
3. Sphgen_cpp:
Sphgen_cpp is an accessory tool of the UCSF Dock program suite [17]. Sphgen_cpp is used for generating sphere points based on the molecular surface files. Sphgen_cpp can be downloaded from the website http://dock.compbio.ucsf.edu/Contributed_Code/sphgen_cpp.htm.
4. OMEGA (optional) [22, 23]
The OMEGA is a program suite released by OpenEye Scientific Software (Santa Fe, NM, USA, <http://www.eyesopen.com/>). The software is free for academic users. OMEGA is used for generating multiple conformations for a given ligand. The input for OMEGA can be either three-dimensional (3D) structures in pdb format or SMILES strings in smi format. One may also use other programs to sample different ligand conformers.

3 Methods

MDock requires the 3D structure of the protein target (or an ensemble of protein structures), the 3D structure of the ligand, and the file that contains the sphere points which represent the negative image of the binding pocket. The preparation of these files is described as follows:

3.1 Preparation of the Protein and Ligand Files

The structures of the protein and the ligand can be either the experimentally determined or theoretically modeled structures. MDock uses the SYBYL mol2 format files for docking. However, for the preparation of the aforementioned sphere points, the pdb file of the protein structure is also required. The pdb file and mol2 file can be easily converted from one to the other using Chimera. Multiple structures of the ligand or multiple ligands can be stored in a single mol2 file. MDock docks the multiple structures in the ligand mol2 file one by one. For ensemble docking, the multiple structures of the protein need to be superimposed together, which can be done by the *MatchMaker* tool in Chimera. The protein structures for ensemble docking can be NMR models, protein structures bound with different ligands, or conformations sampled by computational techniques such as Molecular Dynamics (MD) or Monte Carlo (MC) simulations.

It is noted that solvent molecules, ions, and other co-bound small molecules should be deleted when preparing the protein structures for docking. MDock does not require the addition of hydrogens and charges for the protein and the ligand. The hydrogen and the charge information in the input files is automatically ignored.

3.2 Generation and Selection of Sphere Points for Docking

It takes the following steps to generate and select sphere points for docking purpose:

1. Generating the molecular surface of the protein structure.
The molecular surface of the protein structure can be generated using the following command:

```
dms protein.pdb -a -n -o protein.ms
```

where *dms* is taken from the molecular surface generation software DMS, *protein.pdb* is the pdb file of the protein structure, and *protein.ms* is the output file which contains the coordinates of the dots representing the molecular surface of the protein. Alternatively, Chimera's Write DMS tool can also be used for sphere point generation.

2. Generating sphere points based on the molecular surface of the protein structure.

The file contains the sphere points for the protein structure can be generated by

```
sphgen_cpp -i protein.ms -o protein.sph
```

where *sphgen_cpp* is the executable of the Sphgen_cpp program in the UCSF Dock software. The output *protein.sph* file contains the coordinates of the sphere points of the protein structure.

3. Defining the putative binding site.

A pdb-format file (denoted as “*site.pdb*”) that locates the putative binding site is required to prepare for the selection of the sphere points that cover the binding region. For a protein structure with known binding pocket, this pdb file can be either the coordinates of the residues close to the center of the binding pocket or the coordinates of the co-crystallized ligand(s). For a protein structure with no prior knowledge of its binding pocket, users can use any binding site prediction tools or servers, such as Q-SiteFinder [24], 3DligandSite [25], and GalaxySite [26], to predict the binding pocket.

4. Selecting the sphere points that adequately cover the binding region.

The sphere points which cover all the binding region can be selected using the following command:

```
get_sph site.pdb protein.sph
```

where *get_sph* is an accessory command in the *bin* directory, which selects all the sphere points within a specified distance (default: 3.0 Å) from the atoms in “*site.pdb*.” The default output files are *recn.sph*, *recn.pdb*, and *sph.par*. *recn.sph* contains the selected sphere points that will be used by MDock in the docking calculations. *recn.pdb* is for the display of the sphere points in *recn.sph* using Chimera. *sph.par* saves the record of the parameters used by *get_sph*.

Users should display *recn.pdb* in Chimera to examine whether the binding region was adequately covered. If not, users should use a larger cutoff for sphere points selection. The cutoff distance for sphere points selection and other parameters for *get_sph* can be specified in two ways:

1. Run *get_sph* interactively:

```
get_sph site.pdb protein.sph -param
```

Users will be asked to provide a value for each parameter. If users decide to use the default value for a parameter, simply hit “Enter.” The parameters will be output in the *sph.par* file.

- Run *get_sph* with parameters defined in a parameter file, say, *sph.par*:

```
get_sph site.pdb protein.sph -param sph.par
```

All the parameters will then be read from this parameter file.

A detailed explanation of the parameters in *get_sph* is provided in the manual of *MDock*.

For molecular docking against a single protein structure, the sphere points in *recn.sph* will be directly used for molecular docking. The preparation is more complicated for molecular docking against multiple protein structures. Specifically, the sphere points should be prepared for each protein structure independently. Then, these sphere points generated from individual protein structures are combined and clustered as follows:

```
cat */recn.sph > all.sph
clu_sph all.sph recn.sph
```

The output file, “*recn.sph*”, comprises the coordinates of the sphere points to be used for ensemble docking.

3.3 Molecular Docking

3.3.1 Single (Protein) Docking

The docking command, *MDock*, can be executed in three ways:

- Run *MDock* using default parameters:

```
MDock protein.mol2 ligand(s).mol2
```

The *ligand(s).mol2* file contains a single ligand conformer or multiple conformers of a ligand or multiple ligands. *MDock* automatically docks all the conformers to the protein. This method requires the sphere point file *recn.sph* as a standard input. The default parameters will be used for the docking calculation, the values of the parameters will be output in a parameter file named *MDock.par*.

- Run *MDock* interactively:

```
MDock protein.mol2 ligand(s).mol2 -param
```

MDock will interactively ask users to provide a value for each parameter. If users prefer the default value, hit “Enter” key. The input values of the parameters will be saved in a parameter file named *MDock.par*.

- Run *MDock* by using the parameters pre-defined in a parameter file, say, *MDock.par*:

```
MDock protein.mol2 ligand(s).mol2 -param MDock.par
```

MDock will search in *MDock.par* for the required docking parameters. If any required parameter is missing, *MDock* will interactively ask the user to specify the value for the parameter.

Besides its application to molecular docking, *MDock* can also be applied to binding mode optimization, scoring, and even target selectivity. For detailed descriptions of the parameters for *MDock*, the user can refer to the *MDock* manual.

3.3.2 Ensemble Docking

For ensemble docking, the *mol2* files for the multiple protein structures should be under the same directory, with their file names sharing a user-defined prefix followed by a double-digit number (from 01 to 99) to label the protein structures. For example, if we have eight protein structures for ensemble docking, we define the prefix as *PKA*, then the eight *mol2* files will be named *PKA01.mol2*, *PKA02.mol2*, ..., and *PKA08.mol2*. The command to run *MDock* against multiple protein structures is

```
MDock PKA ligand(s).mol2
```

Similar to single docking, ensemble docking also requires the sphere point file, "recn.sph", and can be run interactively with user-defined parameters or with the parameters defined in a parameter file.

3.3.3 The Output Files of *MDock*

MDock creates three files: a *mol2* file that lists the docked modes including their coordinates and energy scores (default: *MDock.mol2*), an output file that lists the energy scores of the docked modes (for ensemble docking, the corresponding protein structure number for each docked mode is also included) (default: *MDock.out*), and a file that records the information about the consumed CPU time and the number of processed ligand conformers (default: *MDock.log*).

4 The Case Study

The target SYK with its 276 ligands in the CSAR 2014 benchmark is used for the case study. The CSAR benchmark provides the SMILES strings of the ligands in an *smi*-format file (*SYK_set.smi*). The pIC_{50} values of all the ligands for SYK and the complex structures (in *pdb* format) for eight ligands (GTC000222 to GTC000226, GTC000233, GTC000249 and GTC000250) are also released in the benchmark.

In our docking study, up to 500 conformers for each ligand were generated from its SMILES string using Omega 2.4.6 using the following command:

```
omega2 -in SYK_set.smi -out ligs.mol2 -warts true -fromCT true -strictfrags true -maxconfs 500 -flipper true
```

A total of 108,981 3D conformers for 276 ligands were generated and stored in *mol2* files (*ligs.mol2*). Both single docking and ensemble docking were performed. Specifically, the protein

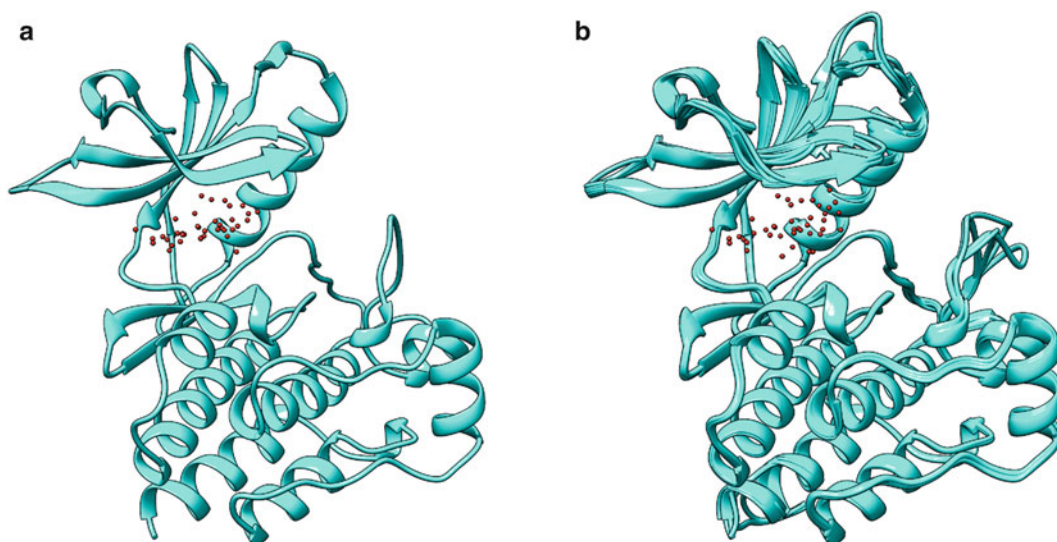


Fig. 1 (a) The protein structure with the sphere points for single (protein) docking. (b) The protein structures with the sphere points for ensemble docking

structure from the SYK-GTC000222 complex was used for single docking. As for ensemble docking, the protein structures from the eight released complexes were superimposed with Chimera, using the protein from the SYK-GTC000222 complex as the reference structure; these eight structures formed the protein conformational ensemble.

The files containing the sphere points for single docking and ensemble docking were prepared as described in Sect. 3.2. For sphere point selection, we used the ligand coordinates (*site.pdb*) from the PDB entry 1XBC [27], which contains SYK bound with a small molecule that binds to the same binding pocket as those 276 ligands. 1XBC was also superimposed to the SYK-GTC000222 complex using Chimera.

Figure 1a, b show typical views of the protein structures and the sphere points being selected for single docking and ensemble docking, respectively. The parameter files used for single docking and ensemble docking are identical and are shown in Fig. 2. For each ligand conformer, up to 1000 poses were rigidly sampled followed by local optimization and scoring. Only the best scored pose for each conformer was saved. The computations were performed using a single 3.40 GHz Intel Core i7 CPU. Single docking took 56,265 s, whereas ensemble docking took 27,103 s. Ensemble docking is more efficient in this case study because by using multiple protein structures, the local optimization was more easily to converge, and because with more protein conformers, ensemble docking requires fewer than 1000 ligand poses to exhaustively sample the possible binding poses.

```

clash_potential_penalty      | 3.0
orient_ligand (yes/no)      | yes
minimize_ligand (yes/no)    | yes
maximum_orientations        | 1000
gridded_score (yes/no)      | yes
grid_spacing (0.3~0.5)      | 0.3
sort_orientations (yes/no)  | yes
write_score_total           | 1
write_orientations (yes/no) | yes
minimization_cycles (1~3)   | 1
ligand_selectivity (yes/no) | no
output_filename_prefix      | mdock_dock
box_filename (optional)     |
grid_box_size                | 8.0
sphere_point_filename       | recn.sph

```

Fig. 2 The parameter file used for single (protein) docking and ensemble docking

For each ligand, the best-scored binding mode (i.e., the mode with the lowest score) among all the docked conformers was considered as the predicted binding mode, and the corresponding score was considered as the predicted binding energy score. The root-mean-square-deviation (RMSD) of the heavy atoms between the predicted binding mode and the native binding mode was used as the metric for evaluating the performance of binding mode prediction, whereas the Pearson correlation coefficient (r) between the predicted binding energy scores and the $-pIC50$ values was used to evaluate the performance of the binding affinity prediction. Table 1 lists the RMSDs of the best predicted binding mode (with the lowest RMSD) for the top 1 prediction and for the top 3 predictions, respectively, for the eight ligands with released complex structures. The results of single docking and ensemble docking are also shown, respectively.

It can be seen from Table 1 that single docking and ensemble docking show comparable performances. An example of successful ensemble docking for binding mode prediction is shown in Fig. 3: The top prediction for the ligand GTC000222 achieved an RMSD of 0.866 Å compared with the native binding mode. For binding affinity prediction, ensemble docking shows a better performance than single docking: In the score versus $-pIC50$ plot shown in Fig. 4, the predicted binding affinities for the 276 ligands with ensemble docking achieved a higher correlation with the $-pIC50$ values (0.72) than the correlation achieved from single docking (0.51).

Other applications of MDock can be found in our publications [13, 20, 28, 29].

Table 1
Results of binding mode prediction on SYK with single (protein) docking and ensemble docking

Ligand	Single docking (Å)		Ensemble docking (Å)	
	Top 1 ^a	Top 3 ^b	Top 1	Top 3
GTC000222	0.864	0.864	0.866	0.866
GTC000223	0.942	0.942	2.124	0.965
GTC000224	3.198	2.555	2.705	2.705
GTC000225	1.318	1.008	1.685	1.032
GTC000226	2.268	2.049	2.172	2.023
GTC000233	1.967	1.310	1.877	0.894
GTC000249	0.656	0.656	1.698	1.698
GTC000250	0.799	0.799	3.422	0.748

^a This column presents the *RMSD* between the native binding mode and the predicted binding mode of the ligand when only the top prediction is considered

^b This column shows the *RMSD* between the native binding mode and best predicted binding mode within the top three predictions

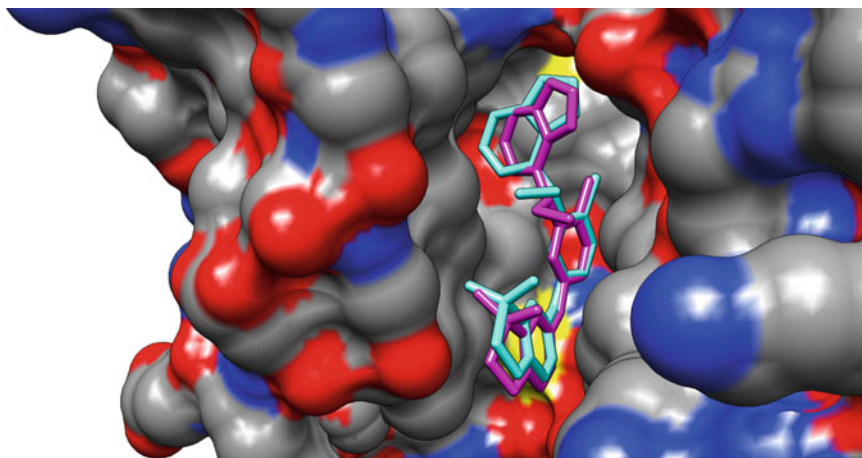


Fig. 3 The binding mode of the ligand GTC000222 was successfully predicted by ensemble docking, with an *RMSD* of 0.866 Å from the native binding mode. The protein is represented by its molecular surface. The native ligand is in *cyan*, and the predicted binding mode is in *magenta*

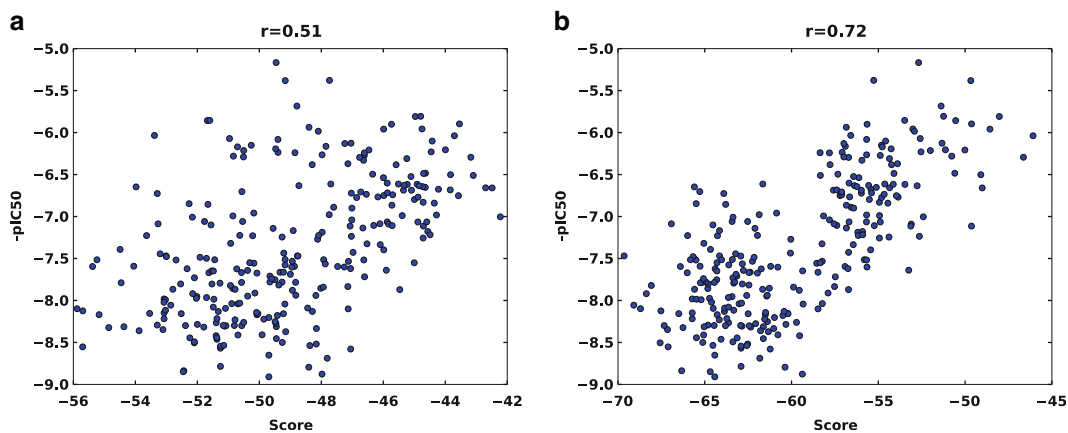


Fig. 4 (a) The score versus $-pIC_{50}$ plot for single (protein) docking. (b) The score versus $-pIC_{50}$ plot for ensemble docking

5 Notes

1. For the preparation of the protein structures for docking, solvent molecules, ions and co-bound small molecules should be removed.
2. MDock does not need to add hydrogens and charges for the protein and the ligand. The information about hydrogens and charges in the input protein and ligand files is automatically ignored.
3. For ensemble docking, the multiple protein structures should be superimposed for preparing the sphere points and docking calculation.
4. For sphere point selection, users should manually examine the sphere points to make sure that the whole binding region is adequately covered by the sphere points. Otherwise, the value of the cutoff distance should be increased to include more sphere points.

Acknowledgements

Xiaoqin Zou is supported by NIH grant R01GM088517, NSF CAREER Award DBI0953839, and American Heart Association (Midwest Affiliate) 13GRNT16990076. The computations were performed on the HPC resources at the University of Missouri Bioinformatics Consortium (UMBC).

References

1. Brooijmans N, Kuntz ID (2003) Molecular recognition and docking algorithms. *Annu Rev Biophys Biomol Struct* 32(1):335–373
2. Kitchen DB, Decornez H, Furr JR, Bajorath J (2004) Docking and scoring in virtual screening for drug discovery: methods and applications. *Nat Rev Drug Discov* 3(11):935–949
3. Huang S-Y, Zou X (2010) Advances and challenges in protein-ligand docking. *Int J Mol Sci* 11(8):3016–3034
4. Huang S-Y, Grinter SZ, Zou X (2010) Scoring functions and their evaluation methods for protein–ligand docking: recent advances and future directions. *Phys Chem Chem Phys* 12(40):12899–12908
5. Sousa SF, Ribeiro AJM, Coimbra JTS, Neves RPP, Martins SA, Moorthy NSHN, Fernandes PA, Ramos MJ (2013) Protein-ligand docking in the new millennium – a retrospective of 10 years in the field. *Curr Med Chem* 20(18):2296–2314
6. Grinter SZ, Zou X (2014) Challenges, applications, and recent advances of protein-ligand docking in structure-based drug design. *Molecules* 19(7):10150–10176
7. Huang S-Y, Zou X (2007) Ensemble docking of multiple protein structures: considering protein structural variations in molecular docking. *Proteins Struct Funct Bioinf* 66(2):399–421
8. Kuntz ID, Blaney JM, Oatley SJ, Langridge R, Ferrin TE (1982) A geometric approach to macromolecule-ligand interactions. *J Mol Biol* 161(2):269–288
9. Ewing TJA, Kuntz ID (1997) Critical evaluation of search algorithms for automated molecular docking and database screening. *J Comput Chem* 18(9):1175–1189
10. Huang S-Y, Zou X (2006) An iterative knowledge-based scoring function to predict protein–ligand interactions: I. Derivation of interaction potentials. *J Comput Chem* 27(15):1866–1875
11. Huang S-Y, Zou X (2006) An iterative knowledge-based scoring function to predict protein–ligand interactions: II. Validation of the scoring function. *J Comput Chem* 27(15):1876–1882
12. Huang S-Y, Zou X (2011) Scoring and lessons learned with the CSAR benchmark using an improved iterative knowledge-based scoring function. *J Chem Inf Model* 51(9):2097–2106
13. Huang S-Y, Zou X (2007) Efficient molecular docking of NMR structures: application to HIV-1 protease. *Protein Sci* 16(1):43–51
14. Dunbar JB Jr, Smith RD, Yang C-Y, Ung PM-U, Lexa KW, Khazanov NA, Stuckey JA, Wang S, Carlson HA (2011) CSAR benchmark exercise of 2010: selection of the protein–ligand complexes. *J Chem Inf Model* 51(9):2036–2046
15. Damm-Ganamet KL, Smith RD, Dunbar JB Jr, Stuckey JA, Carlson HA (2013) CSAR benchmark exercise 2011–2012: evaluation of results from docking and relative ranking of blinded congeneric series. *J Chem Inf Model* 53(8):1853–1870
16. Dunbar JB Jr, Smith RD, Damm-Ganamet KL, Ahmed A, Esposito EX, Delproposito J, Chinnaswamy K, Kang Y-N, Kubish G, Gestwicki JE, Stuckey JA, Carlson HA (2013) CSAR data set release 2012: ligands, affinities, complexes, and docking decoys. *J Chem Inf Model* 53(8):1842–1852
17. Moustakas DT, Lang PT, Pegg S, Pettersen E, Kuntz ID, Brooijmans N, Rizzo RC (2006) Development and validation of a modular, extensible docking program: Dock 5. *J Comput Aided Mol Des* 20(10–11):601–619
18. Wang R, Fang X, Lu Y, Yang C-Y, Wang S (2005) The PDBbind database: methodologies and updates. *J Med Chem* 48(12):4111–4119
19. Cheng T, Li X, Li Y, Liu Z, Wang R (2009) Comparative assessment of scoring functions on a diverse test set. *J Chem Inf Model* 49(4):1079–1093
20. Yan C, Grinter SZ, Merideth BR, Ma Z, Zou X (2015) Iterative knowledge-based scoring functions derived from rigid and flexible decoy structures: evaluation with the 2013 and 2014 CSAR benchmarks. *J Chem Inf Model*. doi:10.1021/acs.jcim.5b00504
21. Pettersen EF, Goddard TD, Huang CC, Couch GS, Greenblatt DM, Meng EC, Ferrin TE (2004) UCSF chimera—a visualization system for exploratory research and analysis. *J Comput Chem* 25(13):1605–1612
22. Hawkins PCD, Skillman AG, Warren GL, Ellingson BA, Stahl MT (2010) Conformer generation with omega: algorithm and validation using high quality structures from the protein databank and Cambridge structural database. *J Chem Inf Model* 50(4):572–584
23. Hawkins PCD, Nicholls A (2012) Conformer generation with omega: learning from the data set and the analysis of failures. *J Chem Inf Model* 52(11):2919–2936
24. Laurie ATR, Jackson RM (2005) Q-sitefinder: an energy-based method for the prediction of

- protein–ligand binding sites. *Bioinformatics* 21 (9):1908–1916
25. Wass MN, Kelley LA, Sternberg MJE (2010) 3DLigandSite: predicting ligand-binding sites using similar structures. *Nucleic Acids Res.* doi:10.1093/nar/gkq406
 26. Heo L, Shin W-H, Lee MS, Seok C (2014) Galaxysite: ligand-binding-site prediction by using molecular docking. *Nucleic Acids Res* 42(W1):W210–W214
 27. Atwell S, Adams JM, Badger J, Buchanan MD, Feil IK, Froning KJ, Gao X, Hendle J, Keegan K, Leon BC et al (2004) A novel mode of gleevec binding is revealed by the structure of spleen tyrosine kinase. *J Biol Chem* 279 (53):55827–55832
 28. Grinter SZ, Liang Y, Huang S-Y, Hyder SM, Zou X (2011) An inverse docking approach for identifying new potential anti-cancer targets. *J Mol Graph Model* 29(6):795–799
 29. Grinter SZ, Yan C, Huang S-Y, Jiang L, Zou X (2013) Automated large-scale file preparation, docking, and scoring: evaluation of ITScore and STScore using the 2012 community structure-activity resource benchmark. *J Chem Inf Model* 53(8):1905–1914