

Chapter 2

Computer-Aided Drug Discovery and Development

Shuxing Zhang

Abstract

Computer-aided approaches have been widely used in pharmaceutical research to improve the efficiency of the drug discovery and development pipeline. To identify and design small molecules as clinically effective therapeutics, various computational methods have been evaluated as promising strategies, depending on the purpose and systems of interest. Both ligand and structure-based drug design approaches are powerful technologies, which can be applied to virtual screening for lead identification and optimization. Here, we review the progress in this field and summarize the application of some new technologies we developed. These state-of-the-art tools have been used for the discovery and development of active agents for various diseases, in particular for cancer therapies. The described protocols are appropriate for all drug discovery stages, but expertise is still needed to perform the studies based on the targets of interest.

Key words: Computer-aided drug discovery, High-throughput screening, Ligand-based drug design, Molecular docking, Quantitative structure–activity relationship, Structure-based drug design, Virtual screening

1. Introduction

Drug discovery and development is a time-consuming and expensive process. On average, it takes 10–15 years and US \$500–800 million to introduce a drug into the market (1, 2). This is why computer-assisted drug design (CADD) approaches have been widely used in the pharmaceutical industry to accelerate the process (3, 4). CADD helps scientists focus on the most promising compounds so that they can minimize the synthetic and biological testing efforts. In practice, the choice of CADD approaches to be employed is usually determined by the availability of the experimentally determined 3D structures of target proteins. If protein structures are unknown, various methods of ligand-based drug

design can be employed, such as Quantitative Structure Activity Relationship (QSAR) and pharmacophore analysis. If the target structures are known, structure-based approaches can be used such as molecular docking, which employs the target 3D structures to design novel active compounds with improved potency. As more structures are becoming available, the prediction accuracy will likely to be improved.

1.1. Ligand-Based Drug Design

In the absence of the receptor 3D information, lead identification and optimization depend on available pharmacologically relevant agents and their bioactivities (5–8). The computational approaches include QSAR, pharmacophore modeling, and database mining (8–10). We will use QSAR as an example to illustrate the workflow. QSAR describes mathematical relationships between structural attributes and target properties of a set of chemicals (9, 11, 12). Many different 2D (two-dimensional) and 3D (three-dimensional) QSAR approaches have been developed during the past several decades (13, 14). The major differences of these methods include chemical descriptors and mathematical approaches that are used to establish the correlation between the target properties and the descriptors.

Many 2D QSAR methods have been extensively studied (15, 16) based on graph theoretic indices. Although the physicochemical meaning of these indices is unclear, they certainly represent different aspects of molecular structures. It has been extensively applied to analytical chemistry, toxicity analysis, and other biological activity prediction (17). 3D QSAR approaches have been developed to address the problems of 2D QSAR techniques including their inability to distinguish stereoisomers. These 3D methods include molecular shape analysis (MSA), distance geometry, and Voronoi techniques (18–20). Comparative Molecular Field Analysis (CoMFA) (21) perhaps is the most popular example of 3D QSAR. It has been widely used in medicinal chemistry and toxicity analysis by elegantly combining the power of molecular graphics and partial least square (PLS) technique. QSAR techniques usually assume the linear relationship between a target property and molecular descriptors. However, the explosive growth of structural and biological data has challenged this assumption. To this end, some nonlinear QSAR methods have been proposed and most of them are based on either artificial neural network (ANN) (22–25) or machine learning techniques (26–28). We have concentrated on the development and application of automated algorithms for QSAR studies, including genetic algorithms-partial least squares, k-Nearest Neighbor (29), and support vector machine (30–32).

Machine learning usually is defined as a discipline concerned with the improvement of the performance of computer algorithms based on their previous experiences (33), and these algorithms establish the correlations between the variables and the output of the system (26, 34–37). In engineering field, it is closely related to pattern recognition and has become steadily more successful

over the past 20 years. Learning approaches have been widely used in cheminformatics and molecular modeling (38–42). For instance, support vector machine (SVM) was found to yield better results compared to multiple linear regressions (MLR) and radial basis functions (RBF) (31). Various versions of such programs have been applied to the calculation of activities of enzyme inhibitors (43). Lazy learning and kNN approaches were employed in the discovery of anticonvulsant compounds and anticancer agents, respectively (41, 42). Machine learning is also being frequently used to conduct ADMET predictions (44–46). Gaussian kernel SVM was used to successfully classify a set of drugs in terms of their potential to cause an adverse drug reaction TdP (47). Although TdP is involved in multiple mechanisms, the SVM prediction accuracy on an independent set of molecules was 90% more than that with ANN and decision tree methods.

1.2. Structure-Based Drug Design

Structure-based design has played an important role in drug discovery and development (48–50). This approach requires the understanding of receptor–ligand interactions. If the target 3D structure is known, it can be used for the design of new ligands (49–51). The structural information is either from X-ray crystallography, NMR, or from homology modeling. SBDD approaches are responsible for evaluating the complementarities and predicting the possible binding modes and affinities between small molecules and their macromolecular receptors. The success of SBDD is well documented (52, 53) and the computational approaches vary widely in methodology, performance, and speed. Some are capable of providing accurate binding modes, while others are more suitable for fast searching of large databases (50, 54–61). Herein we will focus on the most commonly used strategies: molecular docking and scoring.

Molecular docking is used for computational schemes that attempt to find the best matches between a receptor and a ligand. It involves the prediction of ligand conformations and orientation (or posing) within a binding site and attempts to place the ligand into the binding site in configurations and conformations appropriate for interacting with the receptor (62). Docking methods can be divided into matching and simulation methods. The former approaches create a binding site model, typically including the favorable hydrogen binding and steric interactions, and then attempt to dock a ligand into this model by geometrical matching (63). Although early attempts of matching methods only considered the translational and orientational degrees of freedom of the ligand, most of recently developed programs take into account the conformational flexibility of ligands and the limited flexibility of the receptor (63, 64). The examples of this class include DOCK (65, 66), FlexX (67, 68), etc. Simulation methods put a molecule into a binding site by exploring the translations, orientations, and conformations until an ideal binding mode is found. Autodock is the most representative example of this class (69).

One of the major challenges is the scoring function problem, i.e., the problem of fast and accurate evaluation of binding affinities. Several approaches to address this problem have been proposed and developed. Force field scoring is based on the classical molecular force fields, such as AMBER (70), CHARMM (71), MMFF94 (72), etc., to compute nonbonded interaction terms between the receptor and ligand atoms. Additional empirical terms taking into account the solvation and entropy effects have been also considered (73). The second family of methods is the empirical scoring functions, which include LUDI (74–76) and VALIDATE (77). They have been introduced several years ago and are based on the concept that the receptor–ligand interaction energy can be approximated by a multivariate regression of different parameters such as the number of hydrogen bonds, lipophilicity, ionic interactions, entropy penalties, etc. Recently, a third family of methods, knowledge-based scoring functions (DrugScore (78) SmoG (79), PMF (80), BLEEP (81), etc.) has been introduced. These methods employ the statistical analysis of known receptor–ligand complexes to define pairwise interatomic potentials of protein–ligand interactions. After the calibration on the training set of complexes, these scoring functions are validated by predicting binding affinities for the complexes of the test sets.

Recently advances in networking, high-end computers, large data stores, and middleware capabilities are ushering in a new era of high-performance parallel and distributed simulations (82). Based on these technologies, novel high-throughput docking approaches have been developed to enable efficient and inexpensive drug discovery. For instance, we developed an automated DOcking-based VIRTUAL Screening (DOVIS) system (59), which makes sophisticated docking strategies to be carried out on HPC clusters to screen millions of compounds more efficiently. In this chapter, we will discuss the methods using our recent implementation HiPCDock (61).

2. Materials (Hardware and Software)

2.1. Ligand-Based Design Approaches

1. Computer workstations with Linux operating systems.
2. ChemDraw or other molecular structure drawing programs.
3. Descriptor generators such as DRAGON, MolConnZ, and OEChem.
4. Text editors such as UltraEdit, vi, and EMACS.
5. Descriptors normalization programs.
6. Data splitting program such as SE8.
7. Databases such as ZINC, PubChem, and ChemDiv.
8. ALL-QSAR Program.
9. Activity testing facility.

2.2. Structure-Based Design Approaches

1. High-performance computing (HPC) clusters with Linux operating systems.
2. LSF queuing systems for the HPC clusters.
3. Java environment.
4. Perl and Python modules.
5. HiPCDock program.
6. AutoDock3 program.
7. AutoDockTools package.
8. Matlab package.
9. OpenBabel program.

3. Methods

3.1. Ligand-Based Design Approaches

Different programs have different protocols to perform the task. Here we use ALL-QSAR as an example to demonstrate the procedures (Fig. 1). For special notes, please refer Section 4.

1. Prepare molecular structures of interest. Most of programs accept sdf, mol2, or some other formats. The structures can be 2D or 3D, depending on the studies. If 3D QSAR (Fig. 2) is conducted, then 3D structures of the molecules are required.
2. Calculate descriptors for the molecules using the molecular file created above. Most descriptor generation programs just need the molecular file as input. Depending on the study, some options may need to be specified. For instance, with MOE descriptors, you can specify 1D, 2D, or 3D descriptors, or all of them. Other commonly used descriptors include DRAGON descriptors, MolConnZ descriptors, OEChem shape descriptors, etc. During this step, some molecules may be skipped due to the inability to calculate the descriptors by the programs. This can be due to the limitations of the programs or the inaccuracy of the molecule structures.
3. Prepare the molecular activity file. Usually two columns are required: the first column is the molecular name and the second

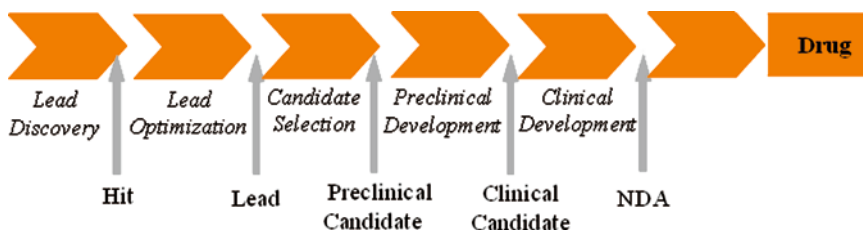


Fig. 1. Drug discovery and development pipeline. The process is very time consuming and expensive. It mainly includes lead discovery and optimization, preclinical and clinical trial, and final NDA and marketing.

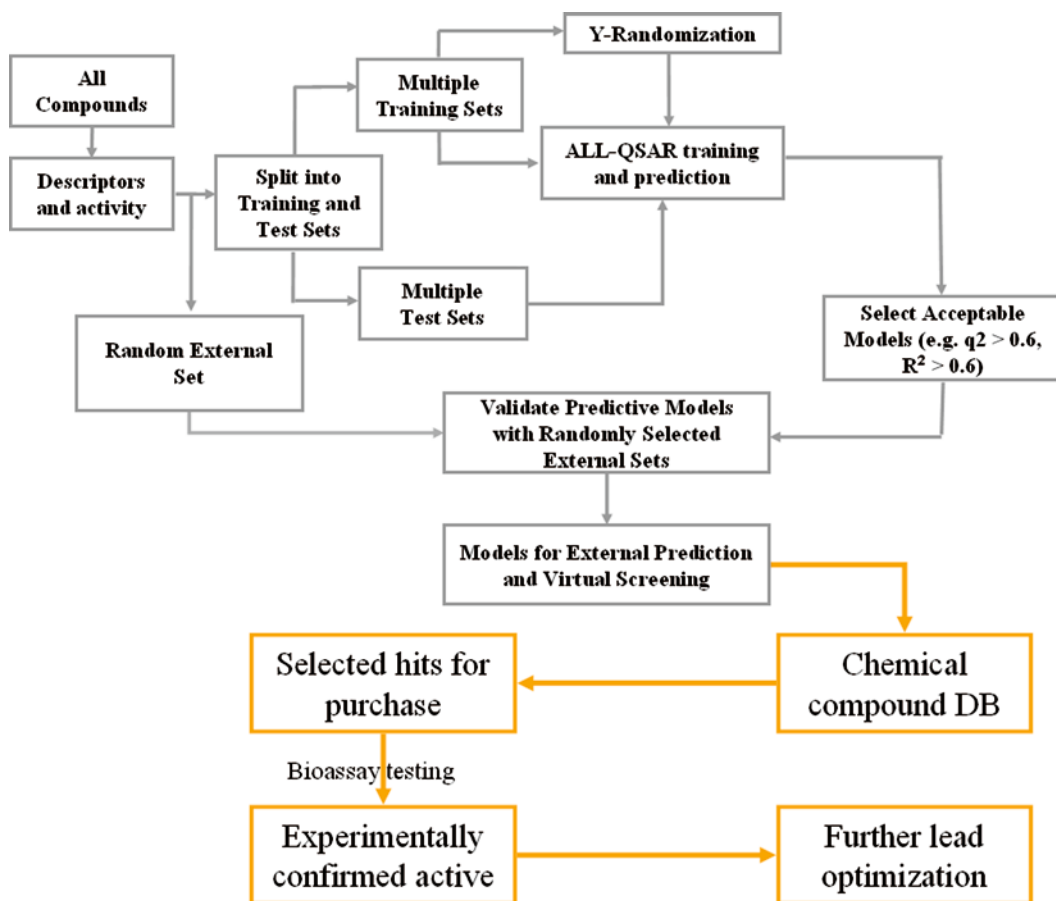


Fig. 2. QSAR and virtual screening workflow. Starting from molecular structures and their bioactivities, QSAR models can be built to perform external predictions and virtual screening. The identified hits are experimentally tested, and the active compounds will be further optimized and the new data can be further fed back to the process.

is the activity values. The order of the molecules in the activity file should be corresponding to the order in the descriptor file.

4. Descriptor normalization. Usually the values of the descriptors are quite different as many of them represent completely different properties with different scale. For instance, the molecular weight is in hundreds but the LogP is usually below 10. To exclude the disproportional influence by the descriptor values, normalization is recommended. The way to do this is to find the maximum and minimum values for all molecules for each descriptor and normalize each values with $(X_i - X_{\min}) / (X_{\max} - X_{\min})$. There are many other ways to do the normalization.
5. Once the descriptors are normalized, the dataset will be split into multiple training and test sets for model building. This can be achieved with SE8 (Sphere Exclusion version 8) algorithm (83).

6. The procedures of SE8 start with the calculation of the distance matrix \mathbf{D} between representative points in the descriptor space. Let D_{\min} and D_{\max} be the minimum and maximum elements of \mathbf{D} , respectively. N probe sphere radii are defined by the following formulas $R_{\min} = R_1 = D_{\min}$, $R_{\max} = R_N = D_{\max}/4$, $R_i = R_1 + (i-1) \times (R_N - R_1) / (N-1)$, where $i = 2, \dots, N-1$.
7. Each probe sphere radius corresponds to one division into the training and test set. In our studies it consisted of the following steps. (i) Select randomly a compound. (ii) Include it in the training set. (iii) Construct a probe sphere around this compound. (iv) Select compounds from this sphere and include them alternatively into test and training sets. (v) Exclude all compounds from within this sphere for further consideration. (vi) If no more compounds left, stop. Otherwise let m be the number of probe spheres constructed and n be the number of remaining compounds. Let d_{ij} ($i = 1, \dots, m$; $j = 1, \dots, n$) be the distances between the remaining compounds and probe sphere centers. Select a compound corresponding to the lowest d_{ij} value and go to step (ii).
8. Once the dataset is split into training and test sets, our ALL-QSAR will load the training set descriptors and activities space into memory and assign a lowest predefined value to the kernel width K .
9. Take a query compound from the test set and calculate the Euclidean distances between it and all compounds of the training set. If the distance from the test set compound to its nearest neighbor is higher than D_{\max} , this compound is out of the applicability domain. Since the activity prediction for it is believed to be not accurate, it will not be predicted. In this case return to step 9 and process the next compound of the test set, or, if there are no more compounds in the test set, go to step 16. If the compound of the test set is within the applicability domain, go to step 10. The applicability domain is calculated as the following: $APD = \hat{y} + Z\sigma$, where \hat{y} is the average of weighted Euclidean distance for the training set. Z is an empirical cutoff value to control the significance (or confidence) level with the default value as 0.5. σ is the standard deviation of all pairwise Euclidean distance in the training set.
10. The weight of every compound in the training set is calculated for the query compound.
11. Calculate coefficients β .
12. Using the values of β , weights and descriptors to predict the target property of the query compound.
13. Repeat step 9 for the next compound. If the procedure was repeated for all compounds, go to step 14.

14. Calculate the correlation coefficient between the predicted and experimental activity values of the test set compounds.
15. If kernel width is lower than the predefined value, add a predefined step to it and repeat the process starting from step 9 for N times until the prediction is converged.
16. Sort models by the R^2 starting from the highest value, and RMSD between predicted and actual target property values and select the top 10 best models.
17. The models can be used for predictions of new molecules or virtual screening.
18. In virtual screening, top hits based on predicted activity (e.g., top 100) are selected for investigation.
19. The selected hits will be inspected for their scaffold, potential toxicity, and other properties by both modelers and synthetic chemists.
20. Only those accepted by both modelers and chemists are submitted to experimental testing.
21. Experimentally confirmed hits will be used to perform lead optimization and new molecule design.
22. The newly designed molecule will be predicted for their activity, starting from step 1, and the measured activities are fed back to our model building process.

3.2. Structure-Based Design Approaches

Here by splitting it into multiple steps including preprocessing of molecules, parallel docking, and postprocessing of result analysis, the overall workflow in Fig. 3 demonstrates how our new HiPCDock (61) works as high-throughput molecular docking protocols for drug discovery and development. For special notes, please refer Section 4.

3.2.1. Preprocessing of Receptors

1. The protein structure is directly downloaded from PDB (pdb format). Hydrogens are added to the structures and appropriate charges are assigned by executing a Python script, which uses the related functions from AutoDockTool (Fig. 4).
2. The structure is converted from pdb format to pdbq format by adding an extra column of charges. If users prefer adding hydrogens and assigning charges with other software, such as SYBYL, it can be done and the saved mol2 file can be used as input into our pipeline.
3. Once our program loads the structure, it uses AutoDock utility program *mol2fftopdbq* to convert the mol2 file to pdbq format.
4. After the pdbq file is obtained, the solvation process is performed using *addsol* module from AutoDock to convert the pdbq file to pdbqs format. The current acceptable input file formats include pdb, pdbq, and mol2.

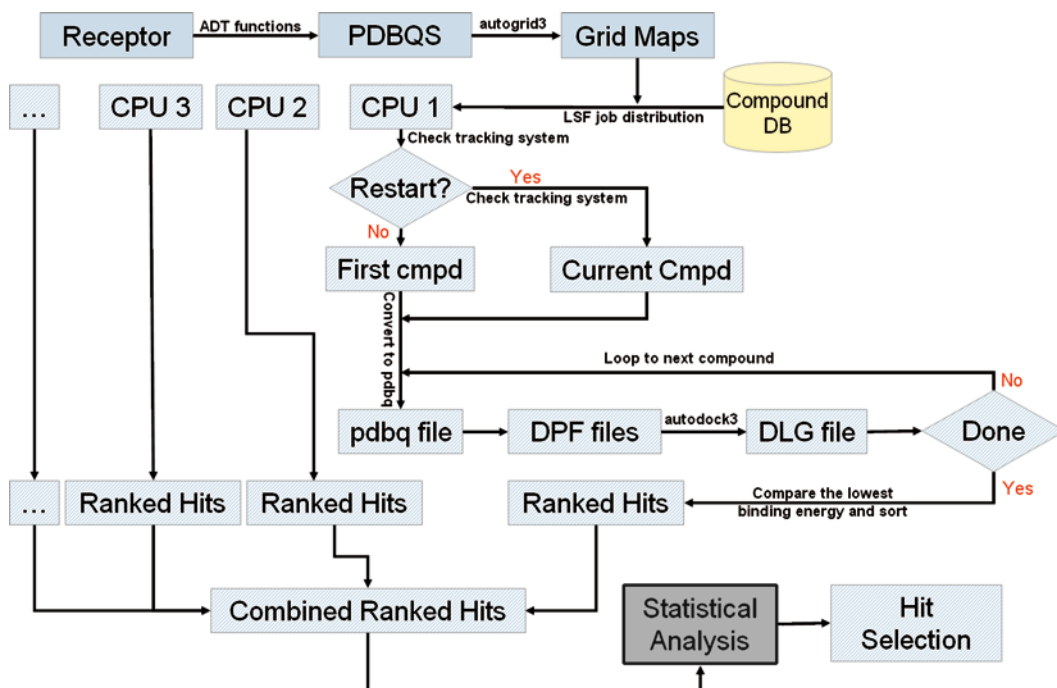


Fig. 3. Overall workflow of HiPCDock. The whole process includes target and ligand preparation, distribution of parallel docking onto multiple CPUs, and final analysis of the results to select promising hits. This is involved in a statistical analysis (77) of docking scores as indicated in the *black box*.

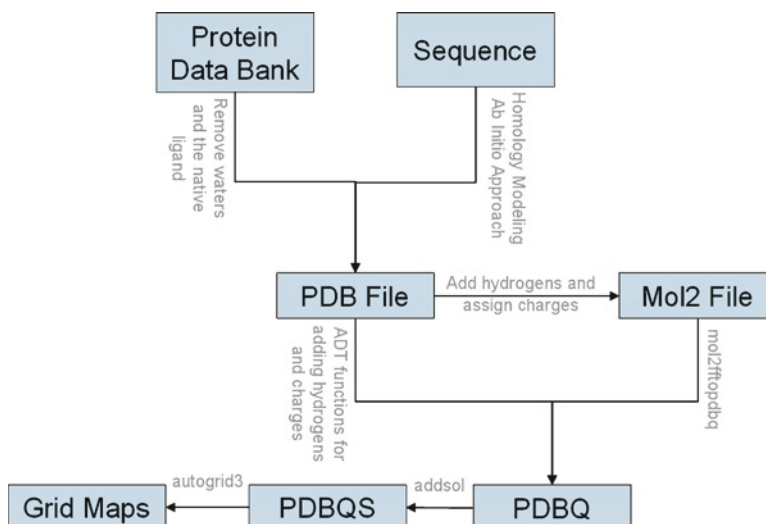


Fig. 4. Workflow for protein preparation. Target preparation starts from the 3D structure of the proteins. Hydrogens and charges are automatically added and grid maps are created using autogrid3, and the pdbqs file is generated for docking process.

5. After the solvation, the created pdbqs is used to generate a grid parameter file (GPF). Currently, ten atom types are used for proteins, including carbon (C), nitrogen (N), oxygen (O), sulfur (S), hydrogen (H), metal (M), phosphorus (P), Zinc (Z), Calcium (L), and X for unknown type.
6. The 3D grids of interaction energy for all possible atom types are calculated at one time. These uniquely defined atom types include nonaromatic carbon (C), aromatic carbon (A), nitrogen (N), oxygen (O), sulfur (S), phosphorus (P), hydrogen (H), metal (M), fluorine (F), chlorine (c), bromine (b), iodine (I), zinc (Z), calcium (L), iron (f), and unknown type (X). They basically cover most of the possible ligand atom types included in databases.
7. The center of the common grids can be either the center of mass coordinates of the ligand that had been removed from the binding site of the target protein under consideration or the geometrical center of a series of key residues provided by users.
8. A modified script (from *gpfgen*) is used to generate the GPF file with our customized atom types and the parameter values provided by users.
9. Based on the GPF file, *autogrid3* is executed to create 16 atom type maps, plus an additional electron density map.

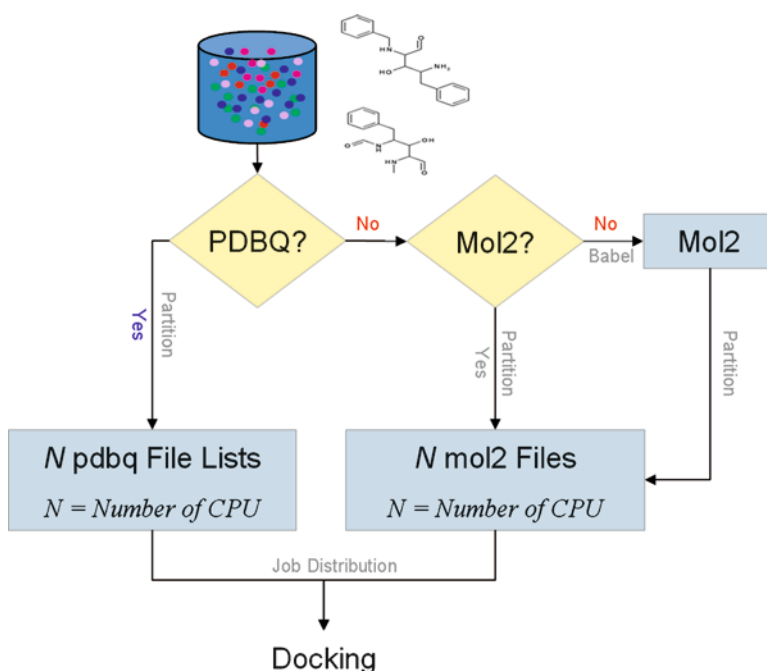


Fig. 5. Workflow for chemical compound preparation. Chemical compound preparation is also performed on multiple CPUs to speed up the process as the chemical database can include millions of compounds. The process will result in ligand pdbq files, which can be repeatedly used in docking.

3.2.2. Preprocessing of Ligand Compounds

The chemical compounds also need to be preprocessed for our program as demonstrated in Fig. 5. The current acceptable ligand input file formats are SMILE strings (smi), sdf/sd, mol, mol2, and pdbq. Files in the first four formats may have single or multiple molecule(s). Each pdbq file can comprise one molecule. The input of chemical compounds is a directory, so it can include multiple files. The pipeline requires that the directory contains either pdbq or acceptable non-pdbq files. The protocol is as follows:

1. HiPCDock converts each non-mol2 file using OpenBabel to mol2 format, which possesses Gasteiger charges.
2. Then each mol2 file is partitioned into multiple pieces of roughly equal size. The number of pieces is determined by the number of CPU requested by users.
3. If the directory has pdbq files, HiPCDock generates multiple (equals to the number of CPU) file lists. Each list includes approximately equal number of compounds.

3.2.3. Parallel Docking on HPC Clusters

Once the grid maps are generated and the chemical compounds are partitioned into the right format, parallel docking can be performed using high-performance computing clusters for virtual screening. Here are the procedures:

1. Each partition is submitted to a CPU and the docking process is performed automatically. The current implementation is using Load Sharing Facility (LSF) queuing system.
2. The LSF job array function is used for the job distribution and scheduling. Once a CPU is available, HiPCDock distributes a job on that CPU and starts docking. Otherwise, it is pending in the queue.
3. The workflow on each CPU is illustrated in Fig. 4. If the chemical database is in mol2 format, *autotors* is executed for each molecule (by looping through all of the molecules in the partition) to define the torsions of the compound and then convert it to pdbq format.
4. The new pdbq file will be saved in a directory so that it can be re-used directly in future runs.
5. If the input is already in pdbq format or the built-in database is used, the above process is skipped and the docking starts to run by executing *autodock3* module.
6. The docking parameter file (DPF) is automatically generated by HiPCDock based on the input from users.
7. After docking each molecule, the result in its docking log file (DLG) is analyzed and the lowest estimated free energy of binding is recorded. This is used for the comparison with other molecules to determine whether this molecule is a strong enough binder to be a hit. If yes, its DLG is kept, otherwise its related files will be deleted in order to save disk space.

8. Since this is the most time-consuming part with a big loop (e.g., tens of thousands of compounds on each CPU if we dock millions of compounds on hundreds of CPUs), a restart function is implemented to improve the robustness of the program. Basically, each successfully processed molecule is recorded in a tracking file. Every time HiPCDock runs, it first checks the tracking file and starts from the molecule where the last run was stopped.

3.2.4. Postprocess of Docked Results

Once the jobs on all CPUs are done, the HiPCDock postprocess module starts to analyze the results.

1. It collects all of the individual hit lists together and generates an overall list.
2. The list is sorted according to their free energy of binding, and the top ranked compounds, as requested by users (e.g., 10% of all database compounds), are selected as the final hits.
3. These hits can be further refined by chemists' knowledge as well as by molecular visualization provided by HiPCDock.
4. For each final hit, all of docked conformations are extracted from the DLG and are converted to sdf files so that the users can visualize their interactions with the receptor.
5. OpenBabel function is utilized to calculate some molecular properties for each hit.

4. Notes

1. The molecular descriptors should be normalized to exclude the influence of those disproportional descriptor values.
2. The descriptors used in training, testing, or new datasets should be consistent.
3. Usually descriptor correlation analysis should be conducted to keep only independent descriptors, also for the reduction of descriptor dimensions.
4. Parameter tuning is usually necessary to obtain best models for predictions, and therefore, it might be a good idea to run the model building multiple times by changing the parameters.
5. Y randomization should be performed to exclude chance correlation between the descriptors and target properties.
6. Databases for virtual screening should be cleaned and their descriptors should be normalized based on training set normalization parameters.

7. Receptors structures should be cleaned by removing its water molecules and by fixing the wrong or missing residues for docking.
8. Different charge types can be tried during docking.
9. The starting conformations of ligands (or chemical databases) should be minimized by using the lowest energy conformations.
10. If the docking process is disrupted, it can be restarted and the docking will continue until finished.
11. The hit selection is based on the conformation with the lowest predicted binding free energies but not necessarily the best binding poses due to the approximation and imperfection of scoring functions.
12. Multiple scoring functions can be applied to conduct consensus docking/scoring to obtain the best predictions.

Acknowledgments

We are grateful to John Morrow for proofreading this chapter, and this work was supported in part by MD Anderson faculty startup fund.

References

1. Workman, P. (2003). How much gets there and what does it do?: The need for better pharmacokinetic and pharmacodynamic endpoints in contemporary drug discovery and development. *Curr Pharm Des.* **9**: 891–902.
2. Brown, D. & Superti-Furga, G. (2003). Rediscovering the sweet spot in drug discovery. *Drug Discov Today.* **8**: 1067–1077.
3. Gomeni, R., Bani, M., D'Angeli, C., Corsi, M. & Bye, A. (2001). Computer-assisted drug development (CADD): an emerging technology for designing first-time-in-man and proof-of-concept studies from preclinical experiments. *Eur J Pharm Sci.* **13**: 261–270.
4. Veselovsky, A. V. & Ivanov, A. S. (2003). Strategy of computer-aided drug design. *Curr Drug Targets Infect Disord.* **3**: 33–40.
5. Stahura, F. L. & Bajorath, J. (2004). Virtual screening methods that complement HTS. *Comb Chem High Throughput Screen.* **7**: 259–269.
6. Guner, O., Clement, O. & Kurogi, Y. (2004). Pharmacophore modeling and three dimensional database searching for drug design using catalyst: Recent advances. *Curr Med Chem.* **11**: 2991–3005.
7. Hansch, C., Leo, A., Mekapati, S. B. & Kurup, A. (2004). Qsar and Adme. *Bioorg Med Chem.* **12**: 3391–3400.
8. Parvu, L. (2003). QSAR – a piece of drug design. *J Cell Mol Med.* **7**: 333–335.
9. Langer, T. & Wolber, G. (2004). Virtual combinatorial chemistry and in silico screening: Efficient tools for lead structure discovery? *Pure App Chem.* **76**: 991–996.
10. Dror, O., Shulman-Peleg, A., Nussinov, R. & Wolfson, H. J. (2004). Predicting molecular interactions in silico: I. A guide to pharmacophore identification and its applications to drug design. *Curr Med Chem.* **11**: 71–90.
11. Perkins, R., Fang, H., Tong, W. D. & Welsh, W. J. (2003). Quantitative structure-activity relationship methods: Perspectives on drug discovery and toxicology. *Environ Toxicol Chem.* **22**: 1666–1679.
12. Tropsha, A. & Zhang, W. F. (2001). Identification of the descriptor pharmacophores using variable selection QSAR: Applications to database mining. *Curr Pharm Design.* **7**: 599–612.
13. Leo, A. J. & Hansch, C. (1999). Role of hydrophobic effects in mechanistic QSAR. *Perspect Drug Discov Design.* **17**: 1–25.

14. Garg, R., Kurup, A., Mekapati, S. B. & Hansch, C. (2003). Searching for allosteric effects via QSAR. Part II. *Bioorg Med Chem.* **11**: 621–628.
15. Kier, L. B. & Hall, L. H. (1993). The generation of molecular-structures from a graph-based Qsar equation. *Quant Struct Act Relat.* **12**: 383–388.
16. Hall, L. H. & Kier, L. B. (2001). Issues in representation of molecular structure – The development of molecular connectivity. *J Mol Graph Model.* **20**: 4–18.
17. Anker, L. S., Jurs, P. C. & Edwards, P. A. (1990). Quantitative structure retention relationship studies of odor-active aliphatic-compounds with oxygen-containing functional-groups. *Anal Chem.* **62**: 2676–2684.
18. Crippen, G. M. (1982). Distance geometry analysis of the benzodiazepine binding-site. *Mol Pharmacol.* **22**: 11–19.
19. Hopfinger, A. J. (1980). A Qsar Investigation of dihydrofolate-reductase inhibition by baker triazines based upon molecular shape-analysis. *J Am Chem Soc.* **102**: 7196–7206.
20. Boulu, L. G. & Crippen, G. M. (1989). Voronoi binding-site models – calculation of binding modes and influence of drug-binding data accuracy. *J Comb Chem.* **10**: 673–682.
21. Cramer, R. D., III, Patterson, D. E. & Bunce, J. D. (1988). Comparative molecular field analysis (CoMFA). I. Effect of shape on binding of steroids to carrier proteins. *J Am Chem Soc.* **110**: 5959–5967.
22. So, S. S. & Richards, W. G. (1992). Application of neural networks – quantitative structure-activity-relationships of the derivatives of 2,4-diamino-5-(substituted-benzyl)pyrimidines as Dhfr inhibitors. *J Med Chem.* **35**: 3201–3207.
23. Tetko, I. V., Tanchuk, V. Y., Chentsova, N. P., Antonenko, S. V., Poda, G. I., Kukhar, V. P. & Luik, A. I. (1994). Hiv-1 reverse-transcriptase inhibitor design using artificial neural networks. *J Med Chem.* **37**: 2520–2526.
24. Ajay, A. & Murcko, M. A. (1995). Computational methods to predict binding free energy in ligand-receptor complexes. *J Med Chem.* **38**: 4953–4967.
25. Andrea, T. A. & Kalayeh, H. (1991). Applications of neural networks in quantitative structure-activity-relationships of dihydrofolate-reductase inhibitors. *J Med Chem.* **34**: 2824–2836.
26. Bolis, G., Dipace, L. & Fabrocini, F. (1991). A machine learning approach to computer-aided molecular design. *J Comput Aided Mol Des.* **5**: 617–628.
27. King, R. D., Muggleton, S., Lewis, R. A. & Sternberg, M. J. E. (1992). Drug design by machine learning – the use of inductive logic programming to model the structure-activity-relationships of trimethoprim analogs binding to dihydrofolate-reductase. *Proc Natl Acad Sci U S A.* **89**: 11322–11326.
28. Jain, A. N., Dietterich, T. G., Lathrop, R. H., Chapman, D., Critchlow, R. E., Bauer, B. E., Webster, T. A. & Lozanoperez, T. (1994). Compass – a shape-based machine learning tool for drug design. *J Comput Aided Mol Des.* **8**: 635–652.
29. Zheng, W. F. & Tropsha, A. (2000). Novel variable selection quantitative structure-property relationship approach based on the k-nearest-neighbor principle. *J Chem Inf Comput Sci.* **40**: 185–194.
30. Xue, C. X., Zhang, R. S., Liu, H. X., Yao, X. J., Liu, M. C., Hu, Z. D. & Fan, B. T. (2004). An accurate QSPR study of O-H bond dissociation energy in substituted phenols based on support vector machines. *J Chem Inf Comput Sci.* **44**: 669–677.
31. Yao, X. J., Panaye, A., Doucet, J. P., Zhang, R. S., Chen, H. F., Liu, M. C., Hu, Z. D. & Fan, B. T. (2004). Comparative study of QSAR/QSPR correlations using support vector machines, radial basis function neural networks, and multiple linear regression. *J Chem Inf Comput Sci.* **44**: 1257–1266.
32. Kovatcheva, A., Golbraikh, A., Oloff, S., Xiao, Y. D., Zheng, W. F., Wolschann, P., Buchbauer, G. & Tropsha, A. (2004). Combinatorial QSAR of ambergris fragrance compounds. *J Chem Inf Comput Sci.* **44**: 582–595.
33. Mjolsness, E. & DeCoste, D. (2001). Machine learning for science: State of the art and future prospects. *Science* **293**: 2051–2055.
34. Herbrich, R. & Williamson, R. C. (2003). Algorithmic luckiness. *J Mac Learn Res.* **3**: 175–212.
35. Schneider, G. & Downs, G. (2003). Machine learning methods in QSAR modelling. *QSAR Comb Sci.* **22**: 485–486.
36. Sebastiani, P., Kohane, I. S. & Ramoni, M. F. (2003). Machine learning in the Genomics era – Editorial: Methods in functional genomics. *Machine Learning* **52**: 5–9.
37. Smith, M. G. & Bull, L. (2003). Feature construction and selection using Genetic Programming and a Genetic Algorithm. *Genetic Programming, Proceedings* **2610**, 229–237.
38. Armengol, E. & Plaza, E. (2003). Discovery of toxicological patterns with lazy learning. *Knowledge-Based Intelligent Information and Engineering Systems, Pt 2, Proceedings* **2774**, 919–926.

39. Oloff, S., Zhang, S., Sukumar, N., Breneman, C. & Tropsha, A. (2006). Chemometric analysis of ligand receptor complementarity: identifying complementary ligands based on receptor information (CoLiBRI). *J Chem Inf Model*. **46**: 844–851.
40. Zhang, S., Golbraikh, A. & Tropsha, A. (2006). Development of quantitative structure-binding affinity relationship models based on novel geometrical chemical descriptors of the protein-ligand interfaces. *J Med Chem*. **49**: 2713–2724.
41. Zhang, S., Golbraikh, A., Oloff, S., Kohn, H. & Tropsha, A. (2006). A novel automated lazy learning QSAR (ALL-QSAR) approach: method development, applications, and virtual screening of chemical databases using validated ALL-QSAR models. *J Chem Inf Model* **46**: 1984–1995.
42. Zhang, S., Wei, L., Bastow, K., Zheng, W., Brossi, A., Lee, K. H. & Tropsha, A. (2007). Antitumor agents 252. Application of validated QSAR models to database mining: discovery of novel tylophorine derivatives as potential anticancer agents. *J Comput Aided Mol Des*. **21**: 97–112.
43. Duch, W., Swaminathan, K. & Meller, J. (2007). Artificial intelligence approaches for rational drug design and discovery. *Curr Pharm Des*. **13**: 1497–1508.
44. Clark, D. E. & Grootenhuys, P. D. (2002). Progress in computational methods for the prediction of ADMET properties. *Curr Opin Drug Discov Dev*. **5**: 382–390.
45. Davis, A. M. & Riley, R. J. (2004). Predictive ADMET studies, the challenges and the opportunities. *Curr Opin Chem Biol*. **8**: 378–386.
46. Li, H., Yap, C. W., Ung, C. Y., Xue, Y., Li, Z. R., Han, L. Y., Lin, H. H. & Chen, Y. Z. (2007). Machine learning approaches for predicting compounds that interact with therapeutic and ADMET related proteins. *J Pharm Sci*. **96**: 2838–2860.
47. Yap, C. W., Cai, C. Z., Xue, Y. & Chen, Y. Z. (2004). Prediction of torsade-causing potential of drugs by support vector machine approach. *Toxicol Sci*. **79**: 170–177.
48. Kubinyi, H. (2003). Drug research: myths, hype and reality. *Nat Rev Drug Discov*. **2**: 665–668.
49. Reddy, M. R. & Erion, M. D. (1998). Structure-based drug design approaches for predicting binding affinities of HIV1 protease inhibitors. *J Enzyme Inhib*. **14**: 1–14.
50. Taylor, R. D., Jewsbury, P. J. & Essex, J. W. (2002). A review of protein-small molecule docking methods. *J Comput Aided Mol Des*. **16**: 151–166.
51. Kuntz, I. D., Meng, E. C. & Shoichet, B. K. (1994). Structure-based molecular design. *Acc Chem Res*. **27**: 117–123.
52. Hardy, L. W. & Malikayil, A. (2003). The impact of structure-guided drug design on clinical agents. *Curr Drug Discov*. **3**: 15–20.
53. Maryanoff, B. E. (2004). Inhibitors of serine proteases as potential therapeutic agents: The road from thrombin to trypsin to cathepsin G. *J Med Chem*. **47**: 769–787.
54. Chen, L. S., Nowak, B. J., Ayres, M. L., Krett, N. L., Rosen, S. T., Zhang, S. & Gandhi, V. (2009). Inhibition of ATP synthase by chlorinated adenosine analogue. *Biochem Pharmacol*. **78**: 583–591.
55. Du-Cuny, L., Song, Z., Moses, S., Powis, G., Mash, E. A., Meuillet, E. J. & Zhang, S. (2009). Computational modeling of novel inhibitors targeting the Akt pleckstrin homology domain. *Bioorg Med Chem*. **17**: 6983–6992.
56. Mahadevan, D., Powis, G., Mash, E. A., et al. (2008). Discovery of a novel class of AKT pleckstrin homology domain inhibitors. *Mol Cancer Ther*. **7**: 2621–2632.
57. Moses, S. A., Ali, M. A., Zuohe, S., Du-Cuny, L., Zhou, L. L., Lemos, R., Ihle, N., Skillman, A. G., Zhang, S., Mash, E. A., Powis, G., Meuillet, E. J. (2009). In vitro and in vivo activity of novel small-molecule inhibitors targeting the pleckstrin homology domain of protein kinase B/AKT. *Cancer Res*. **69**: 5073–5081.
58. Zhang, S., Ying, W. S., Siahaan, T. J. & Jois, S. D. S. (2003). Solution structure of a peptide derived from the beta subunit of LFA-1. *Peptides*. **24**: 827–835.
59. Zhang, S., Kumar, K., Jiang, X., Wallqvist, A. & Reifman, J. (2008). DOVIS: an implementation for high-throughput virtual screening using AutoDock. *BMC Bioinformatics* **9**: 126.
60. Zhang, S., Kaplan, A. H. & Tropsha, A. (2008). HIV-1 protease function and structure studies with the simplicial neighborhood analysis of protein packing method. *Proteins*. **73**: 742–753.
61. Zhang, S. & Du-Cuny, L. (2009). Development and evaluation of a new statistical model for structure-based high-throughput virtual screening. *Int J Bioinform Res Appl*. **5**: 269–279.
62. Kitchen, D. B., Decornez, H., Furr, J. R. & Bajorath, J. (2004). Docking and scoring in virtual screening for drug discovery: Methods and applications. *Nat Rev Drug Discov*. **3**: 935–949.
63. Morris, G. M., Goodsell, D. S., Halliday, R. S., Huey, R., Hart, W. E., Belew, R. K. & Olson, A. J. (1999). Automated docking using a

- Lamarckian genetic algorithm and an empirical binding free energy function. *J Comput Chem.* **19**: 1639–1662.
64. Halperin, I., Ma, B., Wolfson, H. & Nussinov, R. (2002). Principles of docking: An overview of search algorithms and a guide to scoring functions. *Proteins* **47**: 409–443.
 65. Makino, S. & Kuntz, I. D. (1997). Automated flexible ligand docking method and its application for database search. *J Comb Chem.* **18**: 1812–1825.
 66. Shoichet, B. K. & Kuntz, I. D. (1991). Protein docking and complementarity. *J Mol Biol.* **221**: 327–346.
 67. Kramer, B., Metz, G., Rarey, M. & Lengauer, T. (1999). Ligand docking and screening with FlexX. *Med Chem Res.* **9**: 463–478.
 68. Rarey, M., Kramer, B., Lengauer, T. & Klebe, G. (1996). A fast flexible docking method using an incremental construction algorithm. *J Mol Biol.* **261**: 470–489.
 69. Goodsell, D. S., Morris, G. M. & Olson, A. J. (1996). Automated docking of flexible ligands: applications of AutoDock. *J Mol Recognit.* **9**: 1–5.
 70. Cornell, W. D., Cieplak, P., Bayly, C. I., et al. (1996). A second generation force field for the simulation of proteins, nucleic acids and organic molecules. *J Am Chem Soc.* **117**: 5179–5197.
 71. MacKerell, A. D., Jr., Banavali, N. & Foloppe, N. (2000). Development and current status of the CHARMM force field for nucleic acids. *Biopolymers.* **56**: 257–265.
 72. Halgren, T. A. (1996). Merck molecular force field: 1. Basis, form, scope, parameterization, and performance of MMFF94. *J Comput Chem.* **17**: 490–519.
 73. Shoichet, B. K., Leach, A. R. & Kuntz, I. D. (1999). Ligand solvation in molecular docking. *Proteins.* **34**: 4–16.
 74. Bohm, H. J. (1992). Ludi – rule-based automatic design of new substituents for enzyme-inhibitor leads. *J Comput Aided Mol Des.* **6**: 593–606.
 75. Bohm, H. J. (1992). The computer-program Ludi – a new method for the de novo design of enzyme-inhibitors. *J Comput Aided Mol Des.* **6**: 61–78.
 76. Bohm, H. J. (1998). Prediction of binding constants of protein ligands: a fast method for the prioritization of hits obtained from de novo design or 3D database search programs. *J Comput Aided Mol Des.* **12**: 309–323.
 77. Head, R. D., Smythe, M. L., Oprea, T. I., Waller, C. L., Green, S. M. & Marshall, G. R. (1996). VALIDATE: a new method for the receptor-based prediction of binding affinities of novel ligands. *J Am Chem Soc.* **118**: 3959–3969.
 78. Gohlke, H., Hendlich, M. & Klebe, G. (2000). Knowledge-based scoring function to predict protein-ligand interactions. *J Mol Biol.* **295**: 337–356.
 79. DeWitte, R. S. & Shakhnovich, E. I. (1996). SMOG: de novo design method based on simple, fast, and accurate free energy estimates. 1. Methodology and supporting evidence. *J Am Chem Soc.* **118**: 11733–11744.
 80. Muegge, I. & Martin, Y. C. (1999). A general and fast scoring function for protein-ligand interactions: a simplified potential approach. *J Med Chem.* **42**: 791–804.
 81. Mitchell, J. B. O., Laskowski, R. A., Alex, A. & Thornton, J. M. (1999). BLEEP-potential of mean force describing protein-ligand interactions: I. Generating potential. *J Comput Chem.* **20**: 1165–1176.
 82. Johnston, W. E., Jacobson, V. L., Loken, S. C., Robertson, D. W. & Tierney, B. L. (1992). High-performance computing, high-speed networks, and configurable computing environments: progress toward fully distributed computing. *Crit Rev Biomed Eng.* **20**: 315–354.
 83. Golbraikh, A., Shen, M., Xiao, Z. Y., Xiao, Y. D., Lee, K. H. & Tropsha, A. (2003). Rational selection of training and test sets for the development of validated QSAR models. *J Comput Aided Mol Des.* **17**: 241–253.



<http://www.springer.com/978-1-61779-011-9>

Drug Design and Discovery

Methods and Protocols

Satyanarayanajois, S.D. (Ed.)

2011, XII, 69 p., Hardcover

ISBN: 978-1-61779-011-9

A product of Humana Press