Computer-Aided Drug Discovery and Protein-Ligand Docking

LI, Hongjian

A Thesis Submitted in Partial Fulfilment
of the Requirements for the Degree of
Doctor of Philosophy
in
Computer Science and Engineering

The Chinese University of Hong Kong
February 2015

Thesis Assessment Committee

Professor HENG Pheng Ann (Chair)
Professor LEUNG Kwong Sak (Thesis Supervisor)
Professor WONG Man Hon (Thesis Co-supervisor)
Professor WONG Kin Hong (Committee Member)
Professor LEONG Hong Va (External Examiner)

Abstract of thesis entitled:

Computer-Aided Drug Discovery and Protein-Ligand Docking

Submitted by LI, Hongjian
for the degree of Doctor of Philosophy
at The Chinese University of Hong Kong in February 2015

Developing a new drug costs up to US$2.6B and 13.5 years. To save money and time, we have developed a toolset for computer-aided drug discovery, and utilized our toolset to discover drugs for the treatment of cancers and influenza.

We first implemented a fast protein-ligand docking tool called idock, and obtained a substantial speedup over a popular counterpart. To facilitate the large-scale use of idock, we designed a heterogeneous web platform called istar, and collected a huge database of more than 23 million small molecules. To elucidate molecular interactions in web, we developed an interactive visualizer called iview. To synthesize novel compounds, we developed a fragment-based drug design tool called iSyn. To improve the predictive accuracy of binding affinity, we exploited the machine learning technique random forest to re-score both crystal and docked poses. To identify structurally similar compounds, we ported the ultrafast shape recognition algorithms to istar. All these tools are free and open source.

We applied our novel toolset to real world drug discovery. We repurposed anti-acne drug adapalene for the treatment of human colon cancer, and identified potential inhibitors of influenza viral proteins. Such new findings could hopefully save human lives.

摘要

開發一種新藥需要多至 26 億美元和 13 年半的時間。為節省金錢和時間，我們開發了一套計算機輔助藥物研發工具集，並運用該工具集尋找藥物治療癌症和流感。

我們首先實現了一個快速的蛋白與配體對接工具 idock，相比一個同類流行軟件獲得了顯著的速度提升。為輔助 idock 的大規模使用，我們設計了一個異構網站平台 istar，收集了多達兩千三百萬個小分子的大型數據庫。為在網頁展示分子間相互作用，我們開發了一個交互式可視化軟件 iview。為生成全新的化合物，我們開發了一個基於分子片段的藥物設計工具 iSyn。為改進結合強度預測的精度，我們利用了機器學習技術隨機森林去重新打分晶體及預測構象。為尋找結構相似的化合物，我們移植了超快形狀識別算法至 istar。所有這些工俱全是免費和開源。

我們應用了此創新工具集至現實世界藥物尋找中。我們發現抗痙瘡藥阿達帕林可用於治療人類結腸癌，亦篩選出流感病毒蛋白的潛在抑制物。這些新發現可望拯救人類生命。

Acknowledgement

I believe I would never have completed this thesis and gotten this far without five years of generous support of my supervisors, Prof. Kwong-Sak Leung and Prof. Man-Hon Wong. They treated me with considerable respect and understanding, and made me feel like a worthy researcher. I am sure that a simple thank you will never be enough to convey my gratitude for their praises and criticisms along the way.

I feel extremely lucky and humbled to have had the opportunity to learn from an impressive and supportive research group led by Prof. Kwong-Sak Leung, Prof. Man-Hon Wong, Prof. Kin-Hong Lee and Prof. Kevin Yuk-Lap Yip. I would extend my sincere thanks to my fellow postgraduate students, Cyrus Tak-Ming Chan, Shaoke Lou, David Chi-Fai Lam, Peter Leung-Yau Lo, Xihao Hu and Ho Yin Szeto. They have inspired me in their own way to pursue my academic desires, and taught me to enjoy my education by modeling excellence in their teaching, scholarship and innovation. I would also like to express my heartfelt gratitude to my undergraduate students, Chun Ho Chan and Hei Lun Cheung, who implemented useful software.

It has been a great pleasure to collaborate with excellent scholars in my research field. Dr. Pedro J. Ballester provided me encouraging and illuminating instructions in doing novel research and writing academic manuscripts. Takanori Nakane helped me get acquainted with the visualization domain and was always willing to share with me his source code. Prof. Marie

Chia-Mi Lin and Xilan Shi conducted biological experiments and augmented my pure computational research.

I am particularly indebted to my father Qiteng Li, who have made incredible and selfless sacrifices for me over years that I might someday have this privilege. He always listened to me and helped me work out many little troubles in my daily life.

I am thankful to the Direct Grant and GRF Grant (Project No. 2150764) from Chinese University of Hong Kong for financial support.

A decade ago, my mother died of colon cancer.
A decade after, her son manages to find a cure.

十年生死兩茫茫
不思量　自難忘

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

Drug discovery is an expensive and long-term business. Summarized from 13 research articles published from 1980 to 2009, original estimates of the cost of drug development ranged more than 9-fold, from USD$92 million cash (USD$161 million capitalized) to USD$883.6 million cash (USD$1.8 billion capitalized) [1]. The cost of drug development, including the price of failure and the opportunity cost, has more than doubled in the past decade and has now reached US$2.6 billion in 2013 dollars [2]. Discovering and developing a new molecular entity (NME) required 11.4 to 13.5 years using the R&D performance productivity data from 13 large pharmaceutical companies across 2000 to 2007 [3]. An recent report [4] reviewed the rates of NMEs introduction starting from 1827 through to the end of 2013, and found that two-thirds of NMEs are controlled by a handful of companies, and a growing number of NMEs are controlled by marketing organizations that have little or no internal drug dis-

covery or development activities.

The process of modern drug discovery typically includes target identification, hit identification, lead optimization and clinical trials. A biological target is any system that can potentially be modulated by a molecule to produce a beneficial effect. A target could be a fundamental pathological pathway, altering which is expected to be curative or anti-symptomatic. Hits are compounds that have activity at a predetermined level against a target, but little else is known at this early stage. Leads are optimized hits that display strong potency and selectivity, physicochemical characteristics, and absorption, distribution, metabolism, excretion and toxicity (ADMET) properties. Successful candidate leads will then be submitted to the health authorities to get permission to conduct clinical investigations on animals and humans.

An essential ingredient of drug discovery is to discover inhibitory molecules for pharmaceutical protein targets of therapeutic interest. Take the HIV (Human Immunodeficiency Virus) virus for example [5]. The virus comprises several protein enzymes, which play critical roles in viral replication. In HIV-infected cells, the viral reverse transcriptase reversely transcribes viral RNA into viral DNA, the viral integrase integrates viral DNA into human genomic DNA, and the viral protease assemblies viral RNA and viral proteins into a new virion. This replication cycle will be blocked if the viral proteins are inhibited. Such inhibitors are typically small compounds called ligands, which function through binding to the enzymatic or allosteric

sites of target proteins.

Terminologically, screening refers to the process of discovering ligands that show activity towards certain proteins of interest. A library of compounds is routinely screened to shortlist candidate ligands. When this process is done *in silico* using computer simulations, it is called virtual screening. Regarding the methods in use, virtual screening can be classified into structure-based virtual screening and ligand-based virtual screening. Their major difference lies in whether the target protein is present or absent. Structure-based virtual screening uses explicit knowledge of the target protein to suggest candidate protein-ligand complexes commonly via a method called docking, whereas ligand-based virtual screening does not encode target information but infers required characteristics of binders from known bioactive ligands.

To really aid drug discovery, a complete toolset should include tools for both structure-based virtual screening (chapter 2) and ligand-based virtual screening (chapter 9), as well as relevant tools and studies, e.g. a web platform (chapter 3), visualization (chapter 4), drug design and synthesis (chapter 5), binding affinity prediction (chapters 6 and 7), pose generation error reduction (chapter 8), and others. Eventually these tools and studies become useful only when they are applied to real world problems, such as finding cures for cancers (chapter 10) and influenza (chapter 11).

1.2 Motivation

Drug discovery is economy driven *per se*. Biochemical means are both cost- and time-inefficient. This highlights the need for cheaper and faster methods, and computer-aided drug discovery (CADD) thus comes into the scene. Complementing expensive laboratory experiments with cheap computer simulations is obviously the right way to go. Robust computational frameworks are indeed highly demanded by the industry in order to automate the early phases of modern drug discovery such as hit identification and lead optimization.

Although a large amount of CADD tools have been developed over recent decades, the majority of them, unfortunately, suffer from several notable problems. These tools 1) are commercial, selling at a price that most small enterprises and academic institutions cannot afford, 2) are proprietary and closed source, making third parties difficult to study the internal implementations or locate potential bugs, 3) conform to different standards and formats, resulting in weak data portability and information loss, 4) require intensive and tedious configurations and lack a friendly user interface, a great obstacle for new users to get started, 5) run rather slowly, incapable of utilizing the multi- and many-core architectures of modern computers, or even worse, 6) are declared dead immediately upon their initial release due to zero maintenance afterwards. In this thesis, we attempt to address these shortcomings.

1.3 Objective

We aim to develop a pragmatic and concise CADD toolset, and ultimately apply it to the discovery of novel drugs. Keeping several key goals in mind, we design our toolset to 1) be freely available to the general public, 2) be released under permissive open source licenses, 3) conform to official standards, 4) provide a responsive web version, 5) run reasonably fast, and 6) track bugs and issues and incorporate user feedback. We emphasize reproducibility, which has the potential to serve as a minimum standard for judging scientific claims when full independent replication of a study is not possible [6]. Most importantly, we shall utilize our toolset to discover potent drugs against certain diseases of therapeutic interest and hopefully save human lives.

1.4 Thesis contributions and outline

Figure 1.1 paints the overall contributions of this thesis, which is organized into 12 chapters.

Chapter 2 presents idock [7], our multithreaded flexible ligand docking tool for native support of structure-based virtual screening in a superfast fashion. Based on the state-of-the-art AutoDock Vina, our idock substantially revises the numerical approximation model and enhances the fundamental implementation of various components with modern C++11 tricks. Notably, it encapsulates a novel feature for dimension reduction

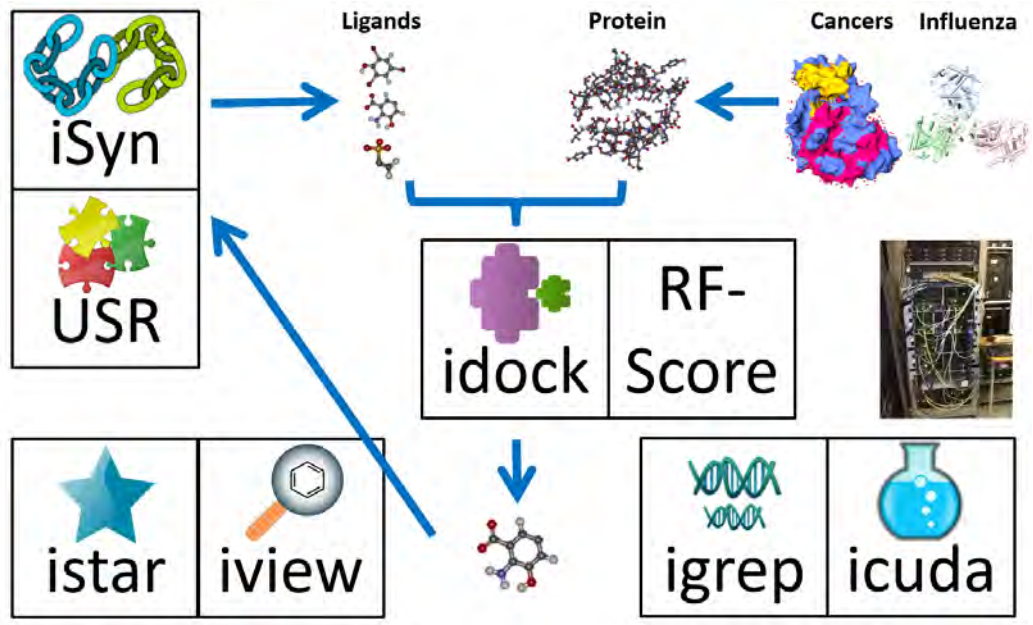


Figure 1.1: Contributions of this thesis.

by detecting inactive torsions. Compared with AutoDock Vina [8], idock obtained a speedup of 3.3 in terms of CPU time and a speedup of 7.5 in terms of elapsed time on average, making it a very competitive tool. We have then used idock in a prospective virtual screening campaign of docking 10,938 drug-like ligands against HIV reverse transcriptase with minimal toxical side effects against four other human proteins for the treatment of AIDS. This project has been published [7].

Chapter 3 presents istar [9], our versatile SaaS (Software as a Service) web platform to promote software usage by a wide variety of users from different disciplines. Particularly, we have hosted idock on istar for large-scale prospective structure-based virtual screening. Our istar website supports three novel fea-

tures: 1) filtering ligands by desired molecular properties and previewing the number of ligands to dock, 2) monitoring job progress in real time, and 3) visualizing docked conformations and outputting supplier information for easy purchasing. We have collected as many as 23,129,083 ligands, revamped our idock to version 2.0, and integrated RF-Score [10] as an alternative rescoring function. We have shown that idock achieved comparable success rates while outperforming AutoDock Vina in terms of docking speed by at least 8.69 times and at most 37.51 times. In combination with RF-Score, istar managed to reproduce Pearson and Spearman correlation coefficients of as high as 0.855 and 0.859, respectively, between the experimental and the predicted binding affinity. We believe istar constitutes a step toward generalizing the use of docking tools beyond the traditional molecular modeling community. idock@istar is freely available at http://istar.cse.cuhk.edu.hk/idock. This project has been published [9]. According to Google Analytics, throughout 2014, istar had served 460 sessions, 271 users, and 631 pageviews from 33 countries.

Chapter 4 presents iview [11], our easy-to-use interactive WebGL visualizer for protein-ligand complex to enable non-experts to quickly elucidate protein-ligand interactions in a 3D manner. As far as we are aware, iview is the only web visualizer that simultaneously utilizes GPU hardware acceleration and supports three pragmatic features: macromolecular surface construction, virtual reality effects, and PDBQT format parsing. Moreover, based on the feature-rich version of iview, we have also de-

veloped a concise version specifically for our idock web service on istar to aid online protein-ligand docking. This demonstrates the excellent portability of iview, which can be easily integrated into any bioinformatics application that requires interactive protein-ligand visualization. iview is freely available at http://istar.cse.cuhk.edu.hk/iview. This project has been published [11].

Chapter 5 presents iSyn [12, 13], our effective and efficient fragment-based drug design tool that generates desired *de novo* compounds with promising potency and molecular mass to complement structure-based virtual screening. It features an evolutionary algorithm that creates novel ligands with drug-like properties and ensures synthetic feasibility with click chemistry. Interfacing with our fast molecular docking engine idock and our interactive WebGL visualizer iview, iSyn substantially reduces the drug candidate evaluation time and increases productivity. Benchmarking results of iSyn in generating novel inhibitors *ex nihilo* of two important drug targets TbREL1 and CDK2 have proved its strength in significantly enhancing the predicted binding affinity of the best generated ligand by more than 3 orders of magnitude in potency within a reasonable time. iSyn is freely available at http://istar.cse.cuhk.edu.hk/iSyn.tgz. This project has been published [12, 13].

Chapter 6 presents our study on the use of random forest (RF) to improve binding affinity prediction, with Cyscore [14] as a baseline. We show that the simple functional form typically implemented in classical scoring functions is detrimental for the

predictive performance, and substituting machine learning techniques like RF for the commonly-used multiple linear regression (MLR) model can improve predictive performance. We point out that a significant drawback for MLR-based scoring functions is their incapability of exploiting abundant training samples, so they cannot benefit from the increasing availability of future experimental data. On the other hand, we perform cross validation to show that feeding more training samples to RF can increase its predictive performance, and using more structural features appropriately can also substantially boost its predictive accuracy. We conclude that one can strongly improve Cyscore by changing the regression model from MLR to RF and expanding the feature set as well as the sample set. This project has been published [15].

Chapter 7 presents our another study on the use of RF to boost the predictive performance of classical scoring functions, this time with AutoDock Vina [8] as a baseline. With the help of a proposed novel benchmark, we demonstrate that the improvement of using RF over MLR will be larger as more data becomes available for training, as regression models implying additive functional forms do not improve with more training data. We discuss how the latter opens the door to new opportunities in scoring function development. We also discuss the applicability domain of MLR- and RF-based scoring functions, and demonstrate that the tendency of RF-based scoring functions to overfit training data is not a limitation but simply a trait. We also suggest that incorporating ligand- and protein-only properties into

the scoring function is a promising path to future improvements. Finally we provide software to directly re-score Vina-generated poses in order to facilitate the translation of this advance to enhance structure-based molecular design. This project has been published [16, 17].

Chapter 8 presents our further study on the use of RF to re-score docked poses instead of crystal poses, because the latter are usually unavailable in the common scenario of large-scale prospective virtual screening, such as our istar web service [9]. We investigate the impact of pose generation error on the predictive performance of both classical and machine-learning scoring functions, and find that re-training the scoring functions on docked poses can be a simple and quick solution to reduce the negative impact of pose generation error. Moreover, we study the scoring functions' capability of predicting the near-native pose that is most conformationally closest to the crystal pose, and observe that machine-learning scoring functions, while excelling at binding affinity prediction, performed much worse than Vina at native pose prediction. We explain that this could be due to the confounding factor that the docked poses were all generated and optimized by Vina. This project has been published [18].

Chapter 9 presents a pragmatic implementation of USR (Ultrafast Shape Recognition) [19] and its extension USRCAT (USR with Credo Atom Types) [20] based on our istar web platform in order to quickly and conveniently search for compounds structurally similar to a query ligand in terms of shape. Our molecu-

lar database is populated with more than 23 million diverse compounds so as to reduce the possibility of missing compounds with similar shape to the query. We perform screening time analysis, based on which we exploit three levels of parallelism with a novel implementation of sum of absolute differences using AVX (Advanced Vector Extensions) to accelerate job execution. Our USR@istar supports a query ligand in SDF, MOL2, XYZ, PDB or PDBQT format, and interfaces with our iview WebGL visualizer for interactive visualization of high-score hits. USR@istar is freely available at http://istar.cse.cuhk.edu.hk/usr. Our results for 19 query ligands of different molecular sizes have shown that USR and USRCAT lead to very different output compounds in their top 5 matches. We are meanwhile surprised to discover that different file formats of the same input ligand affect the classification of pharmacophoric subsets. Our implementation USR@istar requires just 30 seconds to complete a query when the precalculated features are loaded in advance. In addition, we have also briefly described USRT (USR with Torsions), the very first USR-like algorithm that can identify different conformations of the same ligand. One of its biggest applications is to circumvent the task of conformer generation.

It is worthwhile to highlight that all of the above CADD tools are free and open source under permissive licenses. We emphasize reproducibility [6].

The chapters above are well connected in that idock (chapter 2) serves as a fundamental docking engine, istar (chapter 3) provides a web interface, iview (chapter 4) permits online visu-

alization, iSyn (chapter 5) generates new compounds, machine-learning models (chapters 6, 7 and 8) improve binding affinity prediction, and USR@istar (chapter 9) searches for similar compounds. The chapters below describe prospective applications of our pragmatic toolset.

Chapter 10 describes our case study of CDK2-related cancers. CDK2 (Cyclin-dependent kinase 2) is a key factor regulating the cell cycle G1 to S transition and a hallmark for cancers. We used idock [7, 9] prospectively for the first time in identifying potential CDK2 inhibitors from 4,311 approved small molecule drugs using a repurposing strategy so as to minimize drug toxicity. Totally 44 CDK2 structures were collected and ensemble docking was carried out. Among the top compounds sorted by idock score, nine were purchased and tested *in vitro*. Among them, the anti-acne drug adapalene exhibited the highest anti-proliferative effect in human colon cancer. We demonstrated for the first time that adapalene treatment significantly increased the percentage of cells in G1 phase, and decreased the expressions of CDK2, cyclin E and Rb, as well as the phosphorylations of CDK2 on Thr160 and Rb on Ser795. We showed for the first time that oral adapalene treatment significantly and dose-dependently inhibited tumor growth *in vivo* in nude mice subcutaneously xenografted with human colorectal cancer cells. Adapalene (20 mg/kg) showed strong anti-tumor activity, comparable to that of the leading cancer drug oxaliplatin (40 mg/kg). The combination with adapalene and oxaliplatin exhibited the highest therapeutic effect. These results

indicated for the first time that adapalene is a potential inhibitor of CDK2 and a candidate anti-cancer drug for the treatment of human colorectal cancer.

Chapter 11 describes our case study of influenza A. We targeted at three novel protein targets: the tail-loop binding domain of nucleoprotein, the PB1-binding domain of PA and the cap-binding domain of PB2 in the RNA-dependent RNA polymerase (RdRP). We utilized idock [7, 9] to perform structure-based virtual screening of 273,880 cheaply available compounds, and identified hits that were predicted to establish strong interactions with their respective viral protein target and hence believed to exhibit strong inhibitory effects. These identified compounds may serve as promising candidates for subsequent investigations *in vitro* and *in vivo*.

The appendix lists my journal and conference publications during my PhD study career in chronological order.

□ **End of chapter.**

Chapter 2

idock: protein-ligand docking

The increasing availability of macromolecular structural data catalyzes the development of protein-ligand docking methods. AutoDock Vina is a competitive protein-ligand docking tool well known for its fast execution and high accuracy. Nevertheless, when docking a massive number of ligands, Vina has to be run multiple times, repeating protein parsing and grid maps building over and over again. There are tremendous requests for revising Vina to reuse precalculated data and incorporate built-in support for virtual screening.

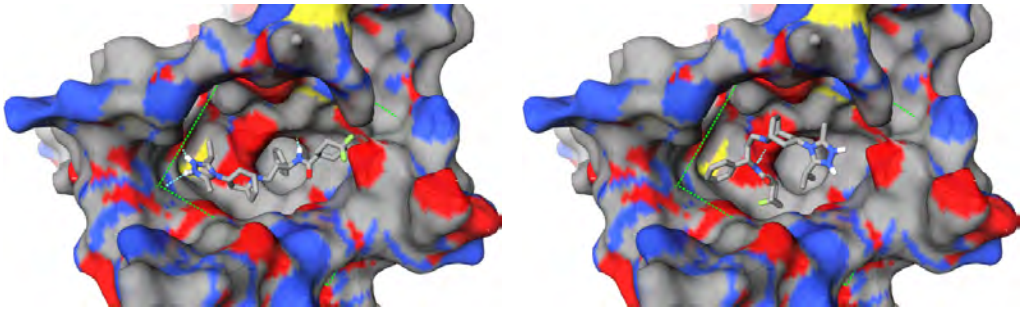
We developed idock, which inherits from AutoDock Vina the accurate scoring function and the efficient optimization algorithm, but substantially improves the fundamental implementation and numerical model for even faster execution. idock achieved a speedup of 3.3 in terms of CPU time and a speedup of 7.5 in terms of elapsed time on average when benchmarked on docking 10,928 drug-like ligands. To demonstrate the pragmatic utility of idock, we presented our effort of finding potentially

promising compounds with strong potency and minimal side effects for the treatment of AIDS. idock is free and open source, available at https://GitHub.com/HongjianLi/idock.

This project was published in *Proceedings of the 2012 IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB)* on 9 May 2012 [7].

2.1 Background

Over recent decades, there has been a continual increase in the number of new therapeutic targets available for drug design. Advances in crystallography and nuclear magnetic resonance spectroscopy have revealed substantial structural details of proteins and protein–ligand complexes. The structures of biological macromolecules at atomic level are being routinely resolved and deposited into the world's largest and freely accessible repository called Protein Data Bank (PDB) [21, 22]. Meanwhile, the biological activity data of small molecules are also being regularly collected into public databases such as ChEMBL [23] and PubChem [24]. As of 15 Sep 2014, there are 103,199 structures in PDB, 1,638,394 compounds records in ChEMBL and 1,112,090 bioassays in PubChem. The rapid evolution of structure resolving techniques and the availability of structural and bioactivity resources highly catalyze the development of protein-ligand docking methods for structure-based virtual screening. Very often, the target protein is a viral enzyme of interest, and the small organic ligands that are predicted to inhibit the viral



(a) Crystal conformation. (b) Docked conformation.

Figure 2.1: An intuitive example of two conformations of a ligand.

enzyme are what we want to discover.

Protein-ligand docking is a method which predicts the preferred conformation and binding affinity of a small ligand when bound to a macro protein to form a stable complex. The ligand conformation refers to its spatial position, orientation, and torsions, if any. Figure 2.1 shows the crystal conformation and a docked conformation of the marketed HIV drug maraviroc in complex with the human CCR5 chemokine receptor (PDB: 4MBS). The protein is rendered in molecular surface representation colored by atom types, where nitrogens (likely positively-charged regions) and oxygens (likely negative-charged regions) are in blue and red, respectively. The ligand is rendered in stick representation with the same color scheme, where nitrogen and oxygen atoms are in blue and red, respectively. The binding cavity on the protein surface is depicted by a green cubic box. The putative intermolecular hydrogen bonds are shown as cyan dashed lines. The docked conformation was predicted by idock [7] and the figure was created by iview [11].

The binding affinity is a numerical value that suggests how strongly the interactions are formed between the protein and the ligand upon binding. Empirically, it is the overall effect of various chemical interactions involved, such as van der Waals force, electrostatic force, salt bridges, hydrogen bonding, hydrophobic effects, pi interactions, halogen interactions, metal interactions, and the like. It is usually estimated by the scoring function employed in a docking tool. The binding affinity is usually expressed in pKd unit, which is the negative logarithm of dissociation constant Kd. In Figure 2.1 the binding affinities of the crystal and docked conformations were predicted to be 8.27 and 8.01 pKd, respectively, by RF-Score [10]. The binding affinity can be alternatively expressed in terms of free energy ΔG in kcal/mol unit, which is usually a negative value. The lower the free energy, the higher the binding affinity.

Structure-based virtual screening can be regarded as a massive version of docking. Instead of a single ligand, a database of ligands are docked against the target protein, then ranked according to their predicted binding affinity, and finally the top ones are selected for further investigations.

When a docking program is treated as a black box, its input includes the 3D structures of a protein and a ligand, and its output includes several predicted conformations and their predicted binding affinity. Regarding the protein input, the PDB database [21, 22] almost serves as the unique *de facto* data source. Regarding the ligand input, there are dozens of data sources. The GDB-17 database [25] enumerates 166 billion organic small molecules.

The PubChem database [26] comprises 53 million unique structures and 160 million substance records. The ZINC database [27, 28] contains over 35 million purchasable compounds in ready-to-dock, 3D formats. The TCM@Taiwan database [29] comprises 37,170 TCM (Traditional Chinese Medicine) compounds from 352 TCM ingredients. Regarding the output visualization and analysis, PyMOL (http://www.pymol.org), Chimera [30], VMD [31], AutoDockTools4 [32], ViewDock TDW [33], PoseView [34] and LigPlot+ [35] are popular tools to visualize docked conformations and plot putative interaction charts. AuPosSOM [36] can be used to cluster docked conformations. BEDROC [37] and SLR [38] can be used as statistical metrics for docking method evaluation.

When a docking program is treated as a white box, it consists of two typical components, an algorithm to explore the conformational space, and a scoring function to predict the binding affinity given a sampled conformation. There is a huge body of docking tools, e.g. DOCK [39, 40], AutoDock 4 [32], AutoDock Vina [8], QuickVina [41], PLANTS [42–44], FITTED [45, 46], CRDOCK [47], LiGenDock [48], and PharmDock [49], as well as scoring functions, e.g. RF-Score [10, 50], SFCscore [51, 52], LISA [53], NNScore 2.0 [54], ID-Score [55], and Cyscore [14]. More can be found in literature surveys [56, 57].

Amongst a sea of docking programs, AutoDock Vina [8] (hereafter Vina for short) is a competitive one because not only it is free and open source under Apache License 2.0, but also it has been shown to improve the average accuracy of the bind-

ing mode predictions [8] and run faster than its counterpart AutoDock 4 [32] by an order of magnitude when benchmarked on virtual screening for HIV protease inhibitors [58]. Released in the second half of 2010, Vina has been cited over 1,700 times and adopted by a wide community of researchers. Apart from the Vina docking tool itself, there is a PyMOL plugin for AutoDock and Vina [59]. MOLA [60] is a bootable, self-configuring system for virtual screening using AutoDock4/Vina on computer clusters. VSDK [61] is a console application system of virtual screening of small molecules using AutoDock Vina on Windows. AUDocker LE [62] is a GUI for virtual screening with AutoDock Vina on Windows. VinaMPI [63] enables multiple receptor high-throughput virtual docking on high-performance computers. VinaLC [64] is another MPI implementation of Vina. All these auxiliary tools facilitate the use of Vina under various settings.

Vina has become increasingly attractive thanks to a series of success stories by third parties. To name a few, Vina was used for 1) docking studies on the HEPT derivatives of HIV-1 reverse transcriptase [65], 2) for side-chain residue flexibility study of VEGFR-2 (Vascular Endothelial Growth Factor Receptor 2), which is a known protein target for anti-angiogenic agents [66], 3) for identification of novel inhibitors of sirtuin 2, which is a NAD^+-dependent histone deacetylase enzyme [67], and 4) for repurposing study of FDA-approved drugs for cancer therapy in order to screen for compounds that potentially inhibit MDM2, which an E3 ubiquitin ligase that polyubiquiti-

nates p53 [68]. Such exciting success stories prove the real power of protein-ligand docking, Vina in particular, for computer-aided drug discovery.

2.2 Motivation

Although Vina is popular and competitive and well known for its fast execution and high accuracy, it is optimized for single-ligand docking rather than virtual screening. When it comes to docking a large pool of ligands, Vina has to be invoked multiple times, repeatedly parsing the same protein and creating the same internal data structures such as grid maps. There are enormous requests from the community for modifying and recompiling Vina to make it support virtual screening in a superfast manner by reusing protein data and grid maps. We were motivated by the desire to provide built-in support for virtual screening and therefore developed idock.

2.3 Objective

We interpreted the source code of Vina, and rewrote it in our own programming style so as to implement advanced features in idock freely without constraints. Our major goal was to significantly increase the docking speed without sacrificing the docking accuracy. To achieve this goal, we revised the underlying numerical approximation model, and implemented a novel feature to reduce the dimensionality of variables to be optimized. We

incorporated a large amount of C++11 implementation tricks to speedup idock. Meanwhile, we decided to extend the software availability to support more chemical elements and more operating systems. Lastly, we utilized idock to attempt to address a real life drug discovery problem.

2.4 Methods

2.4.1 Flowchart

Figure 2.2 shows the overall flowchart of idock. During initialization, idock precalculates the scoring function for all possible combinations of atom type pairs and interatomic distances. It parses the protein and determines the atom types with the help of residue sequence, and creates a thread pool to hold reusable threads. Then it enters a loop and fetches a ligand from a user-specified input folder to perform docking. It parses the ligand and determines the atom types with the aid of branch information, and meanwhile automatically detects and deactivates inactive torsions. It builds grid maps of 0.15625Å granularity by default on the fly with multithreading, and distributes multiple independent Monte Carlo tasks to the thread pool for concurrent execution. Then it merges the conformations from separate threads and clusters them with root mean square deviation RMSD 2.0Å, and writes them to the user-specified output folder and displays the predicted free energy on screen. It automatically proceeds with the next ligand until all are docked. Finally it destroys the thread pool and releases memory resources.

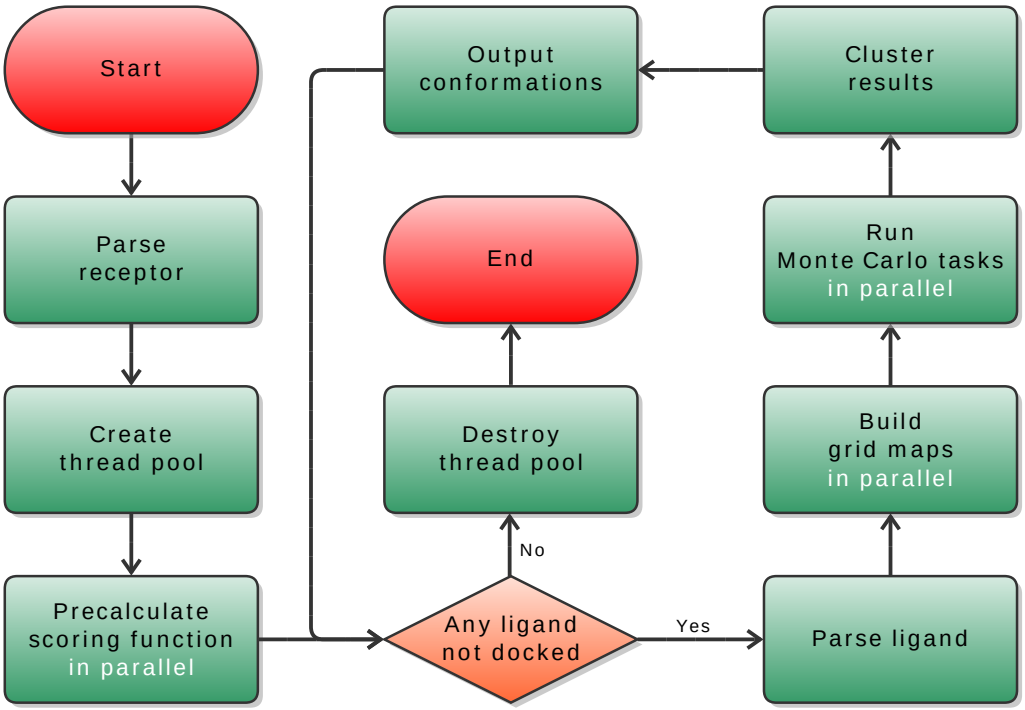


Figure 2.2: idock flowchart.

2.4.2 PDBQT specification

PDBQT is the protein and ligand input and output format used by AutoDock [32, 69], Vina [8], idock [7], and QuickVina [41]. Its official definition is at http://autodock.scripps.edu/faqs-help/faq/what-is-the-format-of-a-pdbqt-file. In PDBQT format (Figure 2.3), ligands can be treated as flexible with the idea of a torsion tree to represent the rigid and rotatable pieces. There is always one root, and zero or more branches. Branches can be nested. Every branch defines a rotatable bond.

The torsion tree is represented with some records specific to the PDBQT format. A ROOT record precedes the rigid part of the molecule, from which zero or more rotatable bonds may

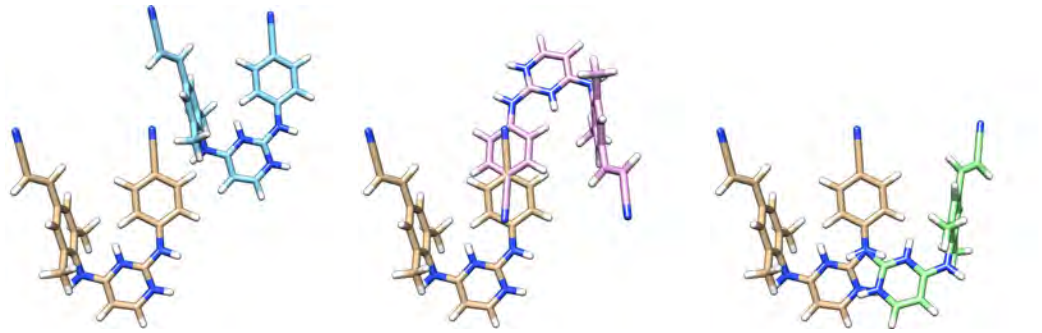
```
ROOT
ATOM      1  C13 T27 A 557      49.799 -31.025  35.312  1.00 25.09     0.044 A
ATOM      2  C14 T27 A 557      50.269 -31.013  33.977  1.00 23.67    -0.061 A
ATOM      3  H14 T27 A 557      50.154 -31.913  33.349  1.00  0.00     0.085 H
ATOM      4  C15 T27 A 557      50.883 -29.861  33.441  1.00 27.38    -0.055 A
ATOM      5  H15 T27 A 557      51.243 -29.877  32.399  1.00  0.00     0.087 H
ATOM      6  C16 T27 A 557      51.048 -28.673  34.221  1.00 27.55     0.049 A
ATOM      7  C17 T27 A 557      50.561 -28.702  35.570  1.00 21.87    -0.055 A
ATOM      8  H17 T27 A 557      50.671 -27.804  36.201  1.00  0.00     0.087 H
ATOM      9  C18 T27 A 557      49.940 -29.861  36.113  1.00 27.25    -0.061 A
ATOM     10  H18 T27 A 557      49.570 -29.855  37.152  1.00  0.00     0.085 H
ATOM     11  N5  T27 A 557      48.698 -33.181  36.272  1.00 26.72    -0.191 NA
ATOM     12  C19 T27 A 557      49.186 -32.213  35.845  1.00 29.50     0.099 C
ENDROOT
BRANCH   6  13
ATOM     13  N4  T27 A 557      51.661 -27.501  33.717  1.00 25.92    -0.192 N
ATOM     14  H4  T27 A 557      51.728 -27.498  32.699  1.00  0.00     0.184 HD
BRANCH  13  15
ATOM     15  C12 T27 A 557      52.195 -26.349  34.296  1.00 26.30     0.684 A
ATOM     16  N3  T27 A 557      51.982 -26.078  35.581  1.00 22.76    -0.176 N
ATOM     17  C11 T27 A 557      52.499 -24.952  36.144  1.00 25.85     0.122 A
ATOM     18  H3  T27 A 557      51.427 -26.719  36.148  1.00  0.00     0.186 HD
ATOM     19  C10 T27 A 557      53.261 -24.038  35.410  1.00 25.43    -0.025 A
ATOM     20  C9  T27 A 557      53.448 -24.380  34.061  1.00 23.08    -0.002 A
ATOM     21  H10 T27 A 557      53.682 -23.121  35.855  1.00  0.00     0.091 H
ATOM     22  N2  T27 A 557      52.922 -25.523  33.487  1.00 25.29    -0.202 N
ATOM     23  H9  T27 A 557      54.045 -23.703  33.427  1.00  0.00     0.112 H
ATOM     24  H2  T27 A 557      53.071 -25.740  32.501  1.00  0.00     0.183 HD
BRANCH  17  25
ATOM     25  N1  T27 A 557      52.219 -24.743  37.509  1.00 19.93    -0.341 N
ATOM     26  H1  T27 A 557      52.727 -24.015  38.011  1.00  0.00     0.167 HD
BRANCH  25  27
ATOM     27  C6  T27 A 557      51.256 -25.511  38.206  1.00 22.06     0.045 A
ATOM     28  C5  T27 A 557      51.633 -26.804  38.759  1.00 27.85    -0.034 A
ATOM     29  C4  T27 A 557      50.673 -27.581  39.439  1.00 26.06    -0.065 A
ATOM     30  C3  T27 A 557      49.357 -27.139  39.629  1.00 25.95    -0.035 A
ATOM     31  H4  T27 A 557      50.967 -28.568  39.834  1.00  0.00     0.085 H
ATOM     32  C2  T27 A 557      48.975 -25.867  39.102  1.00 24.94    -0.065 A
ATOM     33  C1  T27 A 557      49.920 -25.039  38.401  1.00 29.53    -0.034 A
ATOM     34  H2  T27 A 557      47.938 -25.514  39.235  1.00  0.00     0.085 H
BRANCH  28  35
ATOM     35  C8  T27 A 557      53.020 -27.393  38.614  1.00 23.50    -0.049 C
ATOM     36  H83 T27 A 557      53.412 -27.671  39.620  1.00  0.00     0.033 H
ATOM     37  H82 T27 A 557      52.949 -28.397  38.135  1.00  0.00     0.033 H
ATOM     38  H81 T27 A 557      53.779 -26.779  38.076  1.00  0.00     0.033 H
ENDBRANCH  28  35
ENDBRANCH  25  27
ENDBRANCH  17  25
ENDBRANCH  13  15
ENDBRANCH   6  13
TORSDOF 5
```

Figure 2.3: PDBQT content of a ligand.

emanate. The rigid root contains one or more ATOM or HETATM records in PDBQT style. These records resemble their traditional PDB counterparts, but diverge in columns 71-79 inclusive, where the first character in the line corresponds to column 1. The Gasteiger partial charge is stored in columns 71-76 inclusive in %6.3f format, i.e. right-justified, 6 characters wide, with 3 decimal places. The AutoDock atom type is stored in columns 78-79 inclusive in %-2.2s format, i.e. left-justified and 2 characters wide. An ENDROOT record follows the last atom in the rigid root. The ROOT/ENDROOT block of atoms is given first in the PDBQT file.

Sets of atoms moved by rotatable bonds are enclosed by BRANCH and ENDBRANCH records. These BRANCH/ENDBRANCH blocks follow the ROOT/ENDROOT block. Both BRANCH and ENDBRANCH records give two integers specifying the serial numbers of the first and second atoms involved in the rotatable bond. BRANCH/ENDBRANCH blocks can be nested. The last atom in a branch is followed by an ENDBRANCH record, whose serial numbers of the two atoms in the rotatable bond match those in the corresponding BRANCH record.

The last line contains a TORSDOF record, which is followed by an integer specifying the number of torsional degrees of freedom in the ligand.



(a) Positional degree of freedom. (b) Orientational degree of freedom. (c) Torsional degree of freedom.

Figure 2.4: Conformational degree of freedom.

2.4.3 Conformational modeling

A conformation refers to a combination of position, orientation, and torsions, if any. Figure 2.4 shows the positional, orientational and torsional degree of freedom. In (a), the two conformations in different colors only differ in their spatial position. One conformation can be transformed to the other simply by a spatial translation. In (b), the two conformations only differ in their orientation. One conformation can be transformed to the other simply by a spatial rotation. In (c), the two conformations only differ by one torsion. One conformation can be transformed to the other by a rotation along the corresponding rotatable bond, shown in the center of the subfigure, by a certain degree applied only to the child branches of that rotatable bond.

In the root or any branch, the atomic positions are relative to each atom because of no rotatable bond therein. Hence it is possible to model the conformation of the root or a branch

by the the 3D coordinate of a reference atom as well as a normal vector to represent the orientation, while the positions of the other atoms in the same branch can be recovered with the relative atomic positions. The reference atom can be chosen to be the atom connecting the parent branch, i.e. the second atom involved in the rotatable bond or the Y atom in the line of "BRANCH X Y". It is always the first atom of the current branch if the PDBQT file is produced by AutoDockTools4 [32]. As for the representation of orientation, a normalized quaternion typically features better numerical stability than a directional triplet.

Once the conformation of the root or a branch is determined, the orientation of a child branch can be derived by that of the parent branch and a torsion, which is essentially the rotating angle along the connecting rotatable bond and thus falls in the range of $[-\pi, \pi]$. The position of the reference atom of the child branch (the Y atom) is fixed relative to the first atom involved in the connecting rotatable bond (the X atom) and is invariant of the torsion applied. Therefore, the conformation of a child branch can be uniquely identified by the conformation of its parent branch and a torsion value. Eventually in a cascade way, the conformation of a flexible ligand can be modeled by the conformation of the root and a set of torsion values as many as the number of rotatable bonds.

Mathematically, a ligand conformation can be modeled by a numerical vector $\mathbf{c} = (x, y, z, q_0, q_1, q_2, q_3, t_1, t_2, ..., t_n)$, where (x, y, z) represents the position of the reference atom (the first

atom) of the root, (q_0, q_1, q_2, q_3) represents the orientation of the root, with $q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1$, and $(t_1, t_2, ..., t_n)$ represents the torsions of all child branches, with n being the number of rotatable bonds and $t_i \in [-\pi, \pi]$ for $i \in [1, n]$. The conformational degree of freedom is at least 6, 3 from the position and 3 from the orientation. Nowadays modern drugs usually have 4 or more torsions, so the conformational degree of freedom is generally at least 10. In other words, there are generally at least 10 variables to optimize during conformational sampling.

2.4.4 Scoring function

The scoring function estimates the binding affinity given a conformation (equation (2.1)). The binding affinity predicted by idock is expressed in terms of free energy. The lower the free energy, the higher the binding affinity.

$$e = f(\mathbf{c}) = f(x, y, z, q_0, q_1, q_2, q_3, t_1, t_2, ..., t_n) \qquad (2.1)$$

Both idock and Vina share the same scoring function, which consists of a conformation-dependent part and a conformation-independent part. The conformation-dependent part is a weighted sum of five terms over all the pairs of atom i and atom j that can move relative to each other. It is calculated from equations (2.2) and (2.3) where t_i and t_j are the atom types of i and j respectively, and r_{ij} is their interatomic distance. The five terms are calculated from equations (2.4) to (2.8) where d_{ij} is the surface distance calculated from equation (2.9) where R_{t_i} and R_{t_j}

are the Van der Waals radii of t_i and t_j respectively (Figure 2.5). All the units are in Å. The weighting coefficients and the cut off at $r_{ij} = 8$Å of the five terms are borrowed from Vina. The optimization algorithm tries to find the global minimum of e and other low-scoring conformations, which it ranks subsequently.

$$e = \sum_{i<j} e_{ij} \tag{2.2}$$

$$\begin{aligned} e_{ij} &= (-0.035579) * Gauss_1(t_i, t_j, r_{ij}) \\ &+ (-0.005156) * Gauss_2(t_i, t_j, r_{ij}) \\ &+ (+0.840245) * Repulsion(t_i, t_j, r_{ij}) \\ &+ (-0.035069) * Hydrophobic(t_i, t_j, r_{ij}) \\ &+ (-0.587439) * HBonding(t_i, t_j, r_{ij}) \end{aligned} \tag{2.3}$$

$$Gauss_1(t_i, t_j, r_{ij}) = e^{-(d_{ij}/0.5)^2} \tag{2.4}$$

$$Gauss_2(t_i, t_j, r_{ij}) = e^{-((d_{ij}-3)/2)^2} \tag{2.5}$$

$$Repulsion(t_i, t_j, r_{ij}) = \begin{cases} d_{ij}^2 & \text{if } d_{ij} < 0 \\ 0 & \text{if } d_{ij} \geq 0 \end{cases} \tag{2.6}$$

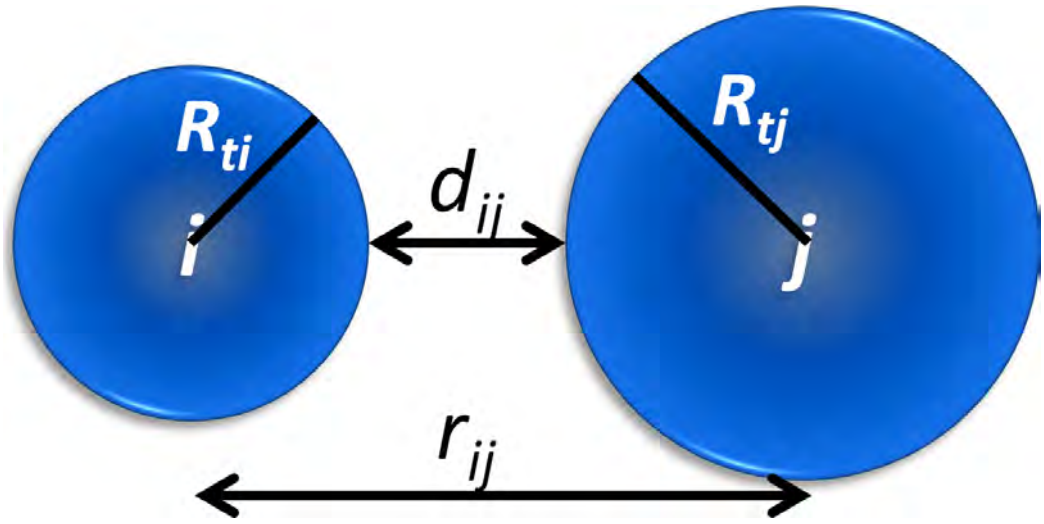


Figure 2.5: Relationship between surface distance d_{ij} and interatomic distance r_{ij}.

$$Hydrophobic(t_i, t_j, r_{ij}) = \begin{cases} 1 & \text{if } d_{ij} \leq 0.5 \\ 1.5 - d_{ij} & \text{if } 0.5 < d_{ij} < 1.5 \\ 0 & \text{if } d_{ij} \geq 1.5 \end{cases} \quad (2.7)$$

$$HBonding(t_i, t_j, r_{ij}) = \begin{cases} 1 & \text{if } d_{ij} \leq -0.7 \\ d_{ij}/(-0.7) & \text{if } -0.7 < d_{ij} < 0 \\ 0 & \text{if } d_{ij} \geq 0 \end{cases} \quad (2.8)$$

$$d_{ij} = r_{ij} - (R_{t_i} + R_{t_j}) \quad (2.9)$$

The conformation-dependent part can be seen as the sum of intermolecular and intramolecular contributions. Hence equation (2.2) can be rewritten into equation (2.10) where e_{inter} is

the summation over all the heavy atoms between the protein and the ligand, and e_{intra} is the summation over all the 1-4 ligand heavy atoms that are separated by at most three consecutive covalent bonds and can move relative to each other.

$$e = e_{inter} + e_{intra} \tag{2.10}$$

The conformation-independent part penalizes e_{inter} for ligand flexibility. The predicted free energy of the kth conformation for output, denoted as e'_k, is calculated from equation (2.11) where k is the subscript for conformation, e_k is the conformation-dependent score of the kth conformation calculated from equation (2.2), $e_{intra,1}$ is the e_{intra} of the first, i.e. lowest-scoring conformation, $N_{InactTors}$ is the number of inactive torsions (i.e. hydroxyl groups —OH, amine groups —NH_2 and methyl groups —CH_3), and $N_{ActTors}$ is the number of active torsions (other than the three types) of the ligand. Note that $e_{intra,1}$, rather than $e_{intra,k}$, acts as subtrahend in order to preserve the ranking.

$$e'_k = \frac{e_k - e_{intra,1}}{1 + 0.05846 * (N_{ActTors} + 0.5 * N_{InactTors})} \tag{2.11}$$

The value of e_{ij} is basically a function of three variables, namely t_i, t_j, and r_{ij}. These three variables have both a known lower bound and a known upper bound, so it is possible to precalculate the scoring function. Since there are 15 atom types implemented in idock, the pair of t_i and t_j can have 120 (=15*16/2)

different combinations. Since r_{ij} is cut off at 8Å, idock uniformly samples 16,384 points in range [0, 8] to turn the continuous domain into a concrete domain, resulting in an average absolute error of merely 0.002 kcal/mol. During program initialization, idock precalculates e_{ij} from equation (2.3) for 120*16,384 possible combinations of t_i, t_j, and r_{ij}. During optimization, idock approximates the true value of e_{ij} by direct assignment rather than linear interpolation so as to fast evaluate e_{ij} at the cost of a little bit longer precalculation time and a bit more memory storage.

2.4.5 Grid maps

Grid maps are often built in order to fast evaluate e_{inter}. A grid map of atom type t is constructed by placing virtual probe atoms of atom type t along the X, Y, Z dimensions of the search box at a certain granularity. Figure 2.6 illustrates a grid map, where the virtual probe atoms are shown in purple, and the surrounding protein residuals are shown in ball-and-stick representation. The e_{inter} value of these probe atoms are precalculated, so the e_{inter} value of a ligand heavy atom can be approximated in some way. In Vina, the grid map granularity is hard coded to be 0.375Å, and the approximation is done by linear interpolation of the 8 corner probe atoms of the residing subbox. This kind of interpolation involves reading of 8 e_{inter} values, computation of 3 α values, 12 floating-point subtractions, 24 floating-point multiplications, and 7 floating-point additions, which turned out to

be a performance bottleneck when we profiled Vina. In contrast, idock exposes grid map granularity as an optional program argument with a tuned default value of 0.15625Å. Likewise, due to a higher density of probe atoms, idock substitutes direct assignment for linear interpolation for much faster evaluation of e_{inter} at the cost of longer precalculation time and larger memory storage. Therefore, grid maps are built on the fly only when necessary and abstracted into parallel tasks, which are then distributed to the thread pool for concurrent execution.

2.4.6 Optimization algorithm

Both idock and Vina use Monte Carlo algorithm for global optimization and Broyden-Fletcher-Goldfarb-Shanno (BFGS) [70] Quasi-Newton method for local optimization. Figure 2.7, modified from the Figure 2 in [56], shows this optimization procedure. A succession of steps consisting of a mutation and a BFGS local optimization are taken, with each step being accepted according to the Metropolis criterion. These steps are repeated over N iterations, where N correlates to the complexity of the ligand regarding, for instance, the number of heavy atoms and the number of torsions. BFGS approximates the inverse Hessian matrix of the scoring function. So it uses not only the value of the scoring function but also its gradient, which are the derivatives of the scoring function with respect to the position and orientation of the ligand, and the torsions for the active rotatable bonds in the ligand. A BFGS iteration derives a descent

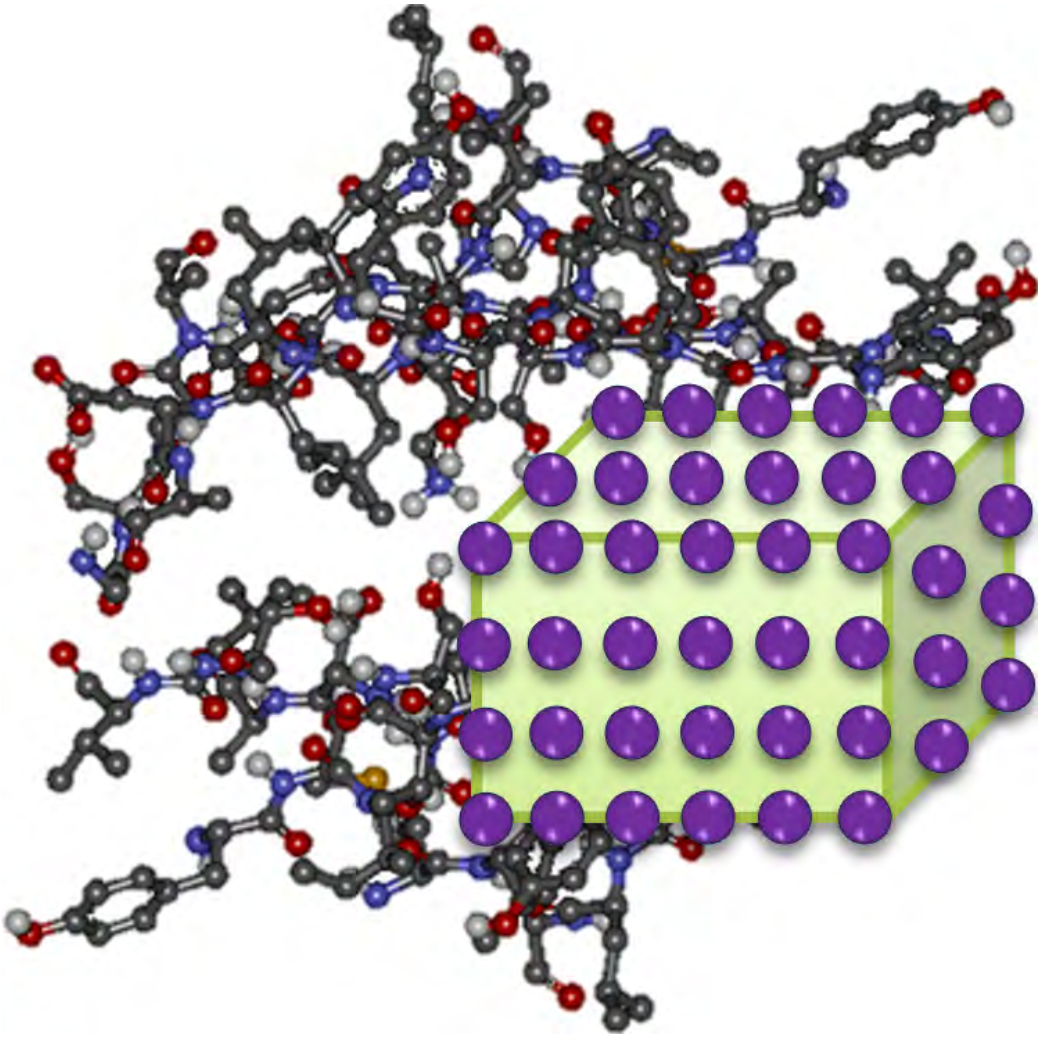


Figure 2.6: Grid map for fast evaluation of e_{inter}

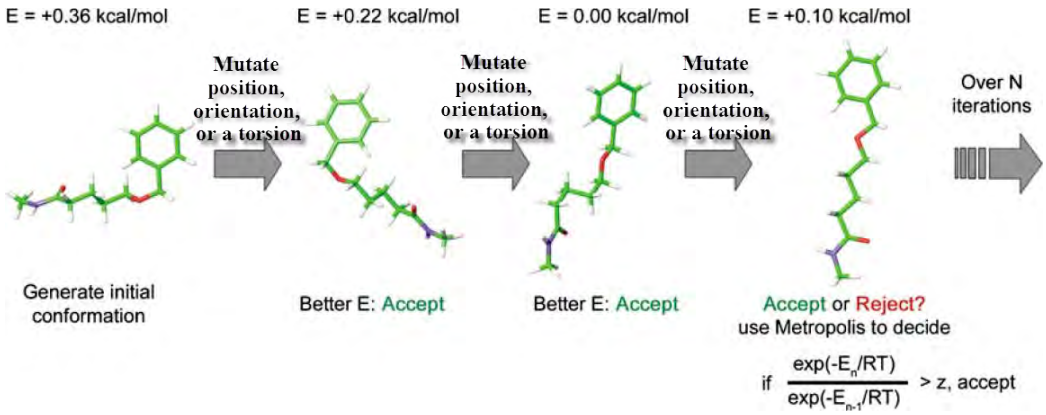


Figure 2.7: Monte Carlo optimization algorithm.

direction from the approximate inverse Hessian matrix, then derives a step length along the descent direction by line search, and updates the approximation of inverse Hessian matrix. Both programs achieve multithreading by concurrently running multiple independent Monte Carlo tasks starting from random initial conformations.

Though both programs share similar optimization algorithms, their fundamental implementations differ considerably. Compared with Vina, the Monte Carlo iterations in idock are far fewer and the BFGS iterations are more. On one hand, the fewer number of Monte Carlo iterations is compensated by a larger number of parallel Monte Carlo tasks, which is 64 by default in idock compared to 8 in Vina, guaranteeing better conformational diversity and higher CPU utilization on modern multi-core computers. On the other hand, the stopping criterion of BFGS local optimization does not depend on an estimated number of iterations, which is the case in Vina, but depends on the outcome of line search. The BFGS local optimization

stops if and only if no appropriate step length can be obtained by line search, thus decreasing the probability of missing local optimums.

2.4.7 Native support of virtual screening

Vina is optimized for single-ligand docking rather than virtual screening. When it comes to docking a large pool of ligands, Vina has to be invoked multiple times, repeatedly parsing the same protein and creating the same grid maps, thus degrading performance.

idock supports virtual screening in a native manner. It docks a directory of ligands instead of a single ligand, and reuses protein and grid maps (note the loop in the flowchart in Figure 2.2). Given a very large amount of ligands to dock, idock indirectly supports two-phase virtual screening via two consecutive runs. In the first run, idock performs coarse but fast virtual screening without writing any conformations to file, aiming to quickly shortlist a few candidate compounds. This can be done by setting the grid map granularity to a coarse value and setting the maximum number of output conformations to zero. In the second run, idock performs fine but slow virtual screening with a significantly larger number of Monte Carlo tasks per ligand, writing as many conformations to file as possible and aiming to refine the predicted free energy as well as predicted conformation of candidate compounds. Such a two-phase docking methodology can remarkably reduce overall execution time while avoiding

the risk of filtering out potentially promising compounds, controlling the false negative rate at an acceptable level.

2.4.8 Detection of inactive torsions

idock automatically detects and deactivates certain torsions which are presented and activated in the input file in PDBQT format but have no impact on the overall scoring, such as hydroxyl groups —OH, amine groups —NH_2 and methyl groups —CH_3, because they only rotate the hydrogens. Figure 2.8 shows an example ligand which contains 4 active torsions defined by the python script *prepare_ligand4.py* provided by AutoDock Tools [32]. Two of them, highlighted in yellow, only rotate hydrogens and thus have no contributions to the scoring. They are reclassified as inactive torsions and deactivated while being parsed in idock. This kind of automatic deactivation of pre-activated torsions reduces the torsional degrees of freedom to optimize in the local optimization step (equation (2.1)), leading to easier finding of local minimums.

2.4.9 Implementation tricks

idock implements a lightweight thread pool in order to reuse threads and maintain a high CPU utilization throughout the entire screening procedure. During program initialization, idock creates a thread pool of N threads, where N is the number of CPU cores automatically detected, or can be specified by user via a command line argument. The threads sleep while idle.

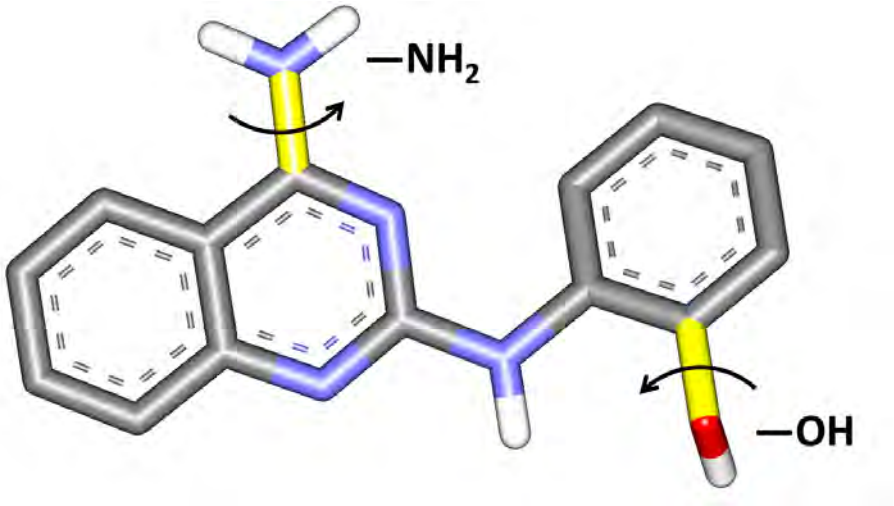


Figure 2.8: An example of inactive torsions, highlighted in yellow.

When tasks arrive, the threads compete for tasks. The thread that completes its current task will automatically fetch a pending one to execute until all are done. Synchronization is implemented to ensure the full completeness of tasks and availability of results. The task here is an abstract concept in programming sense and can be instantiated either as scoring function tasks, grid map tasks or Monte Carlo tasks. To be exact, the thread pool indeed parallelizes the precalculation of scoring function, the creation of grid maps, and the execution of Monte Carlo tasks.

idock implements a lightweight thread-safe progress bar to report progress every 10% Monte Carlo tasks per ligand. idock better supports rvalue references and move semantics in C++11 to boost performance. idock flattens the tree-like recursive data structure of ligand as used in AutoDock Vina into simple linear array structure to ensure a high data cache hit rate and

easy coding. idock accelerates the assignment of atom types by making use of residue information for the protein and branch information for the ligand.

idock supports reading and writing compressed ligand files with in gzip/bzip2 format, resulting in a file footprint as low as just one eighth of the raw size using gzip. This new functionality turns out to be quite handy given an enormous amount of ligands to dock.

Both idock and Vina support 16 common chemical elements, which are H (hydrogen), C (carbon), N (nitrogen), O (oxygen), F (fluorine), Mg (magnesium), P (phosphorus), S (sulfur), Cl (chlorine), Ca (calcium), Mn (manganese), Fe (iron), Zn (zinc), Se (selenium), Br (bromine), and I (iodine). idock adds 9 additional elements, which are Na (sodium), K (potassium), Co (cobalt), Ni (nickel), Cu (copper), Sr (strontium), Cd (cadmium), Hg (mercury), and As (arsenic). Supporting these additional elements is helpful because Na^+, K^+, Co^{2+}, Ni^{2+}, Cu^{2+}, Sr^{2+}, Cd^{2+}, and Hg^{2+} ions are present in some protein-ligand complexes, such as those with PDB ID of 1I2S, 1FPI, 1QCA, 1ELR, 1IBG, 2RIO, 1HSL, and 1AVN, respectively. Hence idock supports as many as 25 chemical elements, covering the majority of protein and ligand atom types.

idock outputs verbose information to docked PDBQT files, including total free energy normalized by torsional degree of freedom, total free energy, inter-ligand free energy, intra-ligand free energy, putative hydrogen bonds, and per-atom inter-ligand free energy. The normalized total free energy is used in ligand rank-

ing. The output of total free energy, inter-ligand free energy and intra-ligand free energy provides an alternative ranking option using derived efficiency indexes [71–73]. The per-atom inter-ligand free energy facilitates interaction hotspot determination, and helps improving potency by altering certain chemical moieties while retaining those critical for binding.

idock extracts the above records from docked PDBQT files, sorts them in the ascending order of normalized total free energy, and writes them to a CSV (Comma-Separated Vector) file for subsequent analysis. Users can derive their own efficiency indexes [71–73] and re-sort the records for their particular applications.

idock enables automatic recovery. While docking is in progress, in case the process gets killed accidentally and restarted some time later, which is common in computer cluster environments, idock not only resumes docking from the previous stopping point, skipping ligands that had been already docked in a previous run, but also detects and reports possible file content errors, ensuring all the output ligands are written appropriately.

2.5 Application

We applied idock to a real world drug discovery problem so as to demonstrate its utility in practice.

2.5.1 Background

At present, 25 drugs have been approved by US Food and Drug Administration (FDA) for the treatment of HIV/AIDS [74]. Among them, tenofovir disoproxil fumarate (TDF) is for the treatments of both human immunodeficiency virus (HIV) and hepatitis B virus (HBV). It is a nucleoside inhibitor of reverse transcriptase (RTs) of HIV and HBV. A considerably greater proportion of HBV+ recipients of TDF 300 mg once daily achieved a complete response at week 48 than oral adefovir dipivoxil 10 mg once daily [75]. TDF is also generally less expensive and more convenient to administer, as it does not require dosing on an empty stomach [76].

However, clinical feedback reveals that TDF exhibits strong side effects, causing osteomalacia and mitochondrial toxicity on the renal proximal tubule [77]. Justification of the side effects shows that 1) S-Adenosyl-L-Homocysteine hydrolase (SAHH), a highly conserved ubiquitous enzyme that catalyzes the hydrolysis of S-Adenosyl-L-Homocysteine (SAH) into adenosine and homocysteine, is affected, leading to defect in DNA methylation-dependent gene silencing [78]. SAHH inhibitor has signs of immunosuppressive activity [79]. 2) Adenosine deaminase (ADA) is inhibited, resulting in reduced breakdown of adenosine from food and decreased turnover of nucleic acids in tissues [80]. ADA inhibitor 2'-deoxycoformycin (dCF) shows signs of hepatic and adrenal toxicity [81]. 3) Purine nucleoside phosphorylase is also inhibited. Figure 2.9, reprinted from Sigma-Aldrich Co., shows

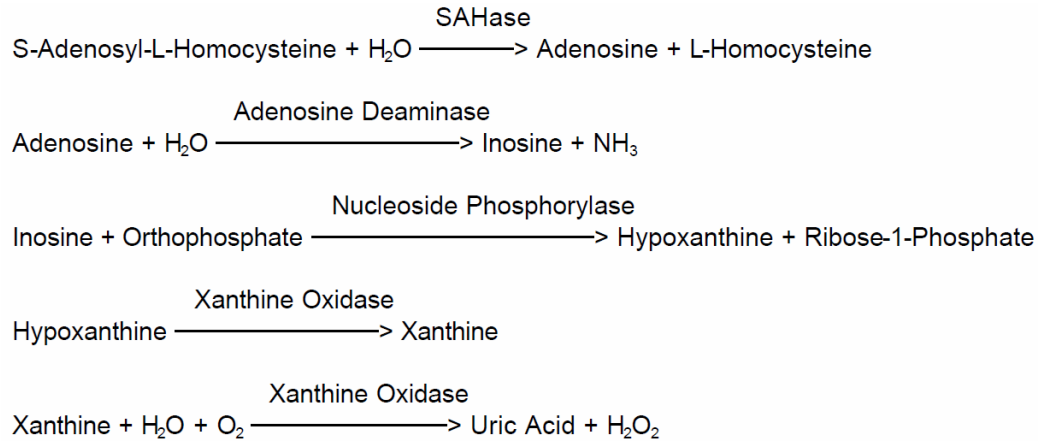


Figure 2.9: Enzymatic assay of S-Adenosyl-L-Homocysteine hydrolase.

the enzymatic assay of the above three enzymes, which break down SAH eventually into uric acid inside human body.

Table 2.1 shows the Tanimoto coefficients between TDF and the inhibitors of HIV RT, SAHH, ADA, and PNP using the Scitegic ECFP4 and Daylight fingerprints through the SEA database [82]. The higher the Tanimoto coefficient, the higher the similarity between two molecules in terms of chemical structure. It is always in the range of 0 to 1. The high Tanimoto coefficient values indicate that TDF is chemically similar to the inhibitors of SAHH, ADA, and PNP, hence TDF is likely to inhibit them in addition to HIV RT. For HIV-infected or HBV-infected patients who take TDF as the primary drug, it is unfortunate that these three essential enzymes are simultaneously inhibited by TDF.

	Scitegic ECFP4 fingerprint	Daylight fingerprint
HIV RT inhibitor	1.00	1.00
SAHH inhibitor	0.51	0.70
ADA inhibitor	0.42	0.75
PNP inhibitor	0.51	0.76

Table 2.1: Tanimoto coefficients between TDF and the inhibitors of HIV RT, SAHH, ADA, and PNP.

2.5.2 Problem definition

The problem is to discover promising compounds that inhibit HIV RT only without affecting SAHH, ADA, or PNP in order to minimize toxicity. From the computational perspective, it is equivalent to shortlisting candidates from existing ligand databases such that they are predicted to bind to HIV RT with a higher affinity but bind to SAHH, ADA, and PNP with a lower affinity. This was done by protein-ligand docking with idock and Vina.

2.5.3 Materials

The crystal structures of HIV RT, SAHH, ADA, and PNP were collected from the Protein Data Bank (PDB) [21, 22]. Protein-ligand complexes with PDB IDs of 2ZD1, 1LI4, 3IAR, and 3BGS were selected because they were crystallized at high resolutions (Table 2.2). Search spaces (box sizes) were then manually defined in cuboid shape to be large enough for ligands to freely translate and rotate inside.

10,928 ligands were collected from the clean drug like subset of the ZINC database [27, 28]. These ligands satisfy Lipinski's

PDB ID	Protein	Ligand	Resolution (Å)	Box size (Å^3)
2ZD1	HIV RT	T27	1.80	18 x 18 x 20
1LI4	SAHH	NAD	2.01	26 x 24 x 18
3IAR	ADA	3D1	1.52	22 x 16 x 16
3BGS	PNP	DIH	2.10	18 x 18 x 20

Table 2.2: Selected PDB entries for HIV RT, SAHH, ADA, and PNP.

Rule of Five [83] with the xLogP value of up to 5, the molecular weight between 150 Da and 500 Da, the number of hydrogen bond donors of up to 5, and the number of hydrogen bonds acceptors of up to 10.

2.5.4 Benchmarks

The benchmarks include 1) validation of Vina and idock to ensure their suitability for docking ligands against the four proteins, and 2) comparison of their virtual screening performance in terms of execution time, memory usage, predicted free energy, and predicted conformations.

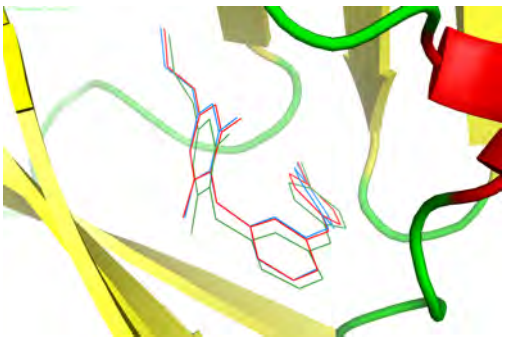
Vina x86 version 1.1.2 and idock x86_64 version 1.0 were used because they were the most recent versions of both programs at the time when the benchmarks were carried out. Both programs were run on desktop computers with dual Intel Xeon Quad Core 2.4GHz and 32GB RAM under Ubuntu 10.04.1 x86_64. The two CPUs support Intel's Hyper-Threading technology, so each computer consists of 8 physical cores and executes up to 16 logical threads simultaneously.

PDB ID	Protein	Ligand	Vina (Å)	idock (Å)
2ZD1	HIV RT	T27	0.465	0.555
1LI4	SAHH	NAD	0.537	0.593
3IAR	ADA	3D1	0.605	0.569
3BGS	PNP	DIH	0.756	1.170

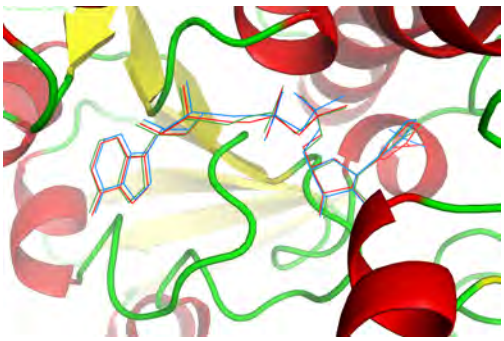
Table 2.3: RMSDs between the crystal and docked conformations of T27, NAD, 3D1, and DIH.

2.5.5 Program validation

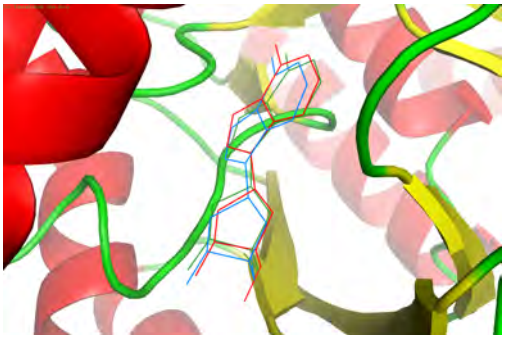
The four crystal ligands extracted from the four PDB complexes were conformationally randomized and redocked against their proteins by Vina and idock to see how much the predicted conformation would deviate from the crystal one. Figure 2.10 shows the four proteins in complex with their corresponding crystal and docked ligands. The ligands rendered in green are the crystal conformations, the ligands rendered in red are the ones docked by Vina, and the ligands rendered in blue are the ones docked by idock. To quantify the differences, Table 2.3 shows the root mean square deviations (RMSDs) between the crystal and docked conformations. The RMSDs are all below 2.0 Å, a publicly accepted positive control for correct bound structure prediction, indicating both programs are suitable for docking ligands against the four proteins. The RMSDs obtained by Vina are slightly better than those obtained by idock, especially for the case of PNP. This is probably due to the coarse estimation of e_{intra} in idock, which does not compute covalent bonds internally but simply relies on rotatable bonds to detect atom pair mobility.



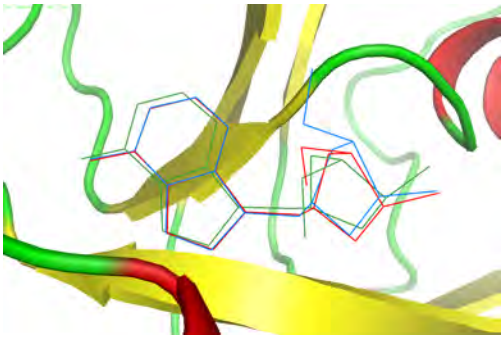
(a) HIV RT in complex with crystal and docked T27.



(b) SAHH in complex with crystal and docked NAD.



(c) ADA in complex with crystal and docked 3D1.



(d) PNP in complex with crystal and docked DIH.

Figure 2.10: HIV RT, SAHH, ADA, and PNP in complex with crystal and docked conformations of T27, NAD, 3D1, and DIH predicted by Vina and idock.

2.5.6 Virtual screening

Virtual screening of 10,928 drug-like ligands was then carried out. They were docked against the four proteins by Vina and idock. Because Vina can only dock one ligand in each run, a script containing 10,928 lines was generated and run instead, with each line being an execution of Vina to dock one individual ligand. Arguments to both programs were left as default. The GNU Time utility was used as a profiler.

Table 2.4 compares the execution time and memory usage of docking 10,928 drug-like ligands against HIV RT, SAHH, ADA, and PNP by Vina and idock. Maximum CPU utilization is 1600% due to Intel's Hyper-Threading technology. Vina required 428 to 504 CPU hours for one protein case, while idock required merely 88 to 184 CPU hours, resulting in a speedup of 2.5 to 4.8 and a screening performance of 1.3 drug-like ligands per CPU minute on average. In terms of elapsed time, the speedup was increased to as high as 6.3 to 10.4 because idock better utilized the CPU cores thanks to its efficient thread pool. idock also better utilized available memory to build grid maps at a high resolution and retained them throughout program execution. Even though idock consumed more memory than Vina, its maximum resident set size did not exceed 1.5 GB, hence idock can be run on mainstream desktop computers.

Table 2.5 summarizes the root mean square errors (RMSEs) of free energies and the root mean square deviations (RMSDs) of conformations predicted by both programs. The RMSEs of

Program	CPU Hours	Elapsed	CPU Util.	Max Mem Usage
HIV RT				
Vina	464	69:15:13	670%	126 MB
idock	162	10:57:46	1474%	856 MB
Ratio	2.9	6.3	0.45	0.15
SAHH				
Vina	460	78:53:59	582%	150 MB
idock	184	12:24:24	1484%	1,368 MB
Ratio	2.5	6.4	0.39	0.11
ADA				
Vina	504	74:22:37	677%	114 MB
idock	127	8:46:12	1452%	764 MB
Ratio	4.0	8.5	0.47	0.15
PNP				
Vina	428	62:19:55	687%	116 MB
idock	88	5:58:19	1479%	857 MB
Ratio	4.8	10.4	0.46	0.13
Average				
Vina	464	71:12:56	654%	124 MB
idock	140	9:31:40	1472%	961 MB
Ratio	3.3	7.5	0.44	0.13

Table 2.4: Execution time and memory usage of docking 10,928 drug-like ligands by Vina and idock.

Protein	RMSE (kcal/mol)	Avg RMSD (Å)	RMSD $\leq$ 2.0 Å
HIV RT	0.35	2.554	61%
SAHH	0.46	4.190	49%
ADA	0.33	2.620	59%
PNP	0.31	2.966	53%

Table 2.5: RMSEs of free energies and RMSDs of conformations predicted by Vina and idock.

free energies predicted by both programs vary from 0.31 to 0.46 kcal/mol, apparently less than 2.85 kcal/mol, the standard error obtained by Vina, indicating both programs predicted very similar free energies. For 27% to 40% of all the 10,928 ligands, the RMSD of the conformations predicted by both programs is equal to or less than 1.0 Å, and for 49% to 61%, the RMSD is equal to or less than 2.0 Å, indicating both programs predicted similar conformations for around half of the cases.

Finally in order to shortlist a few promising ligands that bind to HIV RT with a high affinity but bind to SAHH, ADA, and PNP with a low affinity, filtering criteria were set. Ligands whose predicted free energies against HIV RT are below or equal to -11.0 kcal/mol and whose predicted free energies against the other three proteins are above or equal to -8.5 kcal/mol were shortlisted in Table 2.6. Values are in kcal/mol unit.

The ZINC19888543 compound predicted by Vina and the ZINC44392991 compound predicted by idock were further investigated. Table 2.7 cites their predicted xLogP, number of hydrogen bond donors (HBD), number of hydrogen bond acceptors (HBA), molecular weight (MW), and number of rotatable

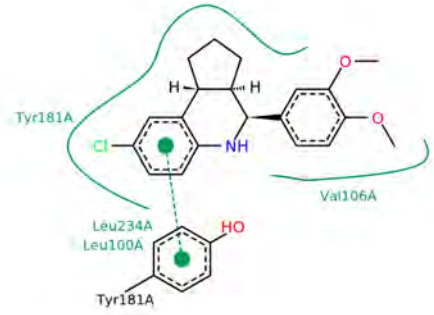
Ligand	HIV RT	SAHH	ADA	PNP
Vina				
ZINC04667184	-11.1	-6.6	-7.5	-7.5
ZINC06720921	-11.0	-7.4	-8.0	-8.4
ZINC14545253	-11.0	-6.5	-8.0	-7.6
ZINC19888543	-11.1	-7.9	-7.8	-7.8
ZINC26423182	-11.1	-6.4	-7.8	-7.4
ZINC49453017	-11.3	-7.1	-8.3	-7.6
ZINC60603133	-11.0	-7.9	-8.3	-7.4
idock				
ZINC03012460	-11.3	-8.3	-7.8	-7.3
ZINC04667184	-11.2	-7.6	-7.7	-8.0
ZINC44392991	-11.1	-7.7	-7.3	-8.1
ZINC49453017	-11.6	-7.3	-8.3	-7.8

Table 2.6: Shortlisted ligands.

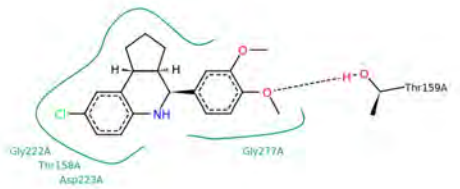
Ligand	xLogP	HBD	HBA	MW (Da)	NRB
ZINC19888543	4.48	1	3	342	3
ZINC44392991	4.24	1	6	391	6

Table 2.7: Chemical properties of ZINC19888543 predicted by Vina and ZINC44392991 predicted by idock.

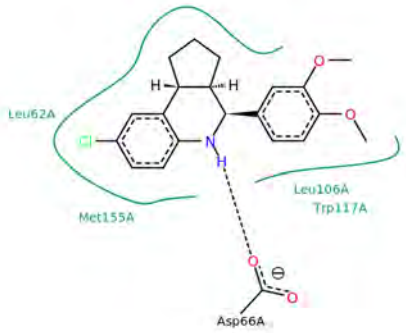
bonds (NRB) from the ZINC database [27, 28]. Figures 2.11 and 2.12, rendered by PoseView 1.0.0 [34], show the interaction charts of the docked ligands. Hydrogen bonds, salt bridges and metal interactions are highlighted as black dashed lines. Hydrophobic interactions are highlighted as green solid lines. Pi-Pi and Pi-cation interactions are highlighted as green dashed lines. It can be seen that ZINC19888543 was predicted to interact with HIV RT mainly through hydrophobic effects and pi interactions, whereas ZINC44392991 was predicted to interact with HIV RT mainly through pi interactions and hydrogen bonds.



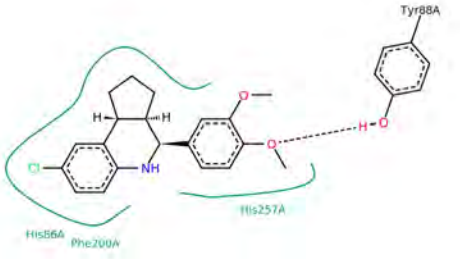
(a) HIV RT in complex with ZINC19888543.



(b) SAHH in complex with ZINC19888543.

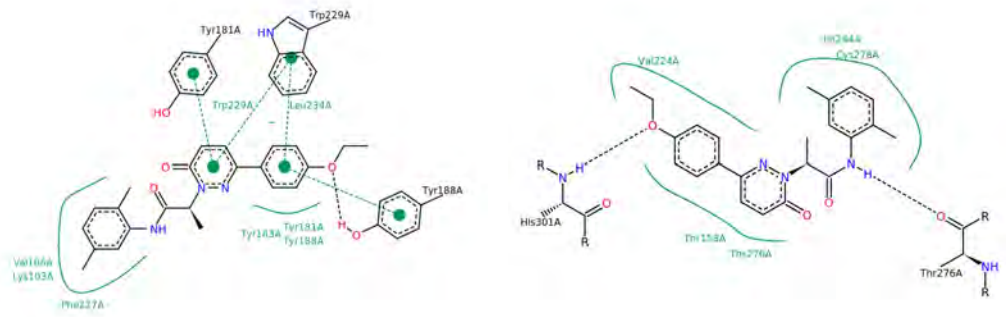


(c) ADA in complex with ZINC19888543.

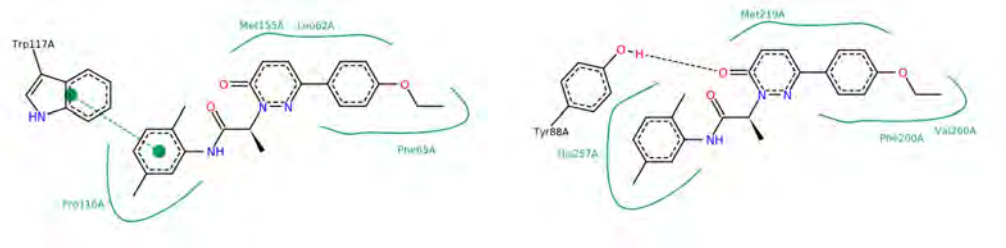


(d) PNP in complex with ZINC19888543.

Figure 2.11: Interaction charts of ZINC19888543 in complex with HIV RT, SAHH, ADA, and PNP.



(a) HIV RT in complex with ZINC44392991. (b) SAHH in complex with ZINC44392991.



(c) ADA in complex with ZINC44392991. (d) PNP in complex with ZINC44392991.

Figure 2.12: Interaction charts of ZINC44392991 in complex with HIV RT, SAHH, ADA, and PNP.

2.6 Discussion

Structure-based virtual screening has become a routine task in pharmaceutical institutions. Faster docking algorithms and implementations are highly desired. AutoDock Vina [8] is an exciting development due to its high performance and open source nature. Vina is mainly optimized for single ligand docking, however. In Vina's official forum, there are tremendous requests for the native support for virtual screening. Our development of idock perfectly fills in this gap.

idock has built-in support for virtual screening. It searches for ligands in a user-specified folder and docks them one by one. It reuses threads and grid maps across multiple ligands. idock inherits from Vina the accurate scoring function and the efficient optimization algorithm, and meanwhile introduces a fruitful of innovations in C++ implementations, data structures, numerical models, and Monte Carlo algorithms. idock implements its own thread pool to maintain a high CPU utilization throughout the entire screening procedure. It intensively utilizes modern C++11 techniques, particularly Rvalue references to avoid frequent reallocations of array data. It flattens Vina's tree-like recursive data structures into simple array structures to guarantee a high data cache hit rate. It automatically detects and deactivates certain torsions and thus reduces the dimension of variables to optimize and increases the probability of finding local optimums. idock's verbose output of free energy permits subsequent analysis using some ligand efficiency indexes [71–73]

for ranking. In terms of usability, idock has very similar input and output arguments as Vina, so it should be quite easy for existing AutoDock and Vina users to transit to idock.

Although the methodology of docking has advanced rapidly, there are still some major challenges yet to be solved [56]. These include the appropriate considerations of protein flexibility upon ligand binding, water molecules critical to the binding [84–86], metal ions in the binding site, etc. The protein is implicitly assumed rigid in this study. Another study [87] investigates the relationship between protein flexibility and binding free energy and presents some useful hints for understanding when, and to what extent, flexibility should be considered. Vina supports flexible protein docking, which has been proven helpful in some cases [66], by rotating flexible side-chains, i.e. technically speaking, by incorporating torsional variables from the protein side-chains for conformationally optimization. However, in idock we have not yet implemented flexible protein docking because it is a challenging task to adequately model the protein flexibility as well as there is a lack of commonly-accepted benchmarks. Users who need this kind of flexible docking should refer to Vina at the moment.

In the application example, we have shown that the docking scores can be used to discriminate between the proteins to which a ligand binds and the non-binding proteins. This approach has recently been applied to the prediction of adverse drug reactions [88]. The TarFisDock web server [89] in conjunction with the PDTD database [90] facilitates the identification of drug targets

of small molecules via a similar approach of docking and ranking.

2.7 Conclusions

We have developed idock, a multithreaded flexible ligand docking tool for structure-based virtual screening. It is capable of screening 1.3 drug-like ligands per CPU minute on average on a modern computer, making it a very competitive tool. Compared with state-of-the-art AutoDock Vina, idock achieved a speedup of 3.3 in terms of CPU time and a speedup of 7.5 in terms of elapsed time on average. But even so, it still required about 10 hours on average to dock 10,928 drug-like ligands against a certain protein, not to mention massive docking of millions of ligands. Virtual screening remains a time-consuming practice.

To demonstrate the utility of idock in real world drug discovery projects, we have performed structure-based virtual screening of 10,928 drug-like ligands to filter candidates that bind to HIV RT with a high affinity but bind to SAHH, ADA, and PNP with a low affinity in order to maximize drug potency while minimizing toxicity. Finally we have shortlisted a few promising compounds available for further clinical investigations.

2.8 Availability

idock 1.0 is free and open source under Apache License 2.0. Precompiled executables for 32-bit and 64-bit Linux, Windows, Mac OS X, FreeBSD and Solaris, 13 docking examples, and a

doxygen file for generating API documentations are available at https://github.com/HongjianLi/idock.

2.9 Future works

Porting idock to GPU (Graphics Processing Unit) using CUDA and OpenCL is one of our future directions, in view of the fact that the modern GPU has evolved from a fixed-function graphics pipeline to a programmable parallel processor with extremely high computational throughput and tremendous memory bandwidth at an affordable price. Performance evaluation of hybrid programming patterns for large CPU/GPU heterogeneous clusters has been carried out [91]. The recent six years have seen a fruitful of algorithms for computer-aided drug discovery being ported to the GPU and gaining orders of magnitude of speedup over single threaded CPU counterparts. To name a few, such GPU-accelerated applications include FTMap [92] for binding site mapping, CUDASW++2.0 [93] for protein database search, the leader and the spread algorithms [94] for compound selection, PIPER [95], PLANTS [44], GPUperTrAmber [96] and a transcription factor-DNA docking program [97, 98], a FFT-based tool [99, 100] and MEGADOCK 4.0 [101] for protein-protein docking, SIML [102], Tanimoto matrix calculation [103], and an all-to-all comparison [104] for chemical similarity calculation, OpenMM [105], MD-GPU [106], SPFP [107] and *ls1 mardyn* [108] for molecular dynamics, PAPER [109] for molecular shape comparison, gWEGA [110] for molecular superposi-

tion and shape comparison, a k-centers algorithm for clustering conformations [111], CAMPAIGN [112] for data clustering, and visualization [113], and GASPRNG [114] for scalable parallel random number generation.

On a different issue, well-studied proteins often have multiple crystallographically determined structures. How to appropriately perform ensemble docking [115–117] remains a challenge. Likewise, how to perform multiple ligand simultaneous docking (MLSD) [118, 119] is also an interesting topic of great potential. Heuristic modeling of torsion angle preferences [120, 121] and ligand flexibility [122] can remarkably reduce the optimization space.

□ **End of chapter.**

Chapter 3

istar: software as a service

Protein-ligand docking is a key computational method in the design of starting points for the drug discovery process. Performing large-scale docking requires tedious configurations of necessary tools and preparations of mandatory materials. Although a few online docking platforms exist, they neither support fine-grained ligand selection based on molecular properties and previewing the number of ligands to dock, nor be able to monitor job progress in real time. They also lack convenient visualization of docking results and straightforward output of compound suppliers.

We are intrigued to automate large-scale docking using our popular docking engine idock and thus have developed a publicly accessible web platform called istar. Without cumbersome software installation, users can submit jobs using our website. Our istar website supports 1) filtering ligands by desired molecular properties and previewing the number of ligands to dock, 2) monitoring job progress in real time, and 3) visualizing

ligand conformations and outputting free energy predicted by idock, binding affinity predicted by RF-Score, putative hydrogen bonds, and supplier information for easy purchasing, three useful features commonly lacked on other online docking platforms like DOCK Blaster or iScreen. We have collected 23,129,083 ligands from the All Clean subset of the ZINC database, and revamped our idock to version 2.0, further improving docking speed and accuracy, and integrating RF-Score as an alternative rescoring function.

To compare idock 2.0 with the state-of-the-art AutoDock Vina 1.1.2, we carried out a rescoring benchmark and a redocking benchmark on the 2,897 and 343 protein-ligand complexes of PDBbind v2012 refined set and CSAR NRC HiQ Set 24Sept2010 respectively, and an execution time benchmark on 12 diverse proteins and 3,000 ligands of different molecular weight. Results showed that, under various scenarios, idock achieved comparable success rates while outperforming AutoDock Vina in terms of docking speed by at least 8.69 times and at most 37.51 times. When evaluated on the PDBbind v2012 core set, our istar platform in combination with RF-Score managed to reproduce Pearson and Spearman correlation coefficients of as high as 0.855 and 0.859, respectively, between the experimental binding affinity and the predicted binding affinity of the docked conformation. idock@istar is freely available at http://istar.cse.cuhk.edu.hk/idock.

This was a collaborative project with Pedro J. Ballester from European Bioinformatics Institute, Cambridge, United Kingdom. It was published in *PLoS ONE* on 24 January 2014 [9].

3.1 Background

Protein-ligand docking predicts the preferred conformation and binding affinity of a small ligand as non-covalently bound to the specific binding site of a protein. Docking can therefore be used not only to determine whether a ligand binds, but also to understand how it binds. The latter is subsequently important to improve the potency and selectivity of binding. To date, there are hundreds of docking tools [56, 57]. The AutoDock series [8, 32, 69] is the most cited docking software in the research community, with over 9,400 citations according to Google Scholar. AutoDock contributed to the discovery of several drugs, including the first clinically approved inhibitor of HIV integrase [123]. Following its initial release, several parallel implementations were developed using either multithreading or computer cluster [124–126].

In 2009, AutoDock Vina [8] was released. As a whole new counterpart of AutoDock 4 [32], AutoDock Vina significantly improves the average accuracy of the binding mode predictions while running two orders of magnitude faster with multithreading [8]. It was compared to AutoDock 4 on selecting active compounds against HIV protease, and was recommended for docking large molecules [58]. Its functionality of semi-flexible protein docking by enabling flexibility of side-chain residues was evaluated on VEGFR-2 [66]. To further facilitate the usage of AutoDock Vina, auxiliary tools were subsequently developed, including a PyMOL (http://www.pymol.org) plugin for program

settings and visualization [59], a bootable operating system for computer clusters [60], a console application for virtual screening on Windows [61], and a GUI for virtual screening on Windows [62].

In 2011, inspired by AutoDock Vina, we developed idock 1.0 [7], a multithreaded virtual screening tool for flexible ligand docking. idock introduces plenty of innovations, such as caching receptor and grid maps in memory to permit efficient large-scale docking, revised numerical model for much faster energy approximation, and capability of automatic detection of inactive torsions for dimensionality reduction. When benchmarked on docking 10,928 drug-like ligands against HIV reverse transcriptase, idock 1.0 achieved a speedup of 3.3 in terms of CPU time and a speedup of 7.5 in terms of elapsed time on average compared to AutoDock Vina, making idock one of the fastest docking software.

Having released idock, we kept receiving docking requests from our colleagues and collaborators. They are mostly biochemists and pharmacologists, outsourcing the docking research to us after discovering pharmaceutical protein targets for certain diseases of therapeutic interest. Consequently, we had to grab the protein structure, do format conversion, define search space, set up docking parameters, and keep running idock in batch for months. Tedious enough, all the above work was done manually, resulting in very low research productivity.

A few online docking platforms already exist. DOCK Blaster [127] investigates the feasibility of full automation of protein-

ligand docking. It utilizes DOCK 3 [40] as the docking engine and ZINC [27, 28] as the ligand database. It also utilizes PocketPickker (CLIPPERS) [128] for binding pocket identification. iScreen [129] is a compacted web server for TCM (Traditional Chinese Medicine) docking and followed by customized *de novo* drug design. It utilizes PLANTS [42–44] as the docking engine and TCM@Taiwan [29] as the ligand database. It also utilizes LEA3D [130] for *de novo* ligand design. SwissDock [131] is a web server dedicated to the docking of small molecules on target proteins. It utilizes EADock DSS [132, 133] as the docking engine. FORECASTER [134] is a web interface consisting of a set of tools for the virtual screening of small molecules binding to biomacromolecules (proteins, receptors, and nucleic acids). It utilizes the flexible-target docking program FITTED [45] as the docking engine. Other web servers for docking can be found on the click2drug web portal http://www.click2drug.org/.

3.2 Motivation

The above docking platforms neither support fine-grained ligand selection based on molecular properties and previewing the number of ligands to dock, nor be able to monitor job progress in real time. They lack straightforward output of compound suppliers, a hurdle preventing users from purchasing high-rank compounds for further wet-lab verification. They also lack convenient and interactive online visualization of the docking results. We aimed to address these obstacles, and therefore developed a web plat-

form called istar in order to automate large-scale protein-ligand docking using our idock.

3.3 Objective

We strongly emphasized docking efficiency, which we believe is the most crucial factor for public large-scale docking platforms, so we tried every endeavor to optimize our docking engine idock as well as our system design. We accelerated even a single job execution by exploiting the computational resources of multiple machines, and thus implemented slice-level parallelism. We aimed to supply a sufficient amount of purchasable ligands for the users to select, and provide a mechanism for monitoring long-running job progress in real time. Furthermore, we adopted the robust RF-Score [10] version 3 as a rescoring function for accurate prediction of binding affinity. Last but not the least, we utilized modern website and database technologies to constitute a user-friendly web interface.

Nowadays, we intend to design istar as a versatile SaaS (Software as a Service) web platform rather than a conventional web server merely for docking purpose. SaaS is part of the nomenclature of cloud computing. It is a software delivery model in which software and associated data are centrally hosted on the cloud, typically accessed by users using a thin client via a web browser. We opt to abstract our software into easy-to-use services to promote their usage by a wide variety of users from different disciplines. In addition to idock for prospective structure-based

virtual screening, we have also hosted on istar several other services, for examples, USR (Ultrafast Shape Recognition) [19] and USRCAT (USR with Credo Atom Types) [20] for prospective ligand-based virtual screening, iview [11] for interactive WebGL visualization of protein-ligand complex, igrep [135] for approximate nucleotide sequence matching, and icuda as an introductory seminar series on CUDA programming.

3.4 Methods and materials

In the following subsections, we introduce our fast docking engine idock, our accurate rescoring function RF-Score, our modern web platform istar, and the experimental settings regarding datasets and benchmarks.

3.4.1 Docking engine idock

The input to idock includes a rigid receptor, a set of flexible ligands, and a cubic box, which is used to restrict the conformational space to a particular binding site of the receptor. The output from idock includes predicted conformations and their predicted binding affinity.

idock consists of two core components, a scoring function to predict binding affinity, and an optimization algorithm to explore the conformational space. idock inherits the same scoring function from AutoDock Vina. The idock score is made up of a conformation-dependent part and a conformation-independent part. The conformation-dependent part is a weighted sum of

five terms over all the pairs of atoms i and j that can move relative to each other, excluding 1-4 interactions, i.e. atoms separated by up to three consecutive covalent bonds. The sum is calculated from equations (3.1) and (3.2) where t_i and t_j are the atom types of i and j respectively, and r_{ij} is their interatomic distance with a cutoff at $r_{ij} = 8$Å. The five terms are calculated from equations (3.3) to (3.7) where d_{ij} is the surface distance calculated from equation (3.8) where R_{t_i} and R_{t_j} are the Van der Waals radii of t_i and t_j respectively. All terms are in Å units. The first three terms account for steric interactions, the fourth term accounts for hydrophobic effect, and the fifth term accounts for hydrogen bonding. Metal ions are treated as hydrogen bond donors. The weighting coefficients are derived from linear regression on the PDBbind [136–138] v2007 refined set (N = 1,300). The optimization algorithm attempts to find the global minimum of e and other low-scoring conformations, which it then ranks.

$$e = \sum_{i<j} e_{ij} \tag{3.1}$$

$$\begin{aligned} e_{ij} &= (-0.035579) \times Gauss_1(t_i, t_j, r_{ij}) \\ &+ (-0.005156) \times Gauss_2(t_i, t_j, r_{ij}) \\ &+ (+0.840245) \times Repulsion(t_i, t_j, r_{ij}) \\ &+ (-0.035069) \times Hydrophobic(t_i, t_j, r_{ij}) \\ &+ (-0.587439) \times HBonding(t_i, t_j, r_{ij}) \end{aligned} \tag{3.2}$$

$$Gauss_1(t_i, t_j, r_{ij}) = e^{-(d_{ij}/0.5)^2} \tag{3.3}$$

$$Gauss_2(t_i, t_j, r_{ij}) = e^{-((d_{ij}-3)/2)^2} \tag{3.4}$$

$$Repulsion(t_i, t_j, r_{ij}) = \begin{cases} d_{ij}^2 & \text{if } d_{ij} < 0 \\ 0 & \text{if } d_{ij} \geq 0 \end{cases} \tag{3.5}$$

$$Hydrophobic(t_i, t_j, r_{ij}) = \begin{cases} 1 & \text{if } d_{ij} \leq 0.5 \\ 1.5 - d_{ij} & \text{if } 0.5 < d_{ij} < 1.5 \\ 0 & \text{if } d_{ij} \geq 1.5 \end{cases} \tag{3.6}$$

$$HBonding(t_i, t_j, r_{ij}) = \begin{cases} 1 & \text{if } d_{ij} \leq -0.7 \\ d_{ij}/(-0.7) & \text{if } -0.7 < d_{ij} < 0 \\ 0 & \text{if } d_{ij} \geq 0 \end{cases} \tag{3.7}$$

$$d_{ij} = r_{ij} - (R_{t_i} + R_{t_j}) \tag{3.8}$$

The conformation-dependent part can be seen as the sum of inter-molecular and intra-molecular contributions. Hence equation (3.1) can be rewritten into equation (3.9) where e_{inter} is the summation over all the heavy atom pairs between the receptor and the ligand, and e_{intra} is the summation over all the non 1-4 heavy atom pairs of the ligand.

$$e = e_{inter} + e_{intra} \tag{3.9}$$

The conformation-independent part penalizes e_{inter} for ligand flexibility. The predicted free energy of the kth conformation for output, denoted as e'_k, is calculated from equation (3.10) where k is the subscript for conformation, e_k is the conformation-dependent score of the kth conformation calculated from equation (3.1), $e_{intra,1}$ is the e_{intra} of the first, i.e. lowest-scoring conformation, $N_{ActiveTors}$ is the number of active torsions and $N_{InactiveTors}$ is the number of inactive torsions of the ligand. A torsion is called inactive in this context if its arbitrary values do not affect the overall output of the scoring function. Note that $e_{intra,1}$, rather than $e_{intra,k}$, is subtracted in order to preserve the ranking of predicted conformations.

$$e'_k = \frac{e_k - e_{intra,1}}{1 + 0.05846 \times (N_{ActiveTors} + 0.5 \times N_{InactiveTors})} \tag{3.10}$$

On one hand, in order to fast evaluate e_{ij}, idock precalculates some of its possible values. Note from equation (3.2) that e_{ij} is essentially a function of three variables, namely t_i, t_j, and r_{ij}, which all have known lower and upper bounds. There are 15 heavy atom types implemented in idock, so there are 15*16/2=120 different combinations of pairs of t_i and t_j. Since r_{ij} is cut off at 8Å, idock uniformly samples 65,536 r_{ij} values in the range [0, 8] and precalculates their e_{ij} values. Subsequently in building grid maps or calculating e_{intra}, given a combination

of t_i, t_j and r_{ij}, idock approximates the true value of e_{ij} simply by a table lookup, rather than by a table lookup followed by a linear interpolation in the case of AutoDock Vina.

On the other hand, in order to fast evaluate e_{inter}, idock precalculates its possible values by building grid maps. A grid map of atom type t is constructed by placing virtual probe atoms of atom type t along the X, Y, Z dimensions of the search box at a certain granularity. The e_{inter} value of these probe atoms are precalculated from equation (3.2). Subsequently in the conformational optimization stage, given a sampled conformation, idock approximates the true values of e_{inter} of ligand heavy atoms simply by a table lookup rather than by a table lookup followed by a linear interpolation in the case of AutoDock Vina. In fact, when we profiled AutoDock Vina, its linear interpolation of the 8 nearest corner probe atoms turned out to be a performance bottleneck because it involves 8 readings, 12 subtractions, 24 multiplications, and 7 additions. The grid granularity is hard-coded to be a coarse value of 0.375Å in AutoDock Vina, while in idock it is exposed as a program option for users to adjust accordingly and has a default fine value of 0.15625Å so as to complement the possible precision loss due to the removal of linear interpolation for the sake of performance.

Likewise in AutoDock Vina, idock also uses Broyden-Fletcher-Goldfarb-Shanno (BFGS) [70], a quasi-Newton algorithm, for local optimization. In each BFGS iteration, a conformational mutation and a line search are taken, with each sampled conformation being accepted according to the Metropolis criterion.

The number of iterations correlates to the complexity of the ligand in terms of the number of heavy atoms and the number of torsions. BFGS approximates the inverse Hessian matrix. In other words, it uses not only the value of the scoring function but also its gradient, which are the derivatives of the scoring function with respect to the position, orientation and torsions of the ligand. Although both programs share similar optimization algorithms, their internal implementations differ substantially. In idock, the BFGS local optimization stops if and only if no appropriate step length can be obtained by line search, thus increasing the probability of finding optimal local minimums. More optimization runs with fewer number of BFGS iterations are executed, better balancing high conformational diversity and short execution time.

idock introduces a novel feature that can automatically detect and deactivate certain torsions which are activated in the input file but indeed have no impact on the overall scoring. These are the torsions of hydroxyl groups —OH, amine groups —NH_2 and methyl groups —CH_3, where mere hydrogens will be rotated and therefore have no contributions to the idock score. idock is capable of re-classifying them as inactive torsions during parsing, thus reducing the dimension of variables to optimize in subsequent BFGS iterations.

idock encapsulates many other improvements. Please refer to its change log for a complete list of new features and bugfixes.

3.4.2 Scoring function RF-Score

RF-Score [10] is a member of a new class of scoring functions that use non-parametric machine learning approach to predict binding affinity in an entirely data-driven manner. It is the first application of RF (Random Forests) [139] to predicting protein-ligand binding affinity. It was rigorously shown [10] to perform better than 16 classical scoring functions in ranking protein-ligand complexes according to their predicted binding affinity. It was also shown to be useful in the discovery of new molecular scaffolds in antibacterial hit identification [140].

In RF-Score, each feature comprises the number of occurrences of a particular protein-ligand atom type pair interacting within a certain distance range. Four common atom types for the protein (C, N, O, S) and nine common atom types for the ligand (C, N, O, F, P, S, Cl, Br, I) constitute a vector of 36 features, and the distance cutoff is chosen to be as sufficiently large as 12Å so as to implicitly capture solvation effects.

RF grows each binary tree without pruning using the CART algorithm [141] from a bootstrap sample of the training data. It selects the best split at each node of the tree from a typically small number of randomly chosen features, and terminates splitting a node when it contains no more than five samples. The prediction from an individual tree is the arithmetic mean of its sample outputs in the traversed leaf node. The final prediction is the arithmetic mean of the individual predictions of all the trees in the forest. The performance of RF-Score does not vary

significantly with the number of trees beyond a certain threshold. The common practice is to use 500 as a sufficiently large number of trees.

The original version of RF-Score [10] was trained on PDBbind v2007 refined set less the core set (N = 1,105). We retrained RF-Score on PDBbind v2013 refined set (N = 2,959) for prospective prediction purpose, and integrated it into our istar platform to re-score predicted conformations. We also implemented a consensus score as the average effect of idock score and RF-Score. Mathematically, equations (3.11) to (3.13) relate equilibrium constant K_{eq} and dissociation constant K_d with Gibbs free energy ΔG, where R is gas constant $R = 1.9858775 \times 10^{-3} kcal/mol$ and T is absolute temperature.

$$\Delta G = -RT \ln K_{eq} \tag{3.11}$$

$$K_d = \frac{1}{K_{eq}} \tag{3.12}$$

$$pK_d = -\log_{10} K_d \tag{3.13}$$

Assuming $T = 298.15K$ at room temperature, plugging equations (3.12) and (3.13) into (3.11) yields

$$pK_d = -0.73349480509 \times \Delta G \tag{3.14}$$

Equation (3.14) transforms the predicted free energy output by idock in $kcal/mol$ into binding affinity in pK_d unit. The con-

sensus score is thus defined in equation (3.15) so that it directly reflects the predicted potency in pK_d unit.

$$ConsensusScore = 0.5 \times (-0.73349480509 \times idockScore + RFScore) \tag{3.15}$$

3.4.3 Web platform istar

Figure 3.1 shows the overall architecture of istar. There are five major components: a website, a web server, a database management system, several workstations, and a network file system. Under typical circumstances, a user browses our website and submits a job. The web server first validates user input and then saves it into the database. Several workstations keep running daemons, which fetch jobs from the database and perform protein-ligand docking. Upon completion, the daemon sends a notification email to the user and writes the results to the network file system, which are cached as static contents by the web server. The user again browses our website to download or visualize the results, or monitor job progress. Our web server also supports REST API for developers to program against. Instructions on how to use the REST API can be found at http://istar.cse.cuhk.edu.hk/idock.

In our idock@istar web page (Figure 3.2), the first section displays summary of existing jobs and the second section allows new job submission. A job comprises compulsory fields and optional fields. The compulsory fields include a receptor in PDB

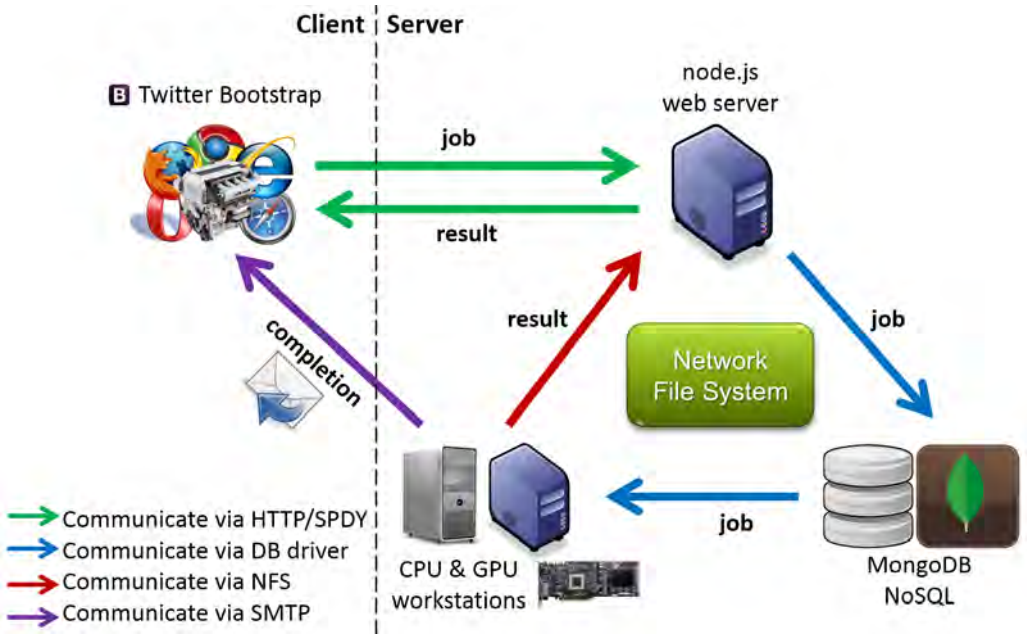


Figure 3.1: istar architecture.

format, a search space defined by a cubic box, a brief description about the job, and an email to receive completion notification. The optional fields include nine ligand filtering conditions. The nine ligand filtering conditions are molecular weight, partition coefficient xlogP, apolar desolvation, polar desolvation, number of hydrogen bond donors, number of hydrogen bond acceptors, topological polar surface area tPSA, net charge, and number of rotatable bonds. These nine molecular descriptors were directly retrieved from our data source, i.e. the ZINC database [27, 28], in which the nine descriptors were already precalculated. Note that although molecular mass in Dalton unit might be a more appropriate descriptor than molecular weight in g/mol unit, we stick to the latter in order to maintain consistency with ZINC, in which the g/mol unit is used for molecular weight.

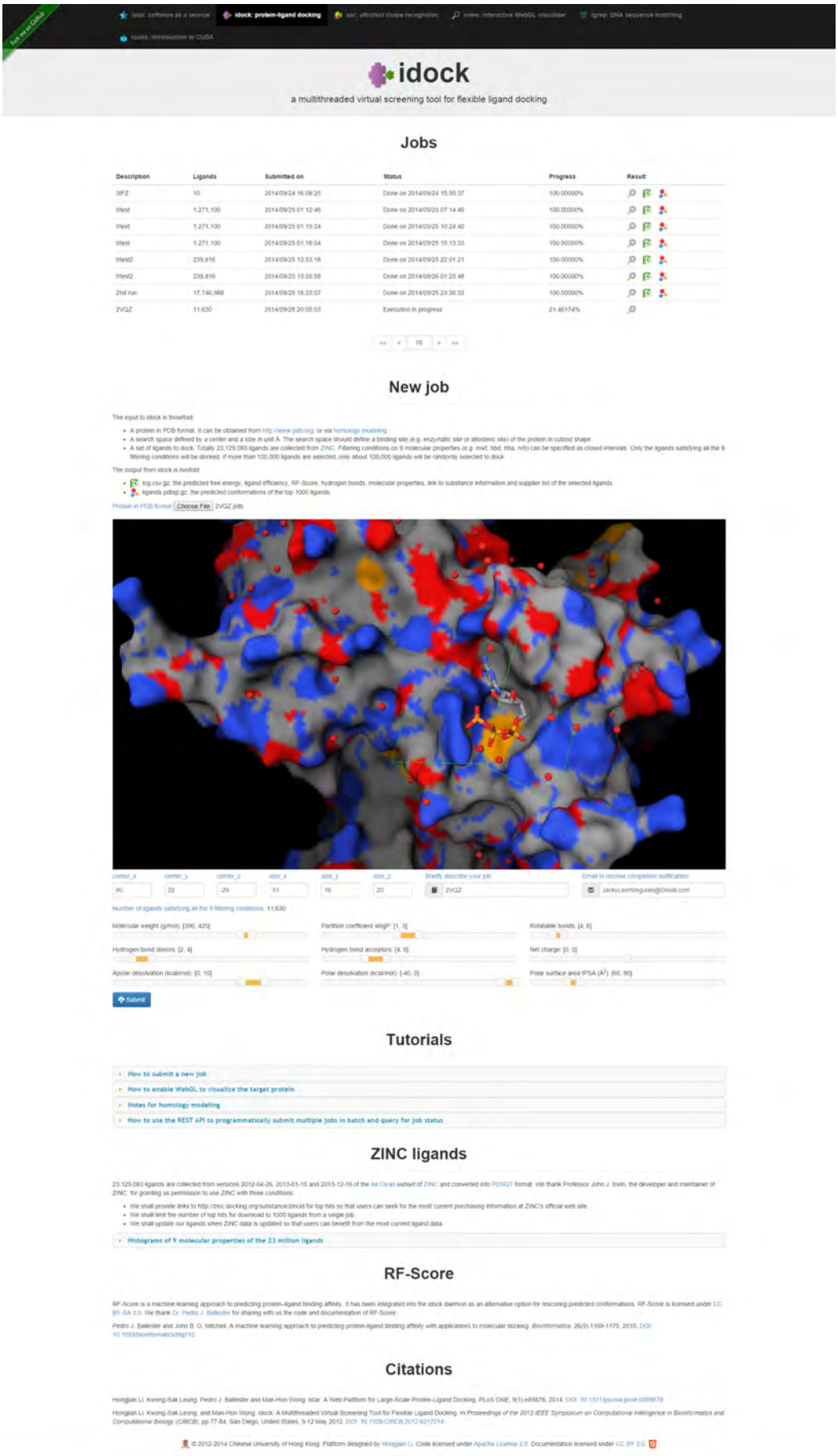


Figure 3.2: idock@istar web page.

We collected 23,129,083 ligands at pH 7 in mol2 format from versions 2012-04-26, 2013-01-10 and 2013-12-18 of the All Clean subset the ZINC database [27, 28] with explicit permission from its major developer and maintainer. The All Clean subset was constituted by applying the strict filtering rules specified at http://blaster.docking.org/filtering, e.g. aldehydes and thiols were removed. We then converted the entire database of ligands in batch into PDBQT format as used by idock. The huge number of 23 million ligands should be sufficient for most prospective applications. In case users need to screen their own ligand libraries, at present we recommend running idock locally on their computers.

istar supports ligand selection by desired molecular properties in a fine-grained manner and previewing the number of ligands to dock in real time (Figure 3.2, middle section). Users can move the nine sliders to filter ligands in the form of closed intervals. Only the ligands satisfying all the nine filtering conditions will be docked. Because of the relationship of logical and, in order to nullify a specific filtering condition, one may expand its closed interval to cover the entire possible range. We set up default values of the lower and upper bounds of the nine molecular properties for novices to get started quickly.

istar supports monitoring job progress in real time (Figure 3.2, top section). We composed a timer to regularly poll the server and automatically fetch and report the latest job progress every second without page refresh. Users can thus have a rough estimation in advance of how long the jobs will take and when

the jobs will complete. This feature is particularly handy when the jobs are long running, which is usually the case of large-scale docking.

istar outputs verbose information in PDBQT format (Figure 3.3). The first REMARK line describes the ZINC ID, molecular weight (g/mol), partition coefficient xlogP, apolar desolvation (kcal/mol), polar desolvation (kcal/mol), number of hydrogen bond donors, number of hydrogen bond acceptors, topological polar surface area tPSA ($Å^2$), net charge, and number of rotatable bonds of a selected ligand. The second REMARK line describes the SMILES representation. The third REMARK line describes the number of suppliers followed by their names, which conform to the nomenclature as used by ZINC. The subsequent REMARK lines describe the free energy and ligand efficiency predicted by idock, putative hydrogen bonds, binding affinity predicted by RF-Score, and consensus score in pK_d or pK_i unit. Columns 71 to 76 of the ATOM lines describe the predicted free energy of each atom. The individual atom contribution to the overall score facilitates the detection of protein-ligand interaction hotspots, and thus assists in *de novo* ligand design.

At the moment, we have deployed a machine equipped with Intel Xeon W3520 @ 2.66 GHz and 8GB DDR3 SDRAM to run the web server, and four identical machines each equipped with dual Xeon E5-2670 @ 2.6GHz and 128GB ECC DDR3 SDRAM to run the idock daemons. We have mounted a 2TB hard disk into our network file system to store docking jobs and results.

```
REMARK     09129460   421.81     3.68     1.09   -15.13   2   6  75   0   6
REMARK     c1ccc2c(c1)c(c[nH]2)CCC(=O)NCc3nnc4n3cc(cc4C(F)(F)F)Cl
REMARK     6 | IBScreen | Innovapharm Make-on-Demand | Molport | PubChem | Vitas-M | eMolecules
MODEL        1
REMARK       NORMALIZED FREE ENERGY PREDICTED BY IDOCK: -10.386 KCAL/MOL
REMARK            TOTAL FREE ENERGY PREDICTED BY IDOCK: -15.152 KCAL/MOL
REMARK     INTER-LIGAND FREE ENERGY PREDICTED BY IDOCK: -14.030 KCAL/MOL
REMARK     INTRA-LIGAND FREE ENERGY PREDICTED BY IDOCK:  -1.122 KCAL/MOL
REMARK           LIGAND EFFICIENCY PREDICTED BY IDOCK:  -0.484 KCAL/MOL
REMARK              HYDROGEN BONDS PREDICTED BY IDOCK:   2 | A:HIS235:O - N1 | A:PRO236:O - N1
REMARK         BINDING AFFINITY PREDICTED BY RF-SCORE:   7.701 pK
REMARK                                 CONSENSUS SCORE:   7.660 pK
ROOT
ATOM      1  C11 <0> d          52.218 -26.820  35.295  0.00  0.00    -0.383 C
ATOM      2  O1  <0> d          52.013 -25.664  35.598  0.00  0.00    -0.364 OA
ATOM      3  N2  <0> d          52.633 -27.699  36.228  0.00  0.00    -0.424 N
ATOM      4  H11 <0> d          52.795 -28.624  35.985  0.00  0.00     0.000 HD
ENDROOT
BRANCH   1   5
ATOM      5  C10 <0> d          52.006 -27.274  33.873  0.00  0.00    -0.217 C
ATOM      6  H9  <0> d          52.184 -26.439  33.195  0.00  0.00     0.000 H
ATOM      7  H10 <0> d          52.699 -28.084  33.643  0.00  0.00     0.000 H
BRANCH   5   8
```

Figure 3.3: Verbose output in PDBQT format.

3.4.4 Datasets

We evaluated and compared idock x86_64 v2.0 and AutoDock Vina x86 v1.1.2 from the perspectives of rescoring, redocking and execution time on three datasets, which are PDBbind [136–138], CSAR [142, 143] and ZINC [27, 28].

The PDBbind v2012 dataset contains a diverse collection of experimentally determined structures carefully selected from PDB (Protein Data Bank) [22, 144]. For each complex, the experimental binding affinity (either dissociation constant K_d, inhibition constant K_i, or half maximal inhibitory concentration IC_{50}) is manually collected from its primary literature reference, thus resulting in the general set of 9,308 complexes, with 7,121 being protein-ligand complexes. Out of them, the complexes with a resolution of 2.5Å or better, with known K_d or K_i values, and with ligand containing merely the common heavy atoms (C, N, O, F, P, S, Cl, Br, I) are filtered to constitute the refined

set of 2,897 complexes. These complexes are then clustered by protein sequence similarity using BLAST at a cutoff of 90%, and for each of the 67 resulting clusters with at least five complexes, the three complexes with the highest, median and lowest binding affinity are selected to constitute the core set of 201 complexes, whose experimental binding affinity spans 10 pK_d or pK_i units.

The CSAR (Community Structure Activity Resource) NRC HiQ Set 24Sept2010 contains 343 diverse protein-ligand complexes selected from existing PDB [22, 144] entries which have binding affinity (K_d or K_i) in Binding MOAD [145–147], augmented with entries from PDBbind [136–138]. Their binding affinity spans 12 pK_d units.

The ZINC database contains over 35 million purchasable small molecules in popular MOL2 and SDF formats.

3.4.5 Benchmarks

In the rescoring benchmark, we evaluated the capability of RF-Score, AutoDock Vina and idock of predicting the binding affinity as close to the experimental binding affinity as possible given a crystal protein-ligand complex. We compared their rescoring performance to 18 other scoring functions on the PDBbind v2007 core set ($N = 195$). The test set was then extended to two larger datasets, i.e. the PDBbind v2012 refined set ($N = 2{,}897$) [136–138] and the CSAR NRC HiQ Set 24Sept2010 ($N = 343$) [142, 143], to enable a more comprehensive comparison.

In the redocking benchmark, we evaluated the capability of

AutoDock Vina and idock of docking a randomized ligand conformation back to its crystal conformation as close as possible. We used the PDBbind v2012 refined set (N = 2,897), the PDBbind v2011 refined set (N = 2,455), and the CSAR NRC HiQ Set 24Sept2010 (N = 343), because they were the latest versions and contained the largest number of high-quality and diverse protein-ligand structures. We wrote a script to automatically define the search box first by finding the smallest cubic box that covers the entire ligand and then by extending the box in X, Y, Z dimensions by 10Å. Note that the 2rio entry of PDBbind contains two strontium ions, which are supported by idock but not by AutoDock Vina, we manually removed them before invoking AutoDock Vina. Both programs were also evaluated on the PDBbind v2012 core set (N = 201) in order to find potential impact factors on their performance. We used root mean square deviation $RMSD$ to measure the closeness between two conformations. The lower the $RMSD$ is, the closer the two conformations are. Usually the $RMSD$ value is calculated between the crystal and the docked conformations. Very often the $RMSD$ of 2.0Å is regarded as the positive control for correct bound structure prediction.

In the execution time benchmark, we collected 12 diverse proteins from the PDB (Protein Data Bank) database [22, 144], and 1000 ligands with a molecular weight of 200-300g/mol, 1000 ligands with a molecular weight of 300-400g/mol, and 1000 ligands with a molecular weight of 400-500g/mol from the All Clean subset of the ZINC database [27, 28]. The 3,000 ligands were

docked against the 12 proteins by AutoDock Vina and idock. Since AutoDock Vina can dock only one ligand in a run, three bash scripts each containing 1,000 lines were executed instead, with each line being an execution of AutoDock Vina to dock one single ligand. The GNU Time utility v1.7 was used as a profiler.

The three benchmarks were carried out on desktop computers with Intel Core i5-2400 CPU @ 3.10GHz and 4GB DDR3 SDRAM under Mac OS X 10.7.4 Build 11E53. Arguments to AutoDock Vina and idock were left as default. By default, both programs output at most 9 predicted conformations per ligand.

3.5 Results

3.5.1 Rescoring results

Table 3.1 compares 21 scoring functions on the PDBbind v2007 core set ($N = 195$) in terms of Pearson correlation coefficient R_p, Spearman correlation coefficient R_s, and standard deviation SD. The scoring functions are sorted in the descending order of R_p. In terms of R_p and SD, RF-Score, AutoDock Vina and idock rank 1st, 7th and 8th respectively, already outperforming the majority of commercial scoring functions. The statistics for AutoDock Vina and idock are reported in this study and the statistics for the other 19 scoring functions are collected from [10, 55, 148, 149]. RF-Score [10], ID-Score [55], SVR-Score [149] and X-Score [150] are the only scoring functions whose training set do not overlap with the PDBbind v2007 core set.

Figure 3.4 plots the pairwise correlations between experimen-

Table 3.1: 21 scoring functions compared on PDBbind v2007 core set.

Scoring function	R_p	R_s	SD
RF-Score	0.774	0.762	1.59
ID-Score	0.753	0.779	1.63
SVR-Score	0.726	0.739	1.70
X-Score::HMScore	0.644	0.705	1.83
DrugScoreCSD	0.569	0.627	1.96
SYBYL::ChemScore	0.555	0.585	1.98
AutoDock Vina	0.554	0.608	1.98
idock	0.546	0.612	1.99
DS::PLP1	0.545	0.588	2.00
GOLD::ASP	0.534	0.577	2.02
SYBYL::G-Score	0.492	0.536	2.08
DS::LUDI3	0.487	0.478	2.09
DS::LigScore2	0.464	0.507	2.12
GlideScore-XP	0.457	0.435	2.14
DS::PMF	0.445	0.448	2.14
GOLD::ChemScore	0.441	0.452	2.15
SYBYL::D-Score	0.392	0.447	2.19
DS::Jain	0.316	0.346	2.24
GOLD::GoldScore	0.295	0.322	2.29
SYBYL::PMF-Score	0.268	0.273	2.29
SYBYL::F-Score	0.216	0.243	2.35

tal binding affinity and predicted binding affinity by RF-Score, AutoDock Vina and idock on the PDBbind v2012 refined set (N = 2,897). Values are in pK_d or pK_i unit. Since both AutoDock Vina and idock were trained on the PDBbind v2007 refined set (N = 1,300), in order to make a fair comparison, in this benchmark we re-trained RF-Score on the same training set. On one hand, the re-trained RF-Score managed to predict the binding affinity accurately with Pearson correlation coefficient $R_p = 0.765$, Spearman correlation coefficient $R_s = 0.755$, root mean square error $RMSE = 1.26$, and standard deviation $SD = 1.26$. On the other hand, although AutoDock Vina and idock claimed to do well in conformation prediction, they could not predict binding affinity very accurately ($R_p = 0.466$, $R_s = 0.464$, $RMSE = 1.74$, $SD = 1.74$ for AutoDock Vina, and $R_p = 0.451$, $R_s = 0.453$, $RMSE = 1.75$, $SD = 1.75$ for idock), a very common obstacle in the entire computational chemistry community. As expected, the correlation between binding affinity predicted by AutoDock Vina and idock is very close to 1 because of their identical scoring function but different numerical approximation methods [7]. The above observations also apply to the results on the CSAR NRC HiQ Set 24Sept2010 ($N = 343$), as can be seen in Figure 3.5, where $R_p = 0.801$, $R_s = 0.795$, $RMSE = 1.34$, $SD = 1.34$ for RF-Score, $R_p = 0.595$, $R_s = 0.612$, $RMSE = 1.79$, $SD = 1.79$ for Vina, and $R_p = 0.597$, $R_s = 0.613$, $RMSE = 1.79$, $SD = 1.79$ for idock.

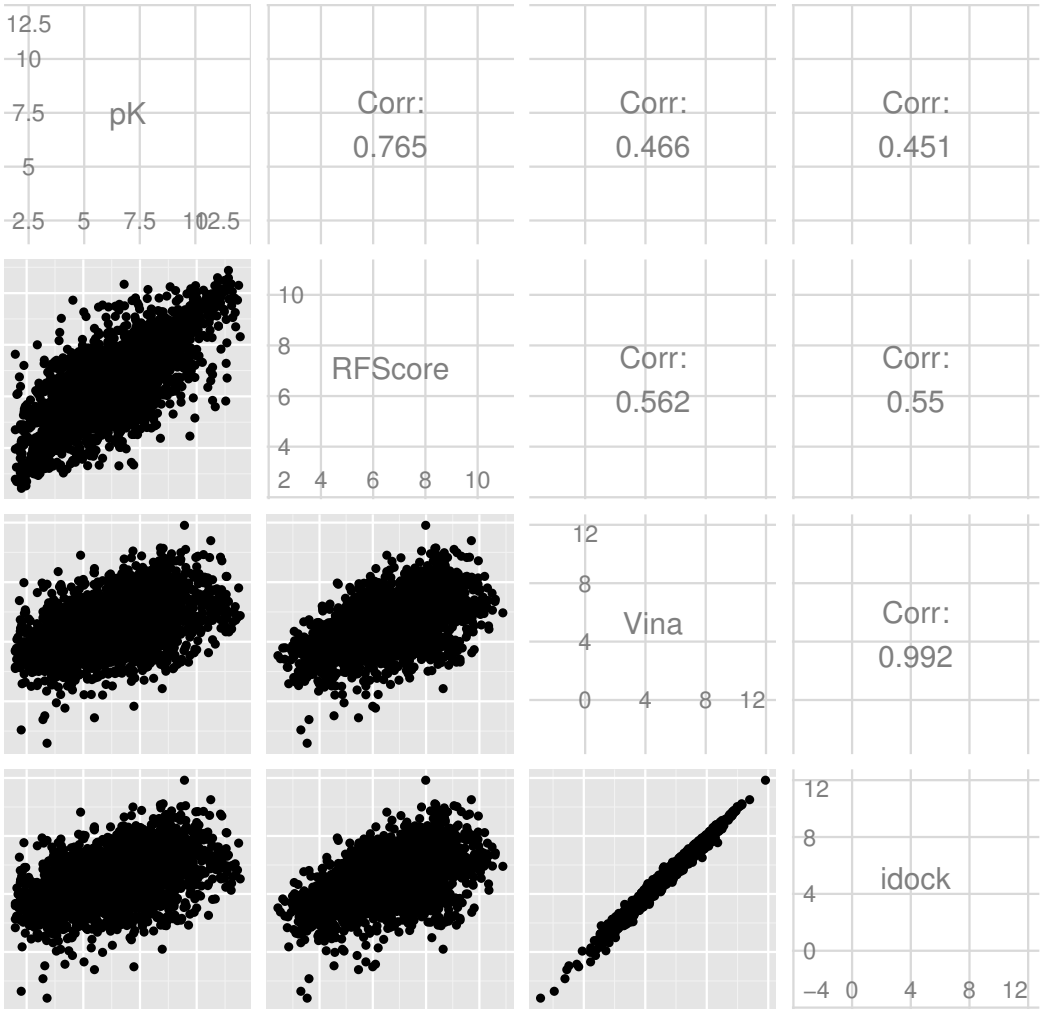


Figure 3.4: Correlations between experimental and predicted binding affinity on PDBbind v2012 refined set.

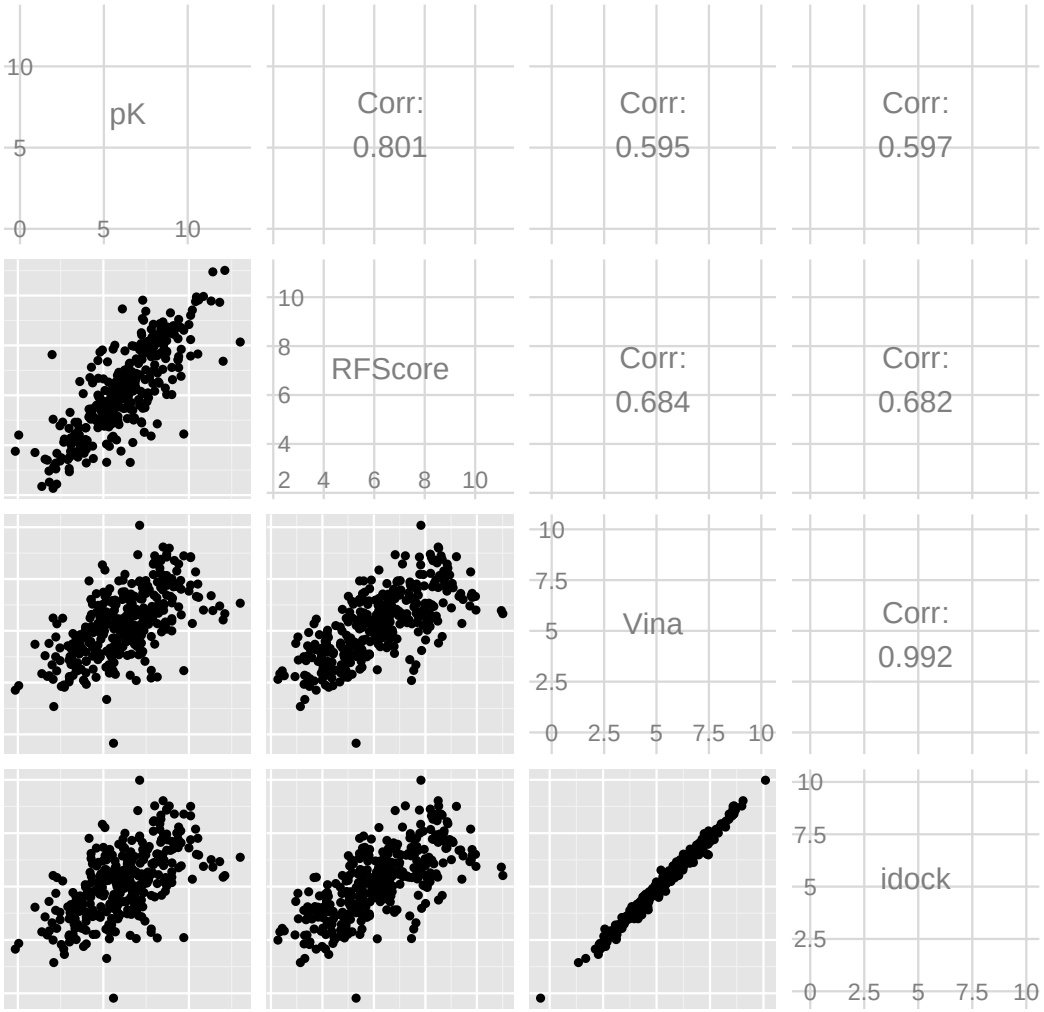
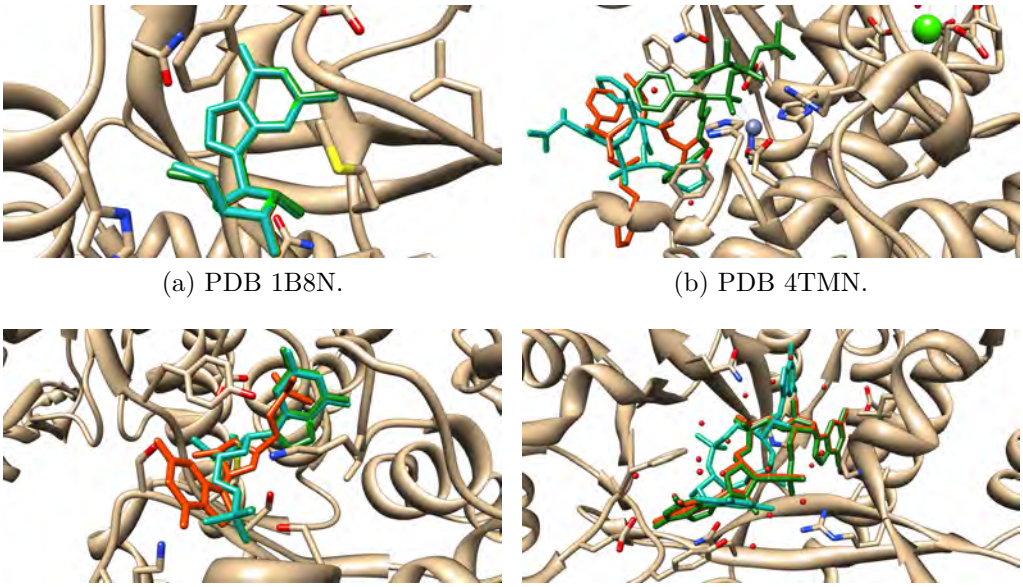


Figure 3.5: Correlations between experimental and predicted binding affinity on CSAR NRC HiQ Set 24Sept2010.

3.5.2 Redocking results

Figure 3.6 visualizes the redocking results of four protein-ligand complexes selected from the PDBbind v2012 refined set. The crystal ligand conformation is rendered in green. The conformation predicted by Vina is rendered in red. The conformation predicted by idock is rendered in blue. In (a), the protein target is purine nucleoside phosphorylase. $RMSD = 0.14$Å for Vina, and $RMSD = 0.13$Å for idock. Both methods managed to predict a conformation sufficiently close to that of the co-crystallized ligand. In (b), the protein target is thermolysin. $RMSD = 8.40$Å for Vina, and $RMSD = 9.91$Å for idock. Both methods failed to predict a conformation sufficiently close to that of the co-crystallized ligand, probably due to the presence of a zinc ion in the binding site. In (c), the protein target is bifunctional purine biosynthesis protein PURH. $RMSD = 7.06$Å for Vina, and $RMSD = 0.21$Å for idock. idock managed to predict a conformation sufficiently close to that of the co-crystallized ligand but AutoDock Vina failed. In (d), the protein target is FimX. $RMSD = 0.29$Å for Vina, and $RMSD = 10.23$Å for idock. AutoDock Vina managed to predict a conformation sufficiently close to that of the co-crystallized ligand but idock failed.

Table 3.2 and Figure 3.7 show the redocking success rates of idock and AutoDock Vina on the PDBbind v2012 refined set ($N = 2{,}897$), the PDBbind v2011 refined set ($N = 2{,}455$), and the CSAR NRC HiQ Set 24Sept2010 ($N = 343$) under various conditions regarding the $RMSD$ values between the crys-



(a) PDB 1B8N. (b) PDB 4TMN.

(c) PDB 1PKX. (d) PDB 3HV8.

Figure 3.6: Redocking visualization of four protein-ligand complexes.

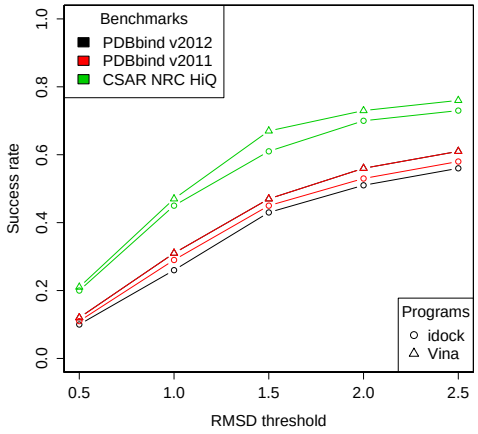
tal and docked conformations. By default, both programs output 9 predicted conformations per ligand. Given a redocking case, $RMSD_i (i = 1, 2, ..., 9)$ refers to the $RMSD$ value between the crystal conformation and the ith docked conformation, i.e. the one with the ith highest predicted binding affinity, whereas $RMSD_{min}$ refers to the $RMSD$ value between the crystal conformation and the closest docked conformation, i.e. the one with the minimum $RMSD$ value. $RMSD_{min} = \min_{i \in [1,9]} RMSD_i$. The condition $RMSD_1 = RMSD_{min}$ therefore tests for how many percent the docked conformation with the highest predicted binding affinity actually turns out to be the closest one among the 9 predicted conformations to the crystal conformation. It can be seen that the success rates of idock are comparable to, albeit slightly lower than, AutoDock Vina, and the

Table 3.2: Redocking success rates of idock and AutoDock Vina.

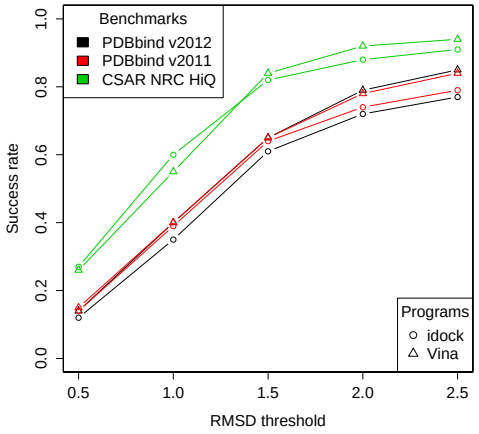
	PDBbind v2012		PDBbind v2011		CSAR NRC HiQ	
Condition	idock	Vina	idock	Vina	idock	Vina
$RMSD_1 = RMSD_{min}$	49%	53%	47%	54%	59%	71%
$RMSD_2 = RMSD_{min}$	15%	16%	16%	14%	18%	13%
$RMSD_3 = RMSD_{min}$	8%	7%	8%	8%	4%	4%
$RMSD_4 = RMSD_{min}$	6%	6%	6%	5%	7%	3%
$RMSD_5 = RMSD_{min}$	5%	4%	5%	5%	3%	1%
$RMSD_6 = RMSD_{min}$	5%	3%	5%	4%	3%	3%
$RMSD_7 = RMSD_{min}$	4%	4%	5%	4%	2%	2%
$RMSD_8 = RMSD_{min}$	5%	3%	4%	3%	3%	2%
$RMSD_9 = RMSD_{min}$	4%	3%	4%	3%	1%	2%
$RMSD_1 < 0.5$ Å	10%	12%	11%	12%	20%	21%
$RMSD_1 < 1.0$ Å	26%	31%	29%	31%	45%	47%
$RMSD_1 < 1.5$ Å	43%	47%	45%	47%	61%	67%
$RMSD_1 < 2.0$ Å	51%	56%	53%	56%	70%	73%
$RMSD_1 < 2.5$ Å	56%	61%	58%	61%	73%	76%
$RMSD_{min} < 0.5$ Å	12%	14%	14%	15%	27%	26%
$RMSD_{min} < 1.0$ Å	35%	40%	39%	40%	60%	55%
$RMSD_{min} < 1.5$ Å	61%	65%	64%	65%	82%	84%
$RMSD_{min} < 2.0$ Å	72%	79%	74%	78%	88%	92%
$RMSD_{min} < 2.5$ Å	77%	85%	79%	84%	91%	94%

success rates on the CSAR NRC HiQ Set 24Sept2010 are consistently higher than the PDBbind v2012 and v2011 refined sets, probably because the scoring function performs well on carefully refined structures. Using a $RMSD$ value of 2.0Å, a publicly accepted positive control for correct bound structure prediction, both programs managed to predict a conformation sufficiently close to that of the co-crystallized ligand as the first conformation in over half of the cases, without any manual tweaking of the protein model.

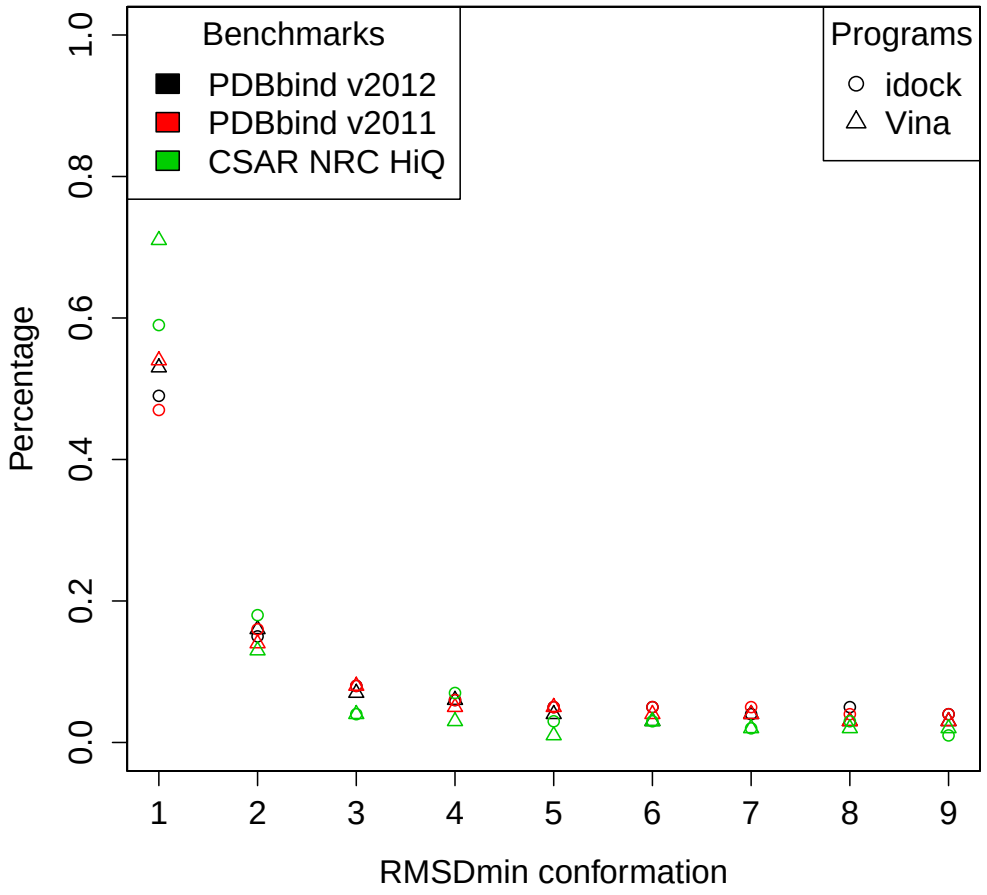
We attempted to examine two possible reasons that might cause idock to fail in some redocking test cases. They are the



(a) $RMSD_1$ success rate.



(b) $RMSD_{min}$ success rate.



(c) $RMSD_{min}$ conformation distribution.

Figure 3.7: Redocking success rates of idock and AutoDock Vina.

number of rotatable bonds of the ligand and the number of metal ions in the binding site. Figure 3.8 plots the impact of rotatable bonds of the ligand on the redocking success rates of idock and AutoDock Vina benchmarked on PDBbind v2012 core set (N = 201). Out of the 201 cases, there are 109 and 114 successful cases for idock and AutoDock Vina respectively. The average number of rotatable bonds of the ligand in successful cases are 7.52 and 7.30 respectively for idock and AutoDock Vina. The average number of rotatable bonds of the ligand in unsuccessful cases are 10.36 and 10.82 respectively for idock and AutoDock Vina. Both programs tended to do well when the ligand contains fewer than 10 rotatable bonds.

Figure 3.9 plots the impact of metal ions in the binding site on the redocking success rates of idock and AutoDock Vina benchmarked on PDBbind v2012 core set ($N = 201$). Out of the 201 cases, there are 158, 31 and 12 cases in which there are 0, 1 and 2 metal ions respectively in the binding site. For idock, the success rates are 0.58, 0.39 and 0.42 when there are 0, 1 and 2 metal ions respectively in the binding site. For AutoDock Vina, they are 0.60, 0.42 and 0.50 respectively. Both programs tended to do well when the binding site contains no metal ions.

Figure 3.10 shows the $RMSD_1$ values for idock plotted against those for AutoDock Vina. The color encodes the number of rotatable bonds (NRB) of the ligand. Many points fall onto the diagonal, suggesting that both programs tended to predict similar conformations.

From the perspective of prospective docking, Figure 3.11 shows

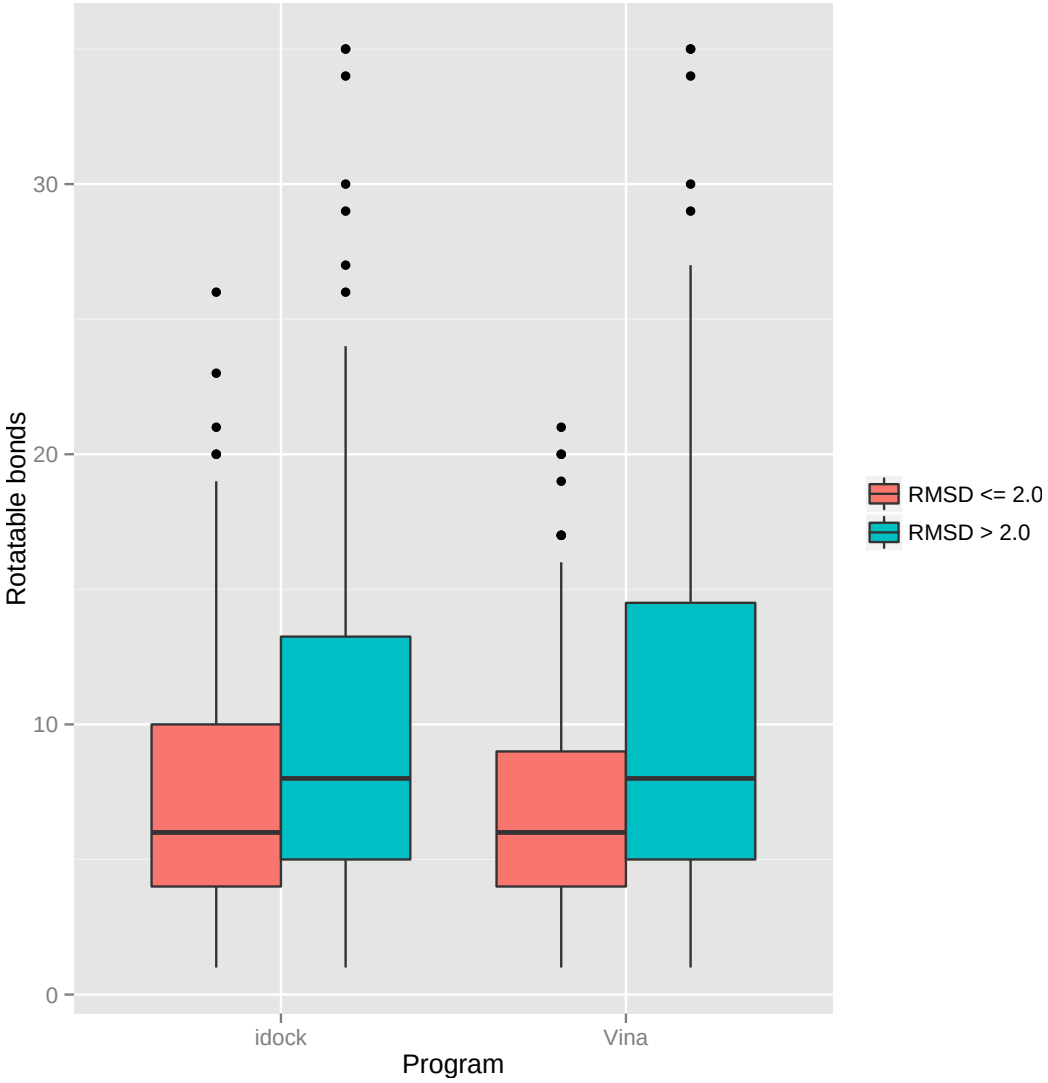


Figure 3.8: Impact of rotatable bonds of the ligand on redocking success rates.

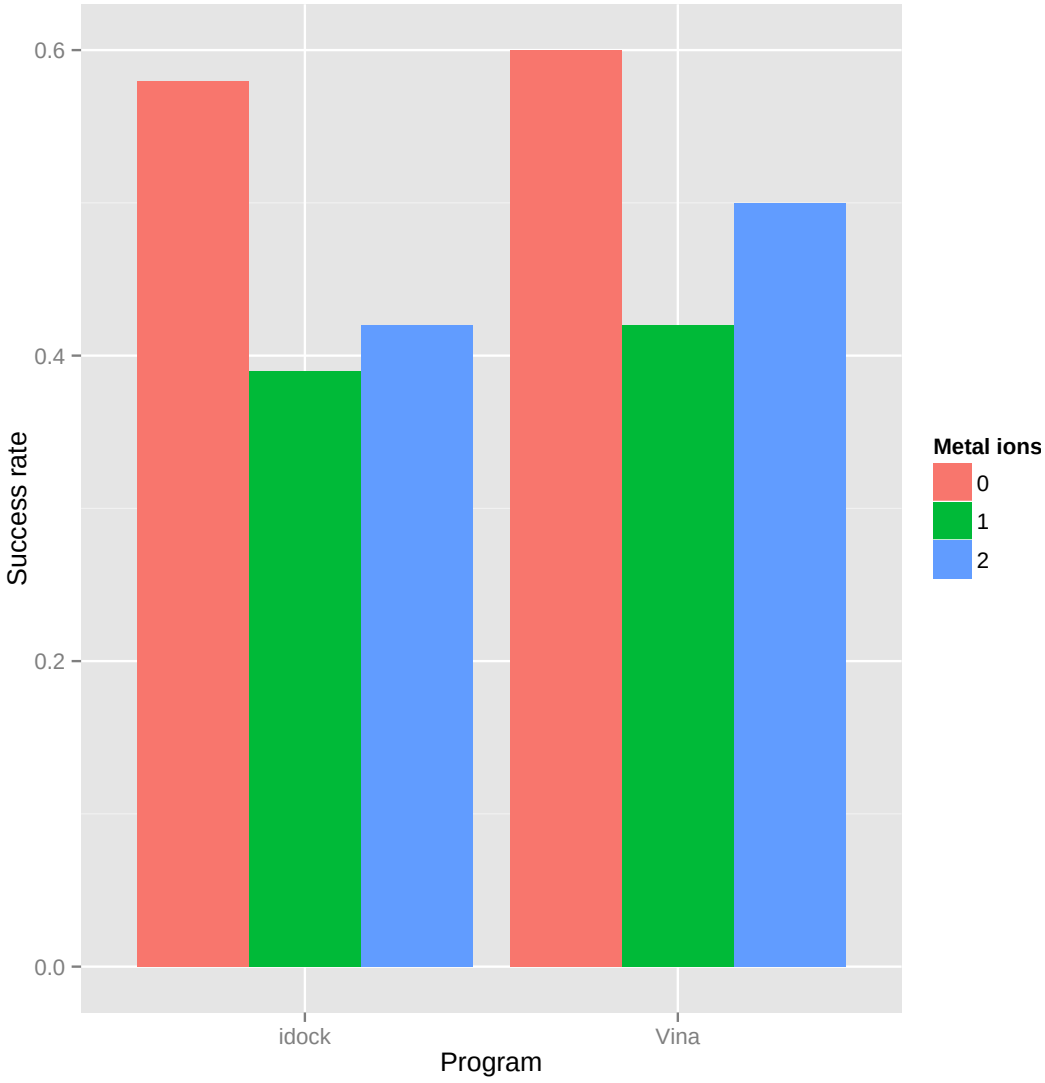


Figure 3.9: Impact of metal ions in the binding site on redocking success rates.

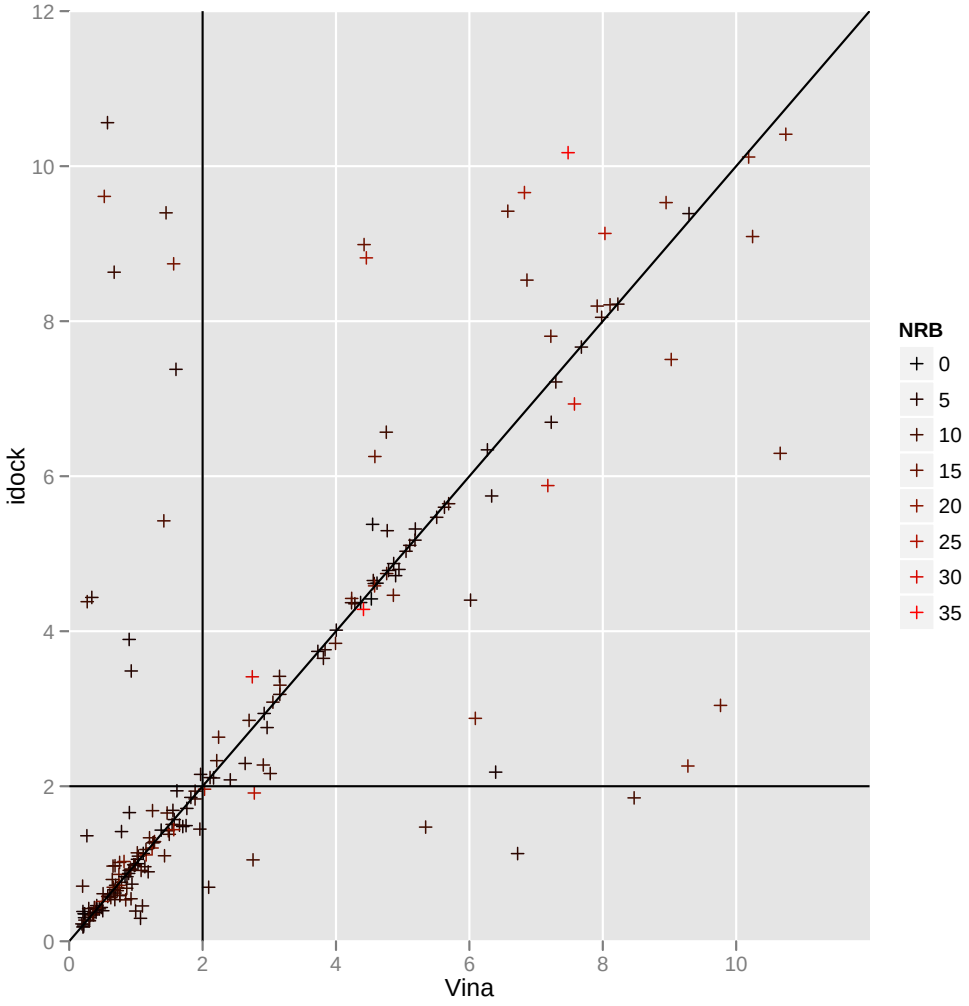


Figure 3.10: $RMSD_1$ of the predicted ligand conformation.

the scatter plot of the highest predicted binding affinity of the 9 docked conformations output by idock against the experimental binding affinity on PDBbind v2012 core set ($N = 201$) in the redocking benchmark. Values are in pK_d or pK_i unit. The weak correlation and large deviation ($R_p = 0.502$, $R_s = 0.530$, $RMSE = 1.31$, $SD = 1.32$) reflect the limitation of using idock alone as scoring function. After re-training RF-Score on PDBbind v2012 refined set ($N = 2{,}897$) and adopting the maximum RF-Score of the 9 docked conformations as predicted binding affinity, the correlation was improved (Figure 3.12, $R_p = 0.815$, $R_s = 0.817$, $RMSE = 0.75$, $SD = 0.76$). Moreover, since for over 50% probability the docked conformation with the highest predicted binding affinity indeed turned out to be the closest to the crystal conformation (i.e. $RMSD_1 = RMSD_{min}$), using RF-Score to re-score the conformation with $RMSD_1$ led to even better prediction (Figure 3.13, $R_p = 0.855$, $R_s = 0.859$, $RMSE = 0.73$, $SD = 0.73$).

3.5.3 Execution time results

Table 3.3 compares the CPU time and elapsed time in hours of docking 3,000 clean ligands in 3 molecular weight sets against 12 diverse receptors by AutoDock Vina and idock. As expected, the execution time varied considerably from protein to protein and from molecular weight set to molecular weight set. Overall, idock outperformed AutoDock Vina by at least 8.69 times and at most 37.51 times, making idock particularly ideal for large-scale

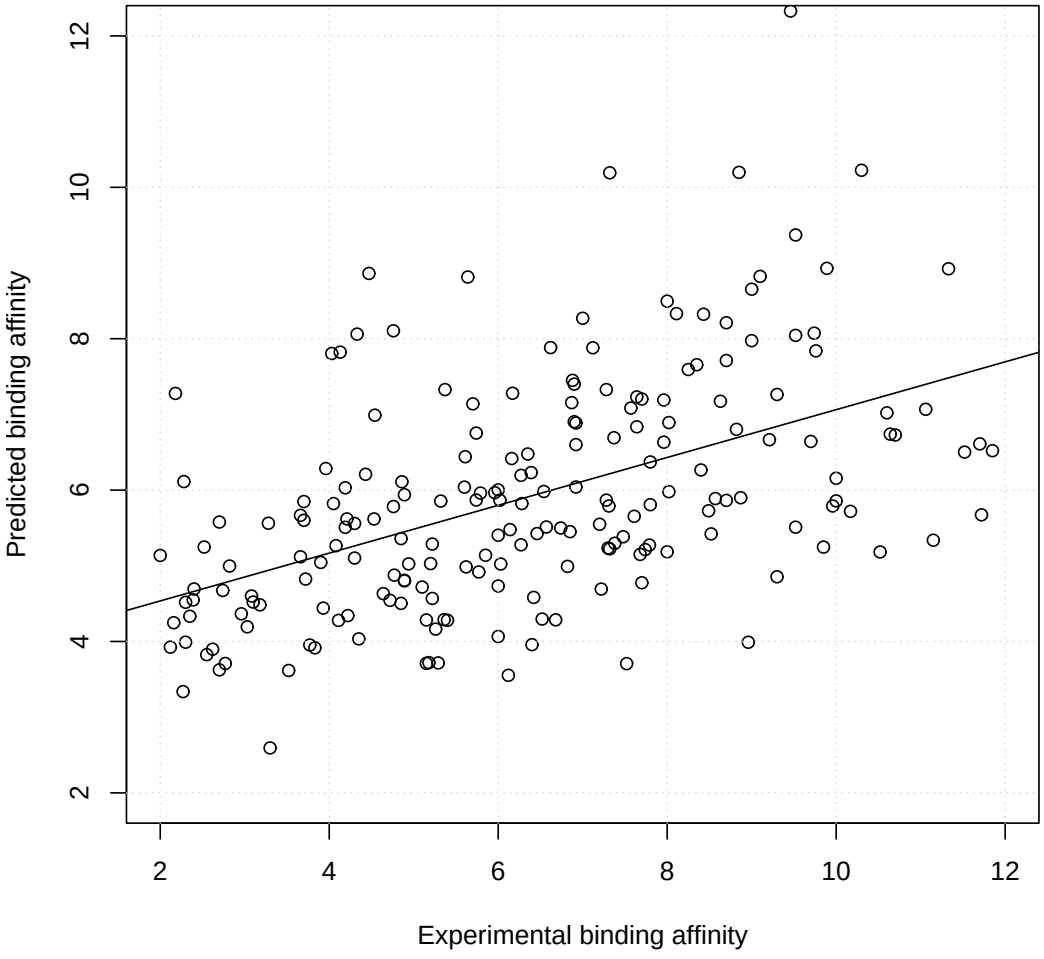


Figure 3.11: Scatter plot of the lowest idock score of the 9 docked conformations against the experimental binding affinity.

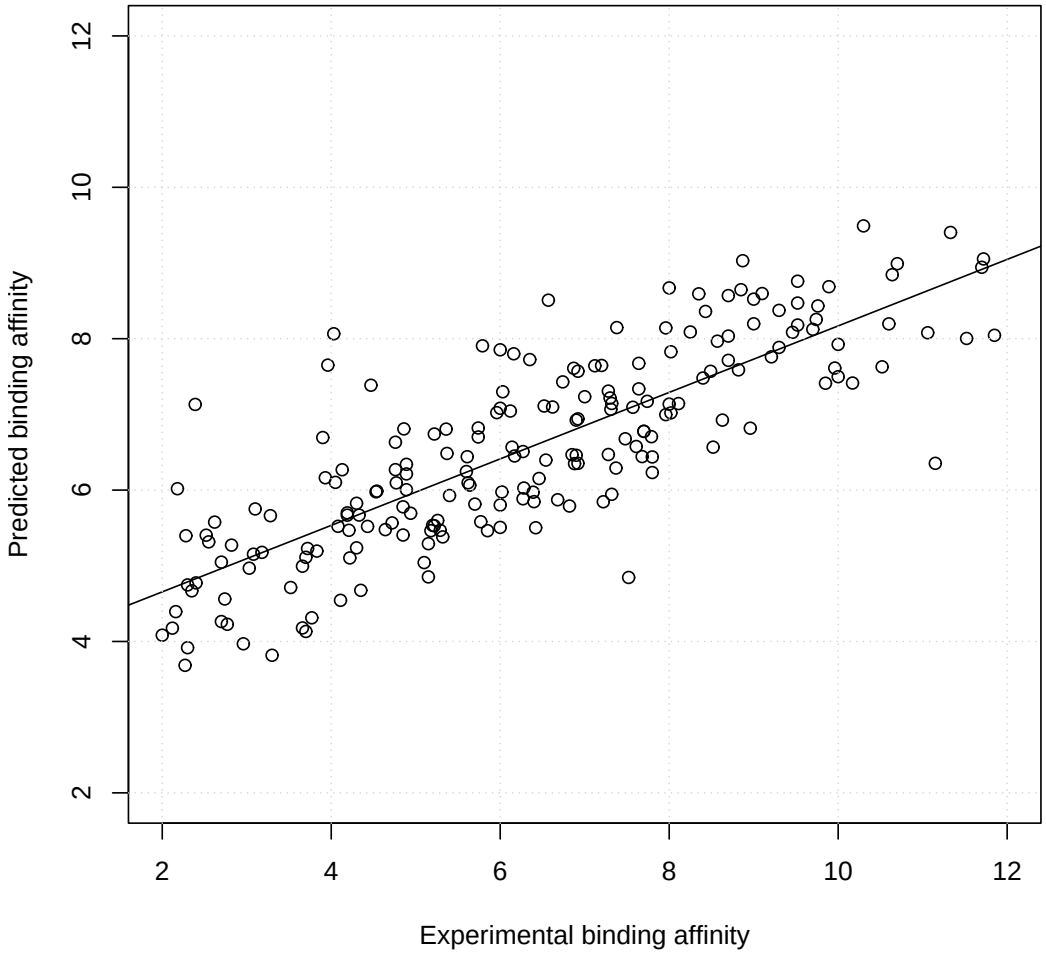


Figure 3.12: Scatter plot of the highest RF-Score of the 9 docked conformations against the experimental binding affinity.

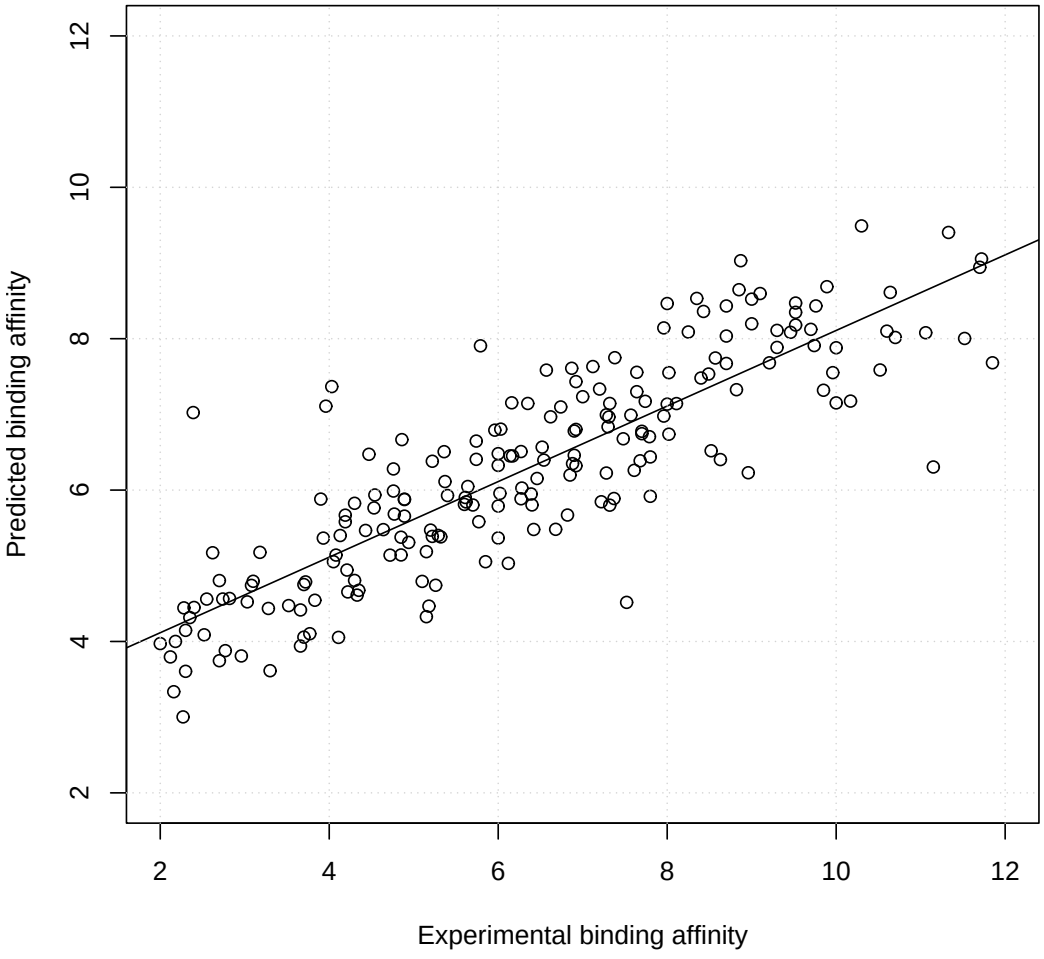


Figure 3.13: Scatter plot of the RF-Score of the first docked conformation against the experimental binding affinity.

docking, as is the case of istar.

3.6 Discussion

Docking is a computational method that investigates how a ligand binds to a protein, and predicts their binding affinity. Hence docking is useful in elaborating inter-molecular interactions and enhancing the potency and selectivity of binding in subsequent phases of the drug discovery process.

In this study, we report a web platform called istar to automate large-scale protein-ligand docking using our popular docking engine idock. Since the initial release of idock, we have been further improving its docking speed and robustness. Compared to AutoDock Vina, our idock features a new numerical model in approximation of the scoring function, replacing slow linear interpolation by fast table lookup. It encapsulates a unique feature that can safely deactivate certain torsions to reduce the dimension of variables. It also implements an efficient thread pool to parallelize multiple components of the program and maintain a high CPU utilization. Results show that idock managed to predict a conformation sufficiently close to that of the co-crystallized ligand as the first conformation in over half of the test cases across a number of diverse datasets, and it outperformed AutoDock Vina by an order of magnitude in terms of docking efficiency at no significant cost of accuracy. It is worthwhile to highlight that in order to use istar, the input protein model requires no manual preprocessing in most cases.

Table 3.3: Execution time of AutoDock Vina and idock.

	200-300g/mol		300-400g/mol		400-500g/mol	
	CPU	Elapsed	CPU	Elapsed	CPU	Elapsed
1HCL human cyclin-dependent kinase 2						
Vina	12.57	3.33	22.55	5.91	51.62	13.41
idock	0.63	0.16	0.92	0.24	1.38	0.36
1J1B human tau protein kinase I						
Vina	9.07	2.47	14.69	3.92	32.28	8.49
idock	0.78	0.21	1.25	0.33	2.35	0.62
1LI4 human S-adenosylhomocysteine hydrolase						
Vina	11.82	3.30	19.08	5.22	39.41	10.64
idock	0.89	0.23	1.55	0.40	3.15	0.82
1V9U human rhinovirus 2 coat protein VP1						
Vina	9.80	2.95	15.55	4.62	29.75	8.49
idock	0.97	0.25	1.64	0.42	3.42	0.89
2IQH influenza A virus nucleoprotein NP						
Vina	9.51	2.66	15.03	4.08	29.64	7.83
idock	0.92	0.24	1.59	0.41	3.41	0.88
2XSK Escherichia coli curli protein CsgC - SeCys						
Vina	10.44	2.71	17.89	4.61	40.58	10.41
idock	0.71	0.19	1.16	0.30	2.16	0.56
2ZD1 HIV-1 reverse transcriptase						
Vina	9.78	2.70	17.67	4.76	42.03	11.33
idock	0.97	0.25	1.52	0.39	2.60	0.69
2ZNL influenza virus RNA polymerase subunit PA						
Vina	9.49	2.60	15.04	4.01	29.97	7.82
idock	0.89	0.23	1.56	0.40	3.41	0.87
3BGS human purine nucleoside phosphorylase						
Vina	9.59	2.57	16.50	4.37	38.42	10.14
idock	0.95	0.25	1.55	0.40	2.81	0.74
3H0W human S-adenosylmethionine decarboxylase						
Vina	9.85	2.64	17.67	4.70	41.69	11.04
idock	0.88	0.23	1.35	0.35	2.20	0.58
3IAR human adenosine deaminase						
Vina	11.25	3.03	20.21	5.39	46.93	12.53
idock	0.80	0.21	1.21	0.32	2.01	0.53
3KFN HIV protease						
Vina	10.53	2.80	18.37	4.83	42.43	11.03
idock	0.77	0.20	1.20	0.32	2.09	0.55
Average across the above 12 receptors						
Vina	10.31	2.81	17.52	4.70	38.73	10.26
idock	0.85	0.22	1.38	0.36	2.58	0.67

We examined two possible reasons that might cause idock to fail in some test cases. They are the number of rotatable bonds of the ligand and the number of metal ions in the binding site. On one hand, a large number of rotatable bonds implies a high dimension of variables to optimize. idock has a higher chance to succeed when the ligand consists of fewer than 10 rotatable bonds. On the other hand, all kinds of metal ions are treated as hydrogen bond donors in the idock scoring function, which might not thoroughly account for their solvation effects and other possible interactions. idock has a higher chance to succeed when the binding site consists of no metal ions.

Although idock performs well in conformation prediction, it displays weakness in binding affinity prediction. In contrast, RF-Score, a new scoring function that circumvents the need for problematic modelling assumptions via non-parametric machine learning, has been recently shown to obtain the best scoring performance among 16 classical scoring functions on PDBbind v2007 core set ($N = 195$) [10]. We have therefore integrated a revised version of RF-Score as an alternative re-scoring function. We have re-trained RF-Score on the entire PDBbind v2012 refined set ($N = 2{,}897$) for prospective prediction purpose. Results show that using RF-Score to re-score the predicted conformations led to a much better prediction with $R_p = 0.855$, $R_s = 0.859$, $RMSE = 0.73$, and $SD = 0.73$. We have successfully demonstrated that RF-Score is a particularly effective re-scoring function for docking purposes.

To compile a more complete list of scoring functions bench-

marked on the PDBbind v2007 core set ($N = 195$) into Table 3.1, we have extracted the performance results for 19 scoring functions from [10, 55, 148, 149], and reported the results for AutoDock Vina and idock on the same test set in this study. This procedure has a number of advantages. Evaluating all the scoring functions on the same test set under the same conditions guarantees a fair and objective comparison. Using a common existing benchmark can also ensure the optimal application of such functions by their authors and avoid the danger of constructing an in-house benchmark on which unrealistically high performance might be produced. Moreover, future scoring functions can be unambiguously incorporated into this comparative assessment. Notably, the top four scoring functions, namely RF-Score [10], ID-Score [55], SVR-Score [149] and X-Score [150], are the only scoring functions whose training set do not overlap with the PDBbind v2007 core set. The prediction power of RF-Score is already superior to many scoring functions employed in commercial docking software. In terms of implementation complexity, a descriptor in RF-Score is just the occurrence count of a particular protein-ligand atom type pair interacting within a certain distance range, while a descriptor in ID-Score can be as mathematically demanding as, for instance, calculating the cosine value of the bond angle between a hydrogen bond donor and a hydrogen bond acceptor. This again demonstrates the adaptiveness of RF-Score to various applications.

One may argue that although the scoring functions are evaluated on the same test set, their training sets are not identi-

cal. Besides, the PDBbind v2007 core set consists of merely 195 complexes, which might not cover sufficient protein-ligand diversity from the perspective nowadays. To address this issue, we re-trained RF-Score on the PDBbind v2007 refined set (N = 1,300), on which AutoDock Vina and idock were also trained, and we expanded the test set to the much larger PDBbind v2012 refined set (N = 2,897). Figure 3.4 clearly shows that all the performance gain ($R_p = 0.765$, $R_s = 0.755$, $RMSE = 1.26$, $SD = 1.26$ for RF-Score versus $R_p = 0.451$, $R_s = 0.453$, $RMSE = 1.75$, $SD = 1.75$ for idock) is guaranteed to come from the scoring function characteristics, ruling out any influence of using different training sets on performance.

In computational biology, ten simple rules have been summarized for providing a scientific web resource [151]. Software and web sites do count for getting ahead as a computational biologist [152]. To design the istar platform in a user-friendly way, we have utilized state-of-the-art web and database technologies. On istar, there are over 23 million ready-to-dock ligands collected from ZINC [27, 28]. These ligands come with supplier information for easy purchasing, and they can be filtered by nine molecular properties in a fine-grained manner. The number of ligands to dock can be previewed in real time. The jobs are transparently split into slices for parallel docking across multiple workstations, and the job progress can be monitored in real time in a browser so that users can have a rough estimation of how long the job will take and when the job will complete. Additionally, our web server supports REST API, by program-

ming against which users can submit multiple jobs in batch. Automation is the major reason of submitting jobs to istar instead of running idock locally on one's computer. With istar at hand, users need not to write special scripts to fetch ligands from some sources, to implement parallelism, or to invoke RF-Score externally by themselves. Users can therefore concentrate on the docking results and subsequent analysis rather than the docking process itself.

We compare our istar to DOCK Blaster [127], an expert system created to investigate the feasibility of full automation of large-scale protein-ligand docking. It uses DOCK 3 [40] as the docking engine and ZINC [27, 28] as the ligand repository. Although DOCK is open source, DOCK Blaster itself is closed source. istar is indeed much easier to use than DOCK Blaster. Given the structure of a target protein, both istar and DOCK Blaster can dock and score a large set of ligands against the target protein and provide a ranked list which users may review and prioritize for purchasing and wet-lab testing. From the perspective of binding site indication, istar automatically detects a site from the co-crystallized ligand, while DOCK Blaster uses PocketPickker (CLIPPERS) [128]. From the perspective of ligand selection, istar features ligand filtering by nine desired molecular properties in a fine-grained fashion, while DOCK Blaster predefines several subsets either by property, by vendor, or by user. From the perspective of documentation and user manual, the istar website presents a series of graphical tutorials on how to submit a new job and other related issues, while DOCK Blaster

deploys a wiki with very rich contents covering all the relevant procedures. As extra features, DOCK Blaster allows the input of known active and inactive binders as heuristic information for docking. In summary, although istar and DOCK Blaster share the identical motivation of automating large-scale protein-ligand docking, their internal implementations and methodologies differ greatly. Users are encouraged to utilize both istar and DOCK Blaster as well as other docking servers to reach a consensus of promising candidate ligands.

According to Google Analytics, throughout 2014, istar had served 460 sessions, 271 users, and 631 pageviews from 33 countries on 6 continents, except Antarctica merely. According to our internal statistics, from October 2013 to October 2014, there had been 635 job submissions with 46,581,105 ligands docked. These jobs came from 126 email addresses, 64 of which were associated with multiple jobs while the other 62 were associated with one single job. In the future, we may ask users to fill in some forms of questionaire in order to systematically collect comments and feedback, as well as their successful drug discovery endeavors using istar.

Due to limited budget, we cannot offer as much hardware resource as DOCK Blaster (i.e. 700 CPU cores plus 20TB RAID-6 storage). However, we emphasize full reproducibility [6] and we have released istar under a permissive open source license so that anyone who possesses sufficient hardware resource is welcome to deploy a copy of istar to his/her own infrastructure with no charge. On a related issue, we can draw on the experience of

QMachine [153], an open-sourced, publicly available web service that acts as a messaging system for posting tasks and retrieving results over HTTP in a PaaS (Platform as a Service) manner. It aggregates commodity hardware and volunteer compute cycles to enable commodity supercomputing in web browsers. Few server resources are required because all analytical and data retrieval tasks are executed by volunteer machines.

3.7 Conclusions

In this study we have reported istar [9], our SaaS platform for bioinformatics and chemoinformatics applications, with idock [7] being a particular instance of large-scale online protein-ligand docking. We believe the huge body of existing AutoDock users can easily transit to the idock service on istar, which we believe constitutes a step toward generalizing the use of docking tools beyond the traditional molecular modeling community.

As a versatile web platform, istar also aggregates our other software and provides them as services. These include a pragmatic implementation of USR (Ultrafast Shape Recognition) [19] and USRCAT (USR with Credo Atom Types) [20] for prospective ligand-based virtual screening, iview [11] for interactive WebGL visualization of protein-ligand complex, igrep [135] for approximate DNA/RNA sequence matching, and icuda as an introductory seminar series about CUDA programming. We encourage our colleagues to host their software as services on istar so that more users can benefit.

3.8 Availability

istar is free and open source under Apache License 2.0. Its source code is available at https://github.com/HongjianLi/istar. Our deployment of istar is running at http://istar.cse.cuhk.edu.hk/.

3.9 Acknowledgements

We thank Professor John J. Irwin for granting us permission to use ZINC [27, 28] with three conditions:

1. We shall provide links to http://zinc.docking.org/substance/zincid for top hits so that users can seek for the most current purchasing information at ZINC's official website.
2. We shall limit the number of top hits for download to 1000 ligands from a single job.
3. We shall update our ligands when ZINC data is updated so that users can benefit from the most current ligand data.

3.10 Future works

idock has been evaluated from the perspectives of rescoring, redocking, and execution time, but it has not been evaluated from the perspective of enrichment, which requires the active ligands be ranked high in a set of decoys. In this case BEDROC [37] and SLR [38] could be used as performance measures for the "early recognition" problem.

□ **End of chapter.**

Chapter 4

iview: molecular visualization

Visualization of protein-ligand complex plays an important role in elaborating protein-ligand interactions and aiding novel drug design. Most existing web visualizers either rely on slow software rendering, or lack the support of macromolecular surface construction. The useful feature of virtual reality is also unavailable.

We have developed iview, an easy-to-use interactive WebGL visualizer for protein-ligand complex. It exploits hardware acceleration rather than software rendering, and supports four surface representations including Van der Waals surface, solvent excluded surface, solvent accessible surface and molecular surface. It features four special effects in virtual reality settings, namely anaglyph, parallax barrier, oculus rift and stereo, leading to visually appealing identification of intermolecular interactions. Moreover, based on the feature-complete version of iview, we have also developed a concise and tailor-made version specifically for our idock@istar to aid online protein-ligand docking

service. This demonstrates the excellent portability of iview.

Using innovative 3D techniques, we provide a user friendly visualizer that is not intended to compete with professional visualizers, but to permit easy accessibility and platform independence. To the best of our knowledge, iview is the only web visualizer that utilizes GPU hardware acceleration and supports three unique features: protein surface construction, virtual reality effects, and PDBQT format input. iview is freely available at http://istar.cse.cuhk.edu.hk/iview.

This was a collaborative project with Takanori Nakane from Graduate School of Medicine, Kyoto University, Japan. It was published in *BMC Bioinformatics* on 25 February 2014 [11]. Notably, this article has been tagged "Highly accessed" by the journal, indicating that it may be of broad interest in the community.

4.1 Background

Protein-ligand visualization serves an important role in elaborating intermolecular interactions and aiding novel drug design. To date, dozens of visualization tools already exist. VMD [31], PyMOL (http://www.pymol.org) and Chimera [30] are quite well-known and highly cited. They can interpret multiple file formats, generate multiple representations, and provide precise control. BINANA [154] characterizes ligand-binding and outputs a state file compatible for display in VMD [31]. AutoDockTools4 [32] provides native support for the PDBQT file format,

which is widely used in various protein-ligand docking software such as AutoDock [32], AutoDock Vina [8], QuickVina [41] and our idock [7]. We also developed our own tool [155] to visualize structures in virtual reality settings and employ fragment-based *de novo* ligand design strategy for use in interactive drug design. PoseView [34] and LigPlot+ [35], on the other hand, plot 2D diagrams of protein-ligand interactions from 3D coordinates.

In addition to the above standalone visualizers, there are web-based visualizers using either Java applet, Adobe Flash, or HTML5 canvas. Jmol (http://www.jmol.org), an open source Java viewer for chemical structures in 3D, has been deployed worldwide and recognized as the *de facto* molecular viewer on the web. GIANT [156], a Jmol derivative, supports analyzing protein-ligand interactions on the basis of patterns of atomic contacts obtained from the statistical analyses of 3D structures. Unfortunately, Java is being disabled on more and more systems due to security concerns. Hence Java-free visualizers are highly desired. JSmol [157], a JavaScript-only version of Jmol, includes the full implementation of the entire set of Jmol functionalities. Although Jmol and JSmol support a large set of advanced features such as scripting, they rely on software rendering which is slow on large display areas and thus prevents detailed inspection of the structure. In contrast, WebGL visualizers benefit from GPU hardware acceleration. For instance, ChemDoodle Web Components (http://web.chemdoodle.com), a pure JavaScript chemical graphics and cheminformatics library, presents 2D and 3D graphics and animations for chem-

ical structures, reactions and spectra, but it lacks protein surface construction. GLmol (http://webglmol.sourceforge.jp), a molecular viewer on WebGL/JavaScript using the three.js library, supports multiple file formats and representations, and features an experimental version of surface construction based on the EDTSurf algorithm [158, 159], an experimental version of atomic labeling and click-to-identifying, and an experimental version of electron density map visualization by isomesh or volume rendering. Another WebGL technology [160] also supports rendering molecular surface using the SpiderGL library [161]. However, none of these WebGL visualizers except JSmol support virtual reality effects.

4.2 Motivation

Surface representation is a convenient way to visualize protein-ligand interactions. Nevertheless, macromolecular surface calculation is computationally and memory intensive. Furthermore, the calculated mesh is fairly complex, often exceeding 500,000 polygons. Therefore its implementation in JavaScript/WebGL has been considered very difficult. Most existing web visualizers either rely on slow software rendering, or lack virtual reality support. Moreover, the important feature of protein surface construction is usually unavailable, and the support for PDBQT format is not implemented. To address the above obstacles, we developed iview, an interactive WebGL visualizer of protein-ligand complex.

4.3 Objective

We aimed at designing iview as a convenient approach to visualize protein-ligand complex directly on the web. In iview, in addition to conventional visualization functionalities, we implemented macromolecular surface representations as well as special effects in virtual reality settings. Furthermore, we designed iview to be flexible enough so that it could be easily modified to adapt to different applications. As an application example, we refactored the feature-rich version of iview and derived a tailor-made version specifically for visualizing idock@istar input data and output results of user-submitted jobs.

4.4 Methods

iview is refactored from GLmol 0.47 and uses three.js as its primary 3D engine with anti-aliasing support. It is based on WebGL canvas and can be easily integrated into existing HTML5 web pages to display molecular models without requiring Java or browser plugins. It loads a protein-ligand structure from the PDB (Protein Data Bank) [162] as its data source via a RESTful interface. It renders four standard representations of primary structure, namely line, stick, ball & stick and sphere, and five standard representations of secondary structure, namely ribbon, strand, cylinder & plate, C alpha trace and B factor tube. It colors the structure by either atom spectrum, protein chain, protein secondary structure, B factor, residue name, residue polarity, or

atom type, by setting the vertex colors of the geometry object of the corresponding representation. It supports user interactions including rotation, translation, zooming and slab with mouse or hand touch manipulation. It provides both perspective and orthographic cameras, and anaglyph, parallax barrier, oculus rift and stereo effects from three.js examples for use in a virtual reality environment. It supports exporting the WebGL canvas to PNG (Portable Network Graphic) which can be subsequently embedded in a document.

We ported EDTSurf [158, 159], a fast algorithm to generating triangulated macromolecular surfaces by Euclidean distance transform, to JavaScript and integrated it into iview to construct and render in real time four representations of protein surface, namely Van der Waals surface, solvent excluded surface, solvent accessible surface and molecular surface, with opacity and wireframe adjustable by users. Note that molecular surface is in fact solvent excluded surface, but EDTSurf uses different ways to derive them. So we provide them both as two different surface representations in iview. Although the JavaScript implementation of the EDTSurf algorithm typically consumes a few seconds and 500MB to 700MB memory for computation, it is sufficiently efficient for practical applications. To limit CPU and memory usage, the calculation grid size is restricted to 180 x 180 x 180.

4.5 Results

Table 4.1 lists the full features of iview.

Table 4.1: iview features.

Category	Features
File format input	PDB
	PDBQT
Camera	perspective
	orthographic
Background	black
	grey
	white
Structure coloring	atom spectrum
	protein chain
	protein secondary structure
	B factor
	residue name
	residue polarity
	atom type
Primary structure	line
	stick
	ball & stick
	sphere
	dot
Secondary structure	ribbon
	strand
	cylinder & plate
	C alpha trace
	B factor tube
Protein surface	Van der Waals surface
	solvent excluded surface
	solvent accessible surface
	molecular surface
Proteins surface opacity	1.0, 0.9, 0.8, 0.7, 0.6, 0.5
Protein surface wireframe	yes, no
Atom and residue labeling	yes, no
Virtual reality effect	anaglyph
	parallax barrier
	oculus rift
	stereo
Canvas manipulation	mouse
	hand touch
Manipulation mode	rotation
	translation
	zooming
	slab
Canvas export	png

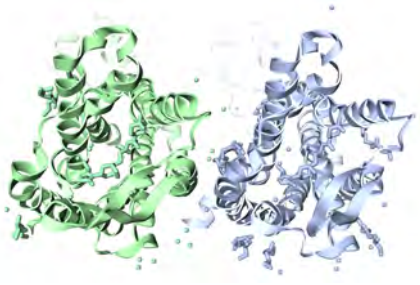
We use as an example CCR5 chemokine receptor-HIV entry inhibitor maraviroc complex (PDB: 4MBS) [163]. Figures in this section are reproducable at http://istar.cse.cuhk.edu.hk/iview/?4MBS.

Figure 4.1 shows the complex in six common coloring schemes, with the protein secondary structure shown in ribbon representation. When the protein is colored by chain, users can clearly see that the protein is a dimer consisting of two polypeptides. When the protein is colored by secondary structure, users can clearly see which segments are alpha helices and which segments are beta sheets. When the protein is colored by B factor, which is also known as temperature facto used to describe the displacement of the atomic positions from an average value, users can clearly see the amino acids in warm colors are flexible.

Figure 4.2 shows the protein secondary structure in four other representations besides ribbon, under the coloring scheme by secondary structure. In the representation of cylinder and plate, alpha helices are rendered as cylinders and beta sheets are rendered as plates. In the representation of C alpha trace, the alpha carbon atoms of consecutive amino acids are connected by lines. In the representation of B factor tube, the B factor value is reflected by the thickness of the tube.

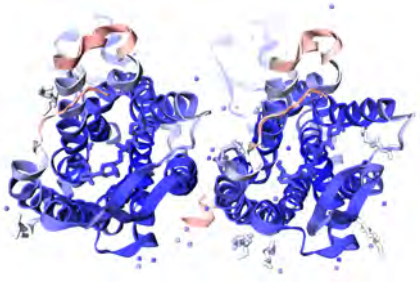
Figure 4.3 shows the protein surface in four common representations colored by atom types, with opacity set to 1.0, i.e. zero transparency. These surfaces were constructed by our JavaScript implementation of the EDTSurf algorithm [158, 159]. Note that molecular surface is equivalently the solvent excluded surface, but EDTSurf uses different ways to derive them. It can

(a) By spectrum.

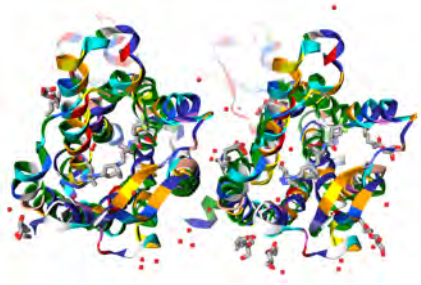


(b) By chain.

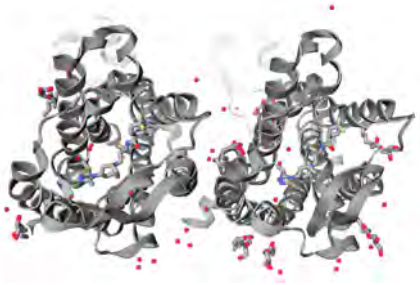
(c) By secondary structure.



(d) By B factor.

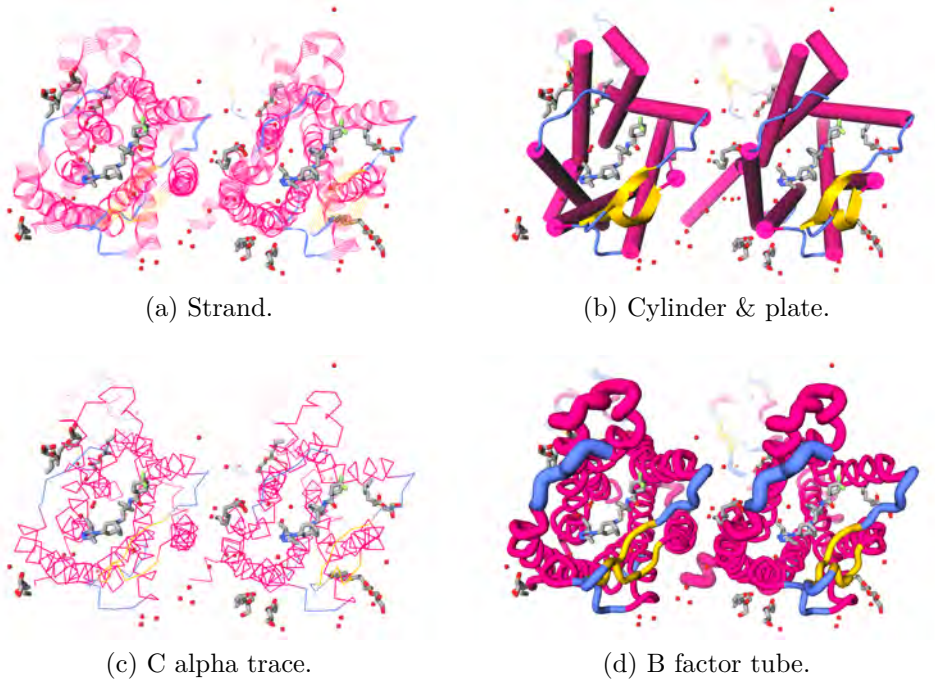


(e) By residue.



(f) By atom.

Figure 4.1: Coloring schemes.



(a) Strand. (b) Cylinder & plate.

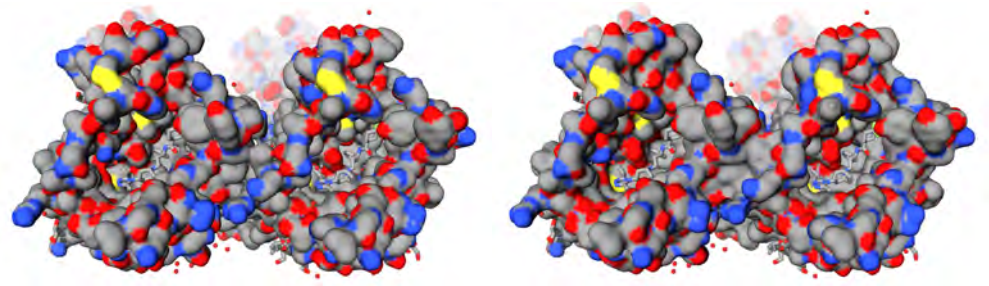
(c) C alpha trace. (d) B factor tube.

Figure 4.2: Secondary structure representations.

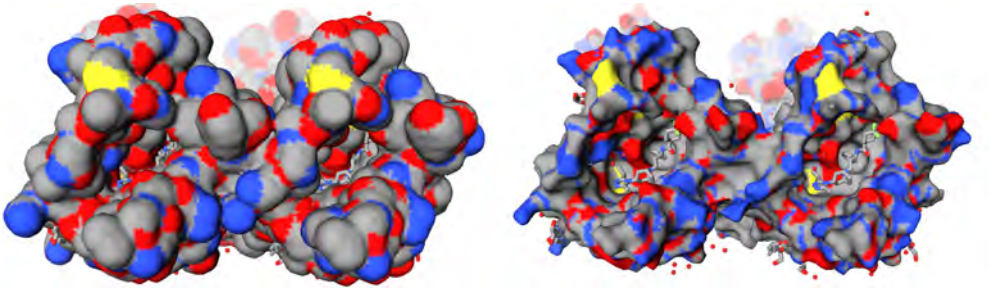
be clearly seen that the asymmetric unit is composed of two complexes, and the protein forms a deep allosteric cavity where the ligand is buried.

Figure 4.4 shows the protein in molecular surface representation in different degrees of opacity, with protein atoms also shown in line representation. When the surface is rendered with transparency to some extent, users can simultaneously inspect the surface and the surrounding amino acids at atomic level.

Figure 4.5 illustrates the four effects in a virtual reality environment. The anaglyph effect encodes each eye's image using filters of chromatically opposite colors to achieve stereoscopic 3D effect. When users wear a spectacle with special filters on both sides, each of the two differently filtered colored images reaches

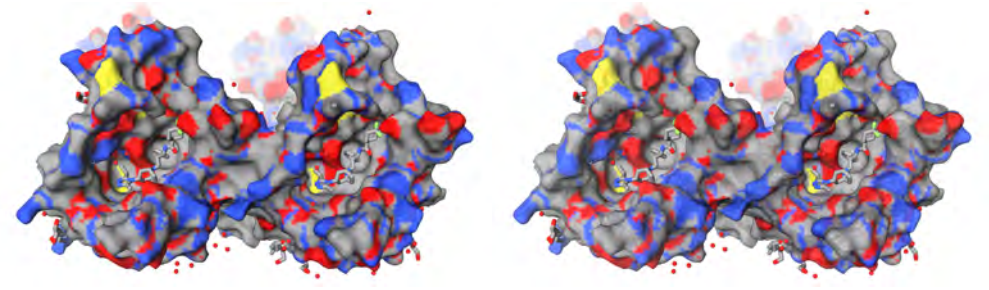


(a) Van der Waals surface. (b) Solvent excluded surface.

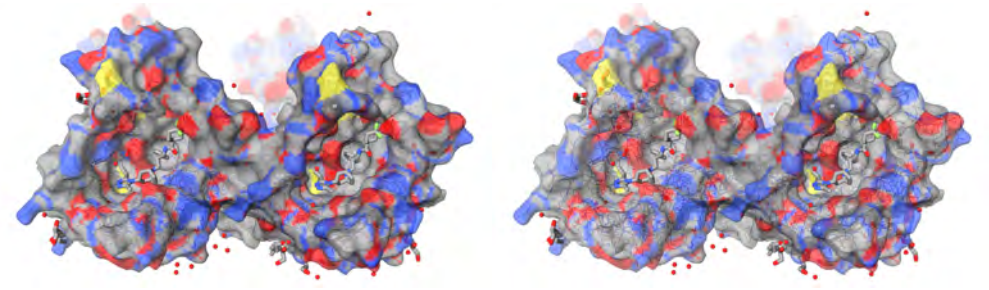


(c) Solvent accessible surface. (d) Molecular surface.

Figure 4.3: Protein surface representations.



(a) 0.9 (b) 0.8



(c) 0.7 (d) 0.6

Figure 4.4: Protein surface opacity.

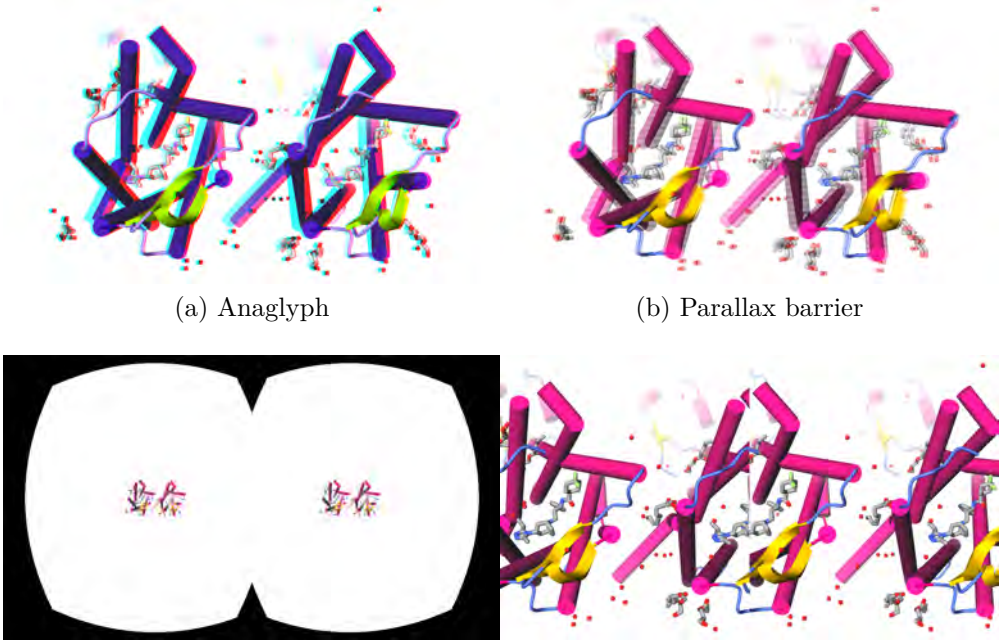
one eye, revealing an integrated stereoscopic image. The disparity between two superimposed molecules creates a perception of depth, leading to visually more appealing identification of intermolecular interactions.

A parallax barrier is a device placed in front of a LCD (Liquid Crystal Display) to permit a stereoscopic or multiscopic image without 3D glasses. The device is composed of a layer of material with precision slits, enabling each eye to see a different set of pixels and thus creating a sense of depth through parallax.

The cculus rift is a virtual reality head-mounted device, which features a high-speed inertial measurement unit and a LCD display, visible via dual lenses positioned over the eyes to provide a 90 degrees horizontal and 110 degrees vertical stereoscopic 3D perspective.

4.6 Application

We emphasize portability and usability, and illustrate that iview can be easily modified to suit one's particular application, given that iview is free and open source under a permissive license. We take protein-ligand docking as an example. Based on the feature-rich version of iview, our tailor-made version specifically for idock@istar cleans up many dispensable functions, enabling a very neat interface. It only retains the rendering of primary structure of protein and ligand, and the construction of protein surface. Most importantly, it implements new features especially for protein-ligand docking purpose. Figure 4.6



(a) Anaglyph (b) Parallax barrier

(c) Oculus Rift (d) Stereo

Figure 4.5: Virtual reality effects.

shows this application-specific version and can be reproduced at http://istar.cse.cuhk.edu.hk/idock/iview/?525a0abab0717fe31a000001.

In the input phase of a docking job, it merely requires a PDB file, which can be obtained either from the PDB database [162] or via homology modeling, and then constructs the protein surface asynchronously in a separate web worker to keep the web page responsive. It automatically detects a binding site from the largest co-crystallized ligand first by finding the smallest cubic box that covers the entire ligand and then by extending the box by 50% in all the three dimensions in order to reserve space for conformational sampling. In case of non-existence of co-crystallized ligand, the binding site is defaulted to the geometric center of the protein. The binding site is visually depicted in

the form of a cubic box whose center and size can be manually adjusted by users in real time.

In the output phase of a docking job, it displays the user-supplied cubic box for users to confirm that the predicted ligand conformations do fall inside the desired binding site. Other than PDB format, its parsers are capable of parsing a protein and multiple top hit ligands in PDBQT format used by idock. It displays the top hit ligand IDs in a horizontally scrollable row and provides a straightforward way to switch ligands easily through a button group. It has built-in support for putative intermolecular hydrogen bond detection by finding hydrogen bond donors and acceptors from protein and ligand and setting the distance threshold to 3.5Å. It automatically annotates important atoms, like those involving in intermolecular hydrogen bonds, by placing labels next to the corresponding atoms in the canvas. It lists the docking result files, predicted free energy and binding affinity values, molecular properties, SMILES representation, compound suppliers and annotations, and putative hydrogen bond positions and their lengths, in order to give users a quick overview of the top hit ligands and assist them in making decisions of which compounds to purchase for subsequent wet-lab experiments.

4.7 Discussion

We developed iview with the purpose to simplify and promote the use of idock@istar. At first we tried ChemDoodle Web Com-

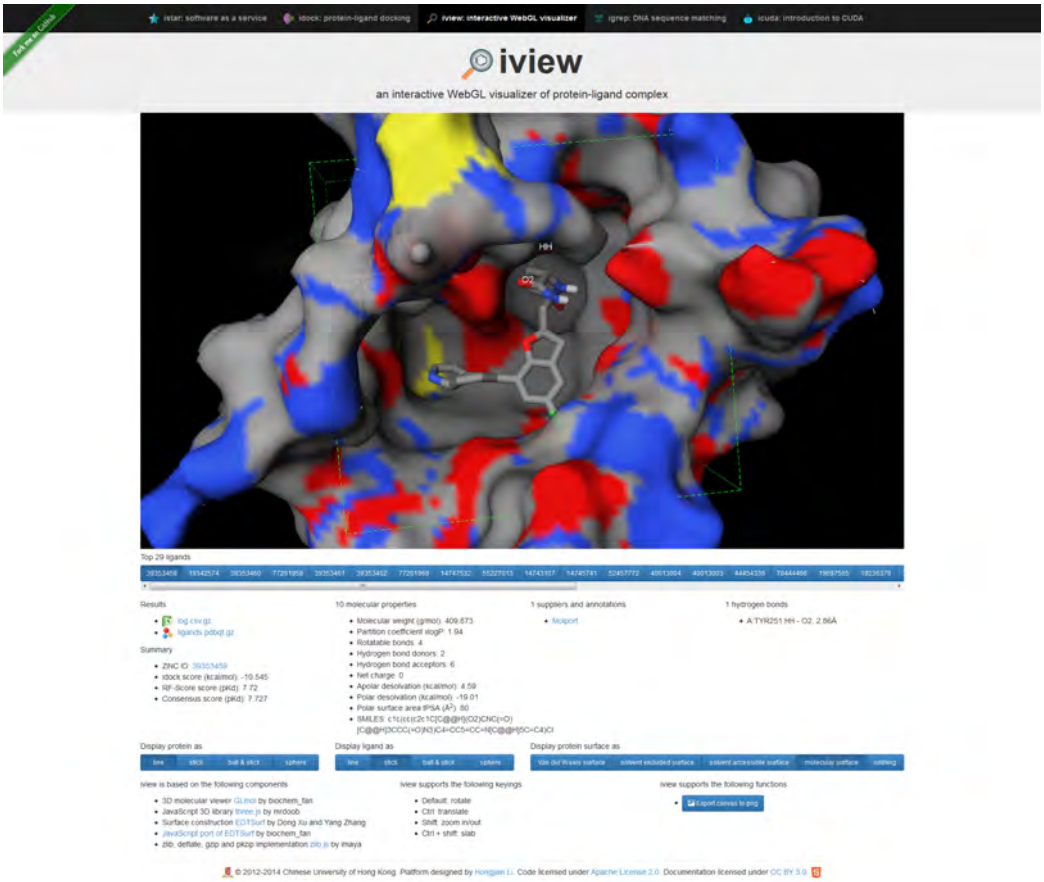


Figure 4.6: Tailor-made version of iview specifically for visualizing idock@istar results of user-submitted jobs.

ponents, but it had strong dependency on its server side components, whose source code we had no access to. Later we turned to GLmol, although which has been discontinued since 29 August 2012, was quite an exciting project because it gracefully built its geometric modeling and relevant functions on top of the three.js foundation, and thus greatly reduced the programming difficulty.

Based on GLmol, iview has fixed many bugs and meanwhile introduced new features. Their differences are as follows. iview supports four virtual reality effects, which GLmol lacks. iview allows users to choose a surface opacity between 1.0 and 0.5 so that users can inspect both protein surface and internal structures at the same time. iview can parse alternate atoms and reconstruct their covalent bonds, but GLmol simply ignores them. iview can identify metal ions and highlight their bonds by dashed lines. GLmol does not distinguish metal ions and displays their bonds by ordinary lines. iview supports as many as 100 atom types, from H to Fm, i.e. the first 100 atoms in the periodic table. GLmol can only recognize about 16 common atoms.

Furthermore, the tailor-made version specifically for protein-ligand docking also supports the following features: parsing of the PDBQT file format, which is widely used in the most cited AutoDock series and our idock software; automatic protein binding site detection using the position and molecular size of the co-crystallized ligand; automatic detection and display of intermolecular hydrogen bonds; automatic annotation of important atoms, such as those involving in intermolecular hydro-

gen bonds; asynchronous generation of protein surface by a web worker, making the web page responsive; result file gzip decompression by the zlib.js library; and fast toggling among different representations via in-memory caching.

It is worthwhile to highlight that iview performs all parsing and rendering in the client browser, with no dependency on the server side at all, thus ensuring the data privacy is maintained. This is unlike ChemDoodle Web Components, some of whose functions send data to a dedicated server for processing and wait for retrieval of results.

4.8 Conclusions

We have designed and developed iview to be a simple and straightforward way to visualize protein-ligand complex. It enables non-experts to quickly elucidate protein-ligand interactions in a 3D manner. Furthermore, iview is free and open source, and can be easily integrated into any bioinformatics application that requires interactive protein-ligand visualization. As far as we are aware, iview is the unique web visualizer that simultaneously utilizes GPU hardware acceleration and supports three pragmatic features: macromolecular surface construction, virtual reality effects, and PDBQT format parsing.

4.9 Availability

iview is free and open source under Apache License 2.0. It is written in JavaScript, HTML5 and CSS3, and available at http://istar.cse.cuhk.edu.hk/iview. It is independent of operating systems but requires a browser and a graphics card with WebGL capability. It has been successfully tested in Chrome 30, Firefox 25, Safari 6.1 and Opera 17. Support for IE 11 is experimental because gl_FrontFacing is unsupported in IE 11. Refer to http://caniuse.com/webgl for compatibility of WebGL support in desktop and mobile browsers.

4.10 Future works

Both BINANA [154] and GIANT [156] can characterize protein-ligand interaction patterns. BINANA, for instance, identifies key binding characteristics like hydrophobic contacts, hydrogen bonds, salt bridges, and pi interactions. Integrating these algorithms will make iview even more pragmatic.

□ **End of chapter.**

Chapter 5

iSyn: fragment-based drug design

Generating *de novo* ligands from molecular fragments can eliminate the diversity limit of compound repositories and lead to the discovery of novel drugs. State-of-the-art fragment-based drug design (FBDD) tools tend to produce oversized compounds with only moderate potency. Worse, these tools require a long execution time in days.

We present iSyn, a WebGL-based tool for interactive FBDD. It features an evolutionary algorithm that automatically designs novel ligands with drug-like properties and synthetic feasibility using click chemistry. iSyn interfaces with our popular and fast molecular docking engine idock, substantially reducing the evaluation and ranking time of drug candidates. Inspired by our user friendly and high-performance WebGL visualizer iview, our iSyn also implements a tailor-made interactive visualizer to aid novel drug design. To illustrate the utility of iSyn in generating novel ligands *ex nihilo*, we designed predicted inhibitors of two

important drug targets, which are RNA editing ligase 1 (REL1) from Trypanosoma brucei, the etiological agent of African sleeping sickness, and cyclin-dependent kinase 2 (CDK2), a positive regulator of eukaryotic cell cycle progression. Results show that iSyn managed to significantly enhance the predicted binding affinity of the best generated ligand by more than 3 orders of magnitude in potency.

iSyn can effectively generate promising compounds with desired potency and molecular mass, and hopefully supplement the efforts of medicinal chemists. iSyn is freely available at http://istar.cse.cuhk.edu.hk/iSyn.tgz.

This was a collaborative project with Chun Ho Chan and Hei Lun Cheung from Department of Computer Science and Engineering, Chinese University of Hong Kong. It was published in *Proceedings of the 2014 Conference Companion on Genetic and Evolutionary Computation Companion (GECCO)* on 12 July 2014 [12] and in *Proceedings of the 18th International Conference on Information Visualisation (IV)* on 15 July 2014 [13].

5.1 Background

Given a pharmacological protein of therapeutic interest, protein-ligand docking tries to discover promising ligands out of existing compound databases. The diversity of its outcome is apparently limited by the diversity of the database. In other words, docking is likely to fail if the selected database contains no promising ligands for that particular protein. Hence constructing *de novo*

ligands from molecular fragments has now become a hot research topic.

FBDD, though displaying a high chance of discovering novel drugs, is indeed rather challenging because the number of chemically feasible, drug-like molecules has been estimated to be in the order of 10^{60} to 10^{100} [164], from which the most promising candidates have to be selected and synthesized. Hence, rather than systematic construction and evaluation of each individual compound, *in silico* FBDD methods rely on the principle of local optimization, which guarantees fast convergence but does not necessarily lead to globally optimal solutions. As a result, most FBDD algorithms are non-deterministic, and feature some extent of stochastic structural optimization.

Recent years have seen a prosperity of FBDD tools, such as LEA3D [130], MORPH [165], GARLig [166], LigBuilder 2 [167], AutoT&T [168], LiGen [48, 169], AutoGrow [170, 171], AutoClickChem [172], CrystalDock [173], LigMerge [174], and the works by Foscato et al. [175, 176], by Shang et al. [177], and by Kawai et al. [178]. More can be found in recent reviews [179, 180]. Meanwhile, FBDD databases have been established, such as e-Drug3D [181]. Retrospectively, a number of compounds that evolved from fragments have entered the clinic, and the approach is increasingly accepted as an additional route to identifying new ligands in inhibitor design [182–186]. Notably, FBDD applications have contributed to the development of a number of FDA-approved drugs [182], demonstrating their usefulness in real life.

In 2009, the AutoGrow algorithm [170] was developed to aid the identification and optimization of predicted ligands in an automatic manner. Although it lacks the insight and intuition which medicinal chemists have, its high degree of automation requires no user interactions beyond the initial setup of fragment libraries and docking parameters. AutoGrow utilizes a genetic algorithm in conjunction with the AutoDock series software [8, 32] to add or exchange moieties of known inhibitors so as to improve their predicted binding affinities. AutoGrow 1.0 and 2.0 use AutoDock4 [32] and AutoDock Vina [8], respectively, for protein-ligand docking and scoring.

In 2011, we were motivated by the desire to design drug candidates in an interactive way under a virtual reality setting, and thus developed a GUI application [155] that enables certain human interactions by translating and rotating the generated ligand amid the evolutionary process in a semi-automatic manner. Particularly, in a virtual reality environment when users wear a spectacle with special filters on both sides, the disparity between two superimposed molecules creates a perception of depth, leading to visually more appealing identification of intermolecular interactions.

In 2013, AutoGrow 3.0 [171] was developed to mainly tackle the synthetic feasibility problem since AutoGrow 1.0 and 2.0 often produce compounds that are neither drug-like nor easily synthesizable. To guide ligand optimization, AutoGrow 3.0 uses the rules of click chemistry, which describes chemistry tailored to generate substances quickly and reliably by joining small units

together. A click chemistry reaction would typically be modular and give high chemical yields, and its process would preferably have simple reaction conditions and use readily available starting materials and reagents. To achieve this goal, AutoGrow 3.0 interfaces with AutoClickChem [172] and LigMerge [174], with the former performing virtual modification and joining reactions and the latter performing crossover reactions. AutoClickChem [172] is capable of performing *in silico* click chemistry reactions, ensuring that chemical synthesis is fast, cheap, and comparatively easy for subsequent testing in biochemical assays. LigMerge [174] is an automated, ligand-based algorithm for systematically swapping the chemical moieties of known ligands to generate novel ligands with potentially improved potency. It has been shown to identify compounds predicted to inhibit peroxisome proliferator-activated receptor gamma, HIV reverse transcriptase, and dihydrofolate reductase with affinities higher than those of known ligands.

5.2 Motivation

Although AutoGrow 3.0 represents the state of the art of FBDD, there are some weak points. AutoGrow 3.0 supports few cutting reactions, which could probably lead to the synthesis of compounds that are too large to absorb by human body. Despite Lipinski's Rule of Fives [83] implemented therein to maintain drug-like properties, generated ligands that violate the rules are simply discarded without further decomposition into small moi-

eties. Moreover, it produces a bunch of new ligands in each generation, and relies on AutoDock Vina [8] to evaluate their predicted binding affinity. The docking efficiency starts to become unacceptable when too many ligands are sampled, prohibiting any more exhaustive approaches. Furthermore, the execution of AutoGrow 3.0 depends on a large set of third-party software including MGLTools [32], Open Babel [187], AutoDock Vina [8] and NumPy/SciPy. This highly coupled dependency and the lack of an easy-to-use user interface hinder its pragmatic applications for novices.

5.3 Objective

We present iSyn, our WebGL-based solution for computationally synthesizing *de novo* drug compounds with click chemistry support plus additional cutting reactions. iSyn interfaces with our popular and fast protein-ligand docking engine idock [7], greatly reducing the time required for computational evaluations of generated ligands by an order of magnitude [9] as compared to AutoDock Vina [8], and thus permitting large-scale executions and exhaustive searching. Most importantly, based on our hardware accelerated WebGL visualizer iview [11] for protein-ligand complex, our iSyn also features a specific variant of iview in order to compose a user friendly interface as well as to inspect intermolecular interactions and aid novel drug design. As for other enhancements, iSyn utilizes the ultrafast shape recognition (USR) algorithm [19] to deduplicate ligands,

and RF-Score-v3 [16, 17] to predict accurate binding affinity of generated ligands.

5.4 Methods

Figure 5.1 shows the overall user interface of iSyn. The UI is refactored from Twitter's sleek, intuitive and powerful HTML5 template Bootstrap, and hosted by the lightweight event-driven and non-blocking I/O model node.js. The UI comprises, from top to bottom, a logo, a protein input field, a WebGL canvas, some parameter input fields, a summary panel, some usage instructions, and an export button.

In a typical workflow, the user loads a protein target of pharmaceutical interest. The WebGL canvas then automatically renders in real time the protein structure in the representation of lines. Meanwhile, the protein atom coordinates and types are sent to a separate web worker to generate *ad hoc* molecular surface using the EDTSurf algorithm [158, 159] in background in order to keep the web UI responsive in all time. The protein surface, having been constructed, is automatically applied on top of the line representation. Hence, the user can clearly see the cavities, most likely ligand binding sites, on the protein surface.

The user can then supply the necessary parameters to run iSyn, such as the center and size of the search space, the number of generations of the evolutionary algorithm, a boolean value indicating whether to use the fragment library that accompanies AutoGrow 3.0, and a brief description of the job. The search

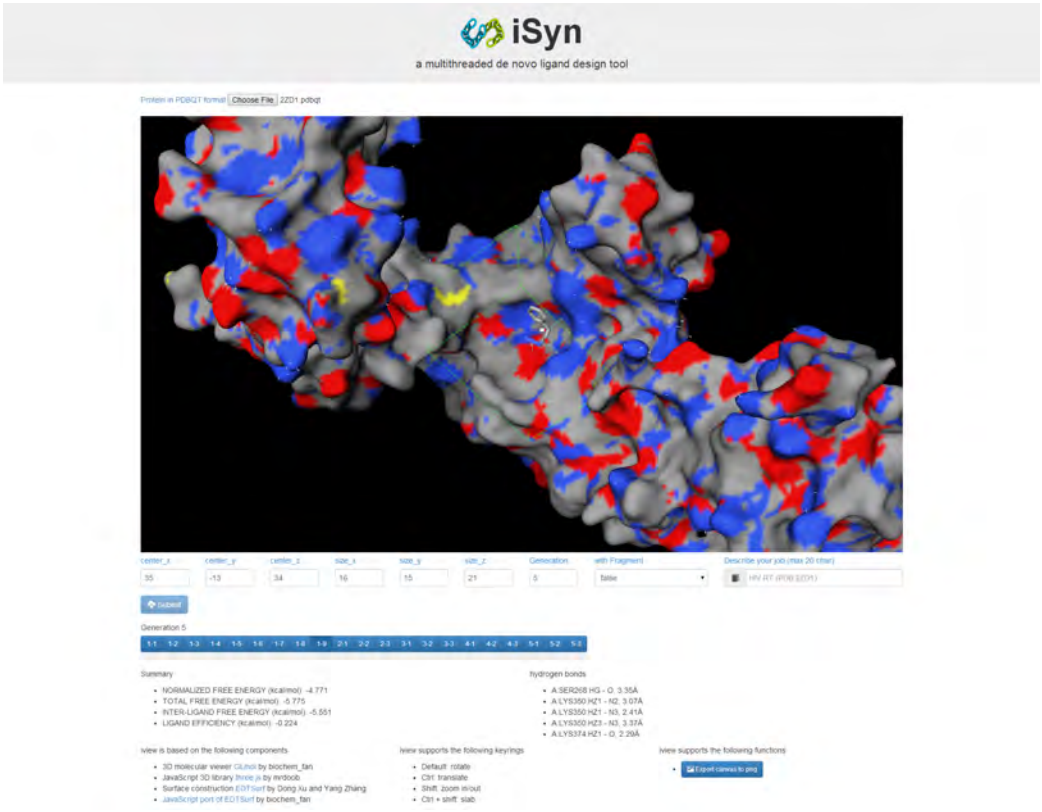


Figure 5.1: iSyn user interface.

space is rendered as a dashed cubic box in green. When the user adjusts the values in the input fields, the box is automatically updates accordingly in real time to reflect the actual position. For those optional parameters, we have set up default values so that even novices can get started easily.

Upon clicking the submit button by the user, both the protein structure and the parameters are passed to the backend iSyn executable, immediately commencing the computational synthesis of novel ligands from some initial ligands and the fragments from the fragment library in an exhaustive manner in order to explore as much structural diversity as possible. The fragment library contains acid anhydride, acyl halide, alcohol, thiol, alkene, alkyne, amine, azide, carbonochloridate, carboxylate, epoxide, ester, halide, isocyanate, isothiocyanate, sulfonylazide, and thio acid moieties, and is collected by performing substructure search of the compounds in the ZINC database [27, 28].

Whenever a population of ligands are generated according to click chemistry reactions, docked against the protein within the binding site, and written to files, the best ligand with the lowest predicted free energy is automatically fetched by the UI in an AJAX fashion using jQuery, a fast, small, yet feature-rich JavaScript library, and visualized in the representation of sticks. Intermolecular putative hydrogen bonds are detected using a cutoff of 3.5Å and rendered as cyan dotted lines.

In the panel beneath the submit button, a number of buttons are dynamically created. Each button represents one conformation of the best ligands in the current generation. In addition,

the ligand properties and docking statistics are shown in the summary panel, which is also dynamically updated when a new generation is completed or when the user switches among different ligands by clicking the corresponding button. Having all the relevant information inside one web page, the user can better examine the results in a neat and intact way.

The UI also supports exporting the canvas view to a production-quality image in PNG format via a button. In this way the user can easily capture the canvas without any auxiliary third-party tools.

5.4.1 Evolutionary algorithm

In the first generation, multiple types of click chemistry reactions and structural modifications are applied to the initial ligands selected from the fragment library to synthesize new ligands, where possible duplicates are detected and removed using the USR (Ultrafast Shape Recognition) algorithm [19]. The generated ligands are then all fed to our popular and fast docking engine idock [7] to predict their preferred conformations as bound to the protein target, and to prioritize them in the ascending order of their predicted free energy, which reflects the binding affinity. The lower the free energy, the higher the binding affinity. Their conversion factor is derived in [9]. An extensive benchmark on 12 diverse proteins [9] has shown that our idock [7] runs 8.69 to 37.51 times faster than the state-of-the-art AutoDock Vina [8] docking software. The utilization of idock by

iSyn promotes the feasibility of large-scale *de novo* drug design and evaluation *in silico*.

The free energy values predicted and sorted by idock are piped to a spreadsheet file in CSV format, which is subsequently parsed by iSyn to retrieve those best ligands with the lowest predicted free energy, i.e. those ligands with the highest predicted binding affinity. To increase the binding affinity prediction accuracy, iSyn provides an alternative option to rescore the docked conformations using our RF-Score-v3 [16, 17]. Then the best ligands in the current generation directly survive into the next generation, and constitute part of the founding ligands. Another part comes from a random selection of the remaining ligands, with the selection probability being proportional to the fitness of a ligand, i.e. its predicted free energy in the case of docking. Our hybrid method, which is essentially a smart combination of the greedy algorithm and the fitness proportionate algorithm, realizes elitism on one hand, while circumvents the risk of over fitting on the other hand.

The evolutionary algorithm either iterates for a number of generations specified by the user, or gets terminated in case of no significant improvement over previous generations.

5.4.2 Click chemistry reaction rules

In iSyn there are four types of operators: addition, mutation, crossover and cutting. They all conform to the requirements of click chemistry. Therefore, the ligands output from iSyn are

guaranteed to be chemically synthesizable, making iSyn really pragmatic for medicinal chemistry and computer-aided drug discovery.

The crossover operators are invoked before the addition and mutation operators, while the cutting operators are called at last in order to prevent the generated ligands from becoming oversized.

Crossover reactions

iSyn uses the LigMerge algorithm [174] to perform crossover reactions. Crossover is done by finding the largest common substructure of two parent ligands and matching the different parts of their fragments attached to that common substructure at each common atom, thereby generating multiple child compounds.

Addition and mutation reactions

iSyn uses the AutoClickChem algorithm [172] to perform addition and mutation reactions. The algorithm consists of 30 click chemistry reactions for different functional groups in the reactants, e.g. azide-alkyne and 1,3-dipolar cycloaddition. In order to enumerate the feasibility of performing a particular reaction, ligands are first classified according to their functional groups such as amine, alcohol, azide, etc.

Cutting reactions

Through the addition and mutation operators, ligands could possibly "grow" too large in terms of molecular size, and might

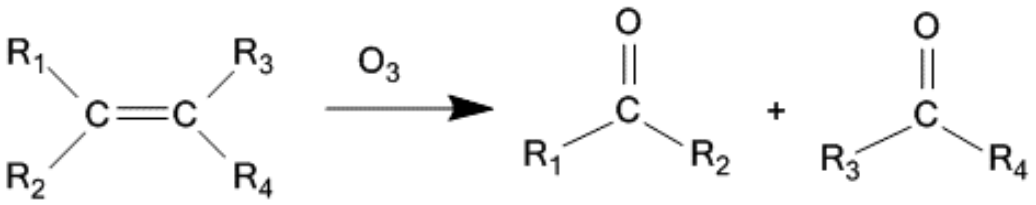


Figure 5.2: Ozonolysis of alkene.

therefore lose drug-like properties. For example, if a generated ligand has a molecular mass of over 500 Daltons, it is unlikely to be absorbed inside the human body and thus unlikely to be optimized into a potential drug. In light of this issue, we have implemented four novel cutting operators to break down oversized ligands.

Ozonolysis is the cleavage of an alkene with ozone to form organic compounds where the carbon-carbon double bond is replaced by a double bond to oxygen (Figure 5.2). The alkene functional group is oxidized with ozone to form aldehydes or ketones, depending on the structure of the ligand under a certain chemical environment. Through this reaction, a ligand containing a carbon-carbon double bond can be broken down into two child ligands.

Vigorous oxidation on alkene can form carbolic acid (Figure 5.3). While oxidation of alkene gives out aldehydes or ketone, further oxidation gives out carboxylic acid and alcohol as products. Aldehyde can be easily oxidized by all sorts of oxidizing agents. As for ketone, although it has certain resistance to oxidation, it can also be oxidized to carboxylic acid by using strong oxidizing agents such as potassium manganite VII solu-

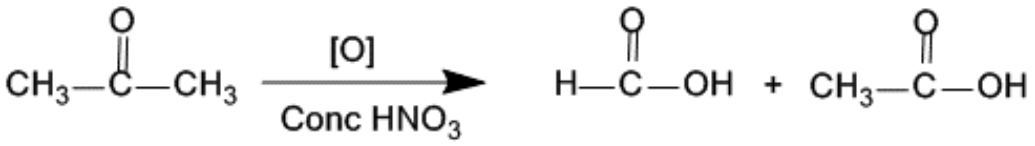


Figure 5.3: Oxidation of alkene to carboxylic acid.

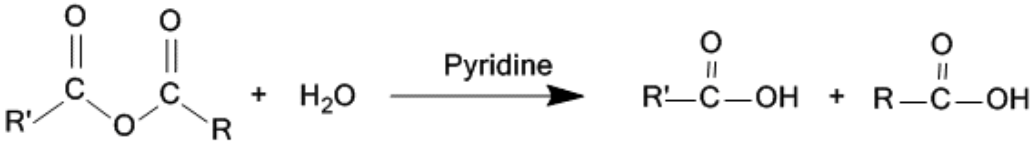


Figure 5.4: Acid anhydride to carboxylic acid.

tion. Through this reaction, a ligand containing carbon-carbon double bonds can be broken down into four child ligands.

Acid anhydrides can react with water to form carboxylic acid (Figure 5.4). An acid anhydride has two acyl groups bound to the same oxygen atom. The two acyl groups are derived from the same carboxylic acid. In reverse, the acid anhydride can be broken down into two original carboxylic acids by reacting with water. Through this reaction, a ligand containing two acyl groups bound to the same oxygen atom can be broken down into two child ligands.

Hydrolysis of ester is the reverse of esterification (Figure 5.5), which is a reversible reaction. The reaction is catalyzed by diluted acid, such as diluted hydrochloric acid, and is heated under reflux. As the reaction is reversible, excessive water has to be used. Under such condition, carboxylic acid and alcohol are produced. Through this reaction, a ligand containing ester groups can be broken down into two child ligands.

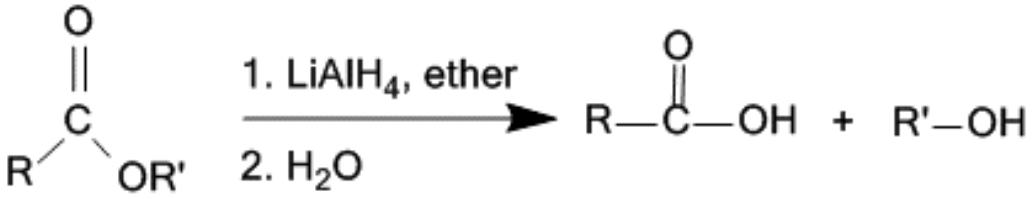


Figure 5.5: Hydrolysis of ester.

5.4.3 WebGL visualizer

Perhaps the most vital difference that distinguishes iSyn apart from many other FBDD tools is the availability of a WebGL visualizer, which is not only user friendly, but also of high performance. Unlike Java applet-based visualizers that require Java installation and depend on software rendering which is slow on large display areas and prevents detailed inspection of the structure, iSyn's WebGL visualizer is refactored from our iview [11] using three.js as its primary 3D engine with anti-aliasing support, and benefits from GPU hardware acceleration. Because of no dependency on any third-party browser plugins, our visualizer demonstrates excellent portability and usability.

The visualizer uses EDTSurf [158, 159], a fast algorithm to generating triangulated macromolecular surfaces by Euclidean distance transform, to construct and render in real time four representations of protein surface, namely Van der Waals surface, solvent excluded surface, solvent accessible surface and molecular surface. Note that molecular surface is indeed solvent excluded surface, but EDTSurf uses different ways to derive them.

The visualizer supports certain kinds of user interactions in-

cluding rotation, translation, zooming and changing slab with mouse or hand touch manipulation. It is functional not only on desktop computers, but also on mobile devices such as Android phones and tablets that support WebGL.

It is noteworthy to point out that iSyn performs all sorts of parsing and rendering in the client browser, ensuring the data privacy and confidentiality are retained.

5.5 Results and discussion

Thus far there are no well-established systematic evaluation metrics and benchmarks for FBDD tools, therefore selected examples are often used for testing purpose, as was the case in AutoGrow [170, 171]. To demonstrate the utility of our iSyn in generating novel ligands *ex nihilo*, we designed predicted inhibitors of two important drug targets, which are RNA editing ligase 1 (REL1) from Trypanosoma brucei, the etiological agent of African sleeping sickness, and cyclin-dependent kinase 2 (CDK2), a positive regulator of eukaryotic cell cycle progression.

We evaluated and compared our iSyn and the state-of-the-art AutoGrow 3.0 from the perspectives of the lowest predicted free energy obtained and the program execution time on a Linux server equipped with 2 Xeon E5-2670 @ 2.6GHz and 128GB ECC DDR3.

5.5.1 Inhibitors of Trypanosoma brucei RNA editing ligase 1

As TbREL1 is crucial for the survival of the Trypanosoma brucei parasite, it has been the target of several drug discovery projects over recent years.

PDB entry 1XDN was used. 75 initial ligands were chosen as input from the MW250 subset of the AutoGrow 3.0 fragment library by randomly picking 5 fragments from each of the 15 categories.

AutoGrow 3.0 was run for 2 days and 10 hours for 17 generations, and the best resultant compound had predicted free energy of -12.7 kcal/mol.

iSyn was run for 6 hours and 40 minutes for 2 generations, and the best resultant compound, 2_1314_1, had predicted free energy of -14.176 kcal/mol, with as many as 17 putative hydrogen bonds (Figure 5.6).

iSyn is capable of tracking the synthetic path of child ligands from their ancestors. Figure 5.7 shows the evolutionary steps taken to produce 2_1314_1, whose starting ligand had predicted free energy of -9.671 kcal/mol. After 2 generations, the best ligand 2_1314_1 had 4.505 kcal/mol lower predicted free energy. Since the value is in logarithmic scale, it effectively translates to 2016 fold increase in drug potency. In other words, a small concentration of the 2_1314_1 molecule would be sufficient to modulate the biological function of the TbREL1 target.

It can also be seen that its parent ligand, 2_1314, had a

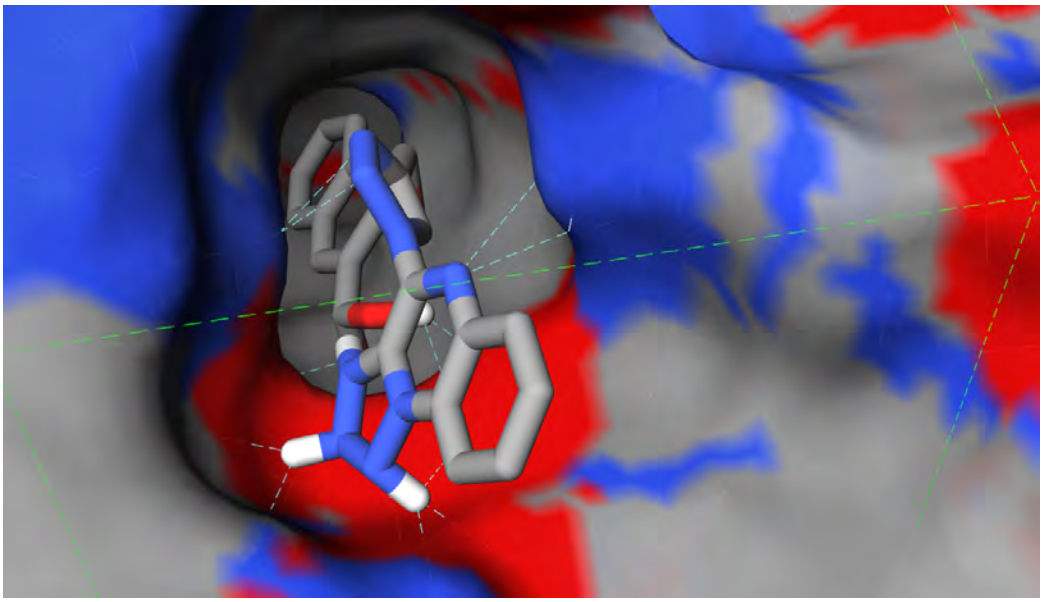


Figure 5.6: TbREL1 in complex of 2_1314_1.

molecular mass of as large as 624.211 Daltons, which is unlikely to be optimized into potent drugs. Thanks to our four novel cutting operators, it got broken down and resulted in the ever best ligand. This demonstrates the helpfulness of our newly-implemented cutting operators with click chemistry support.

In another run of iSyn with the same target, 20,392 initial ligands from the MW250 subset were used as input. iSyn was run for 2 days and 4 hours for 3 generations, and the best resultant compound, Gen2_m24517, had predicted free energy of -15.393 kcal/mol. This indicates that iSyn was able to generate even better ligands when more generations were achieved. Figure 5.8 shows the evolutionary steps taken to generate Gen2_m24517.

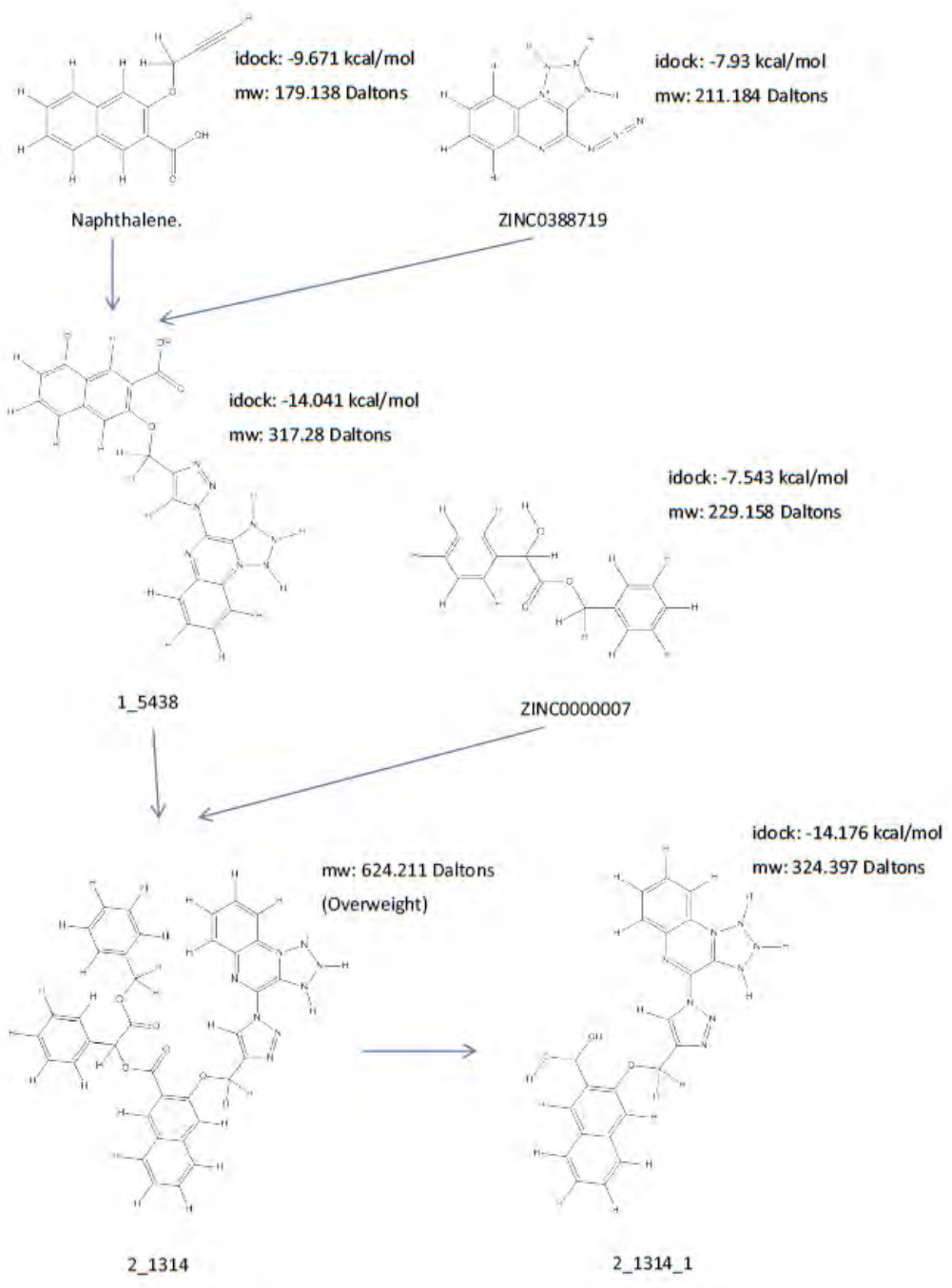


Figure 5.7: The evolutionary steps taken to generate 2_1314_1.

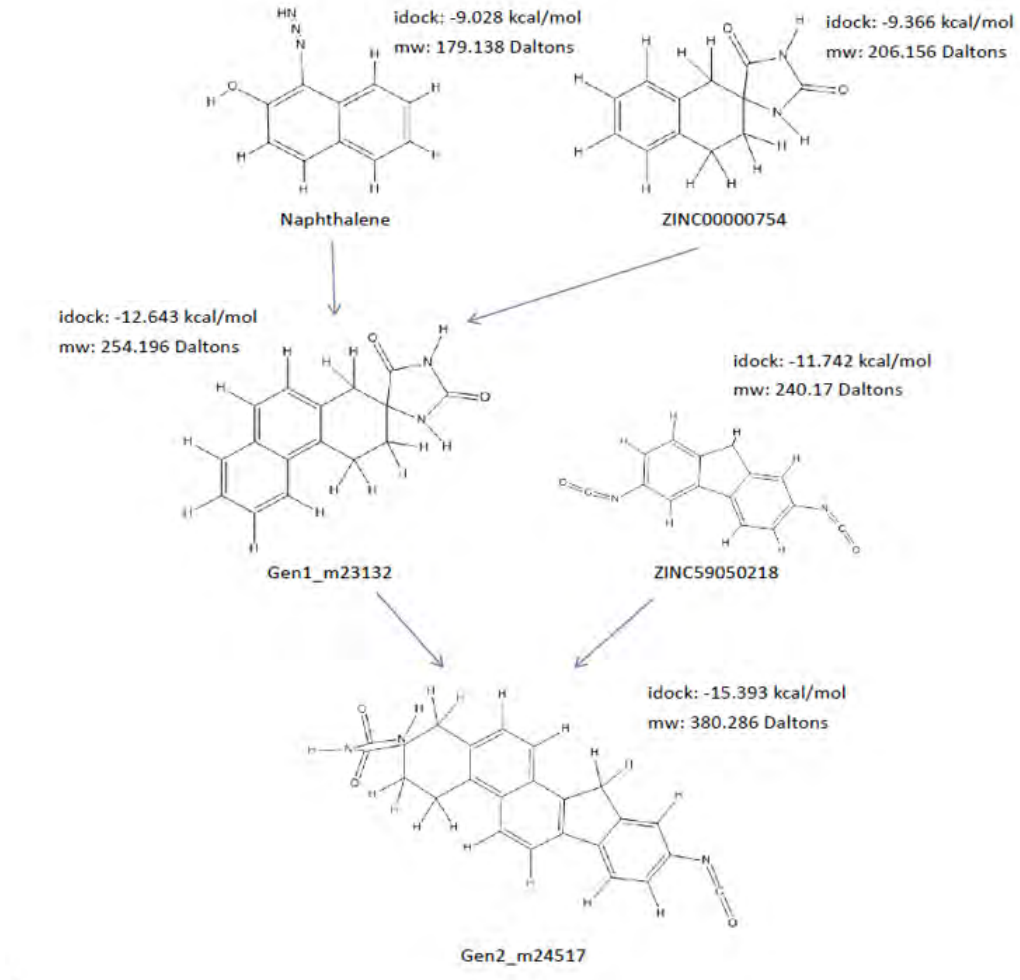


Figure 5.8: The evolutionary steps taken to generate Gen2_m24517.

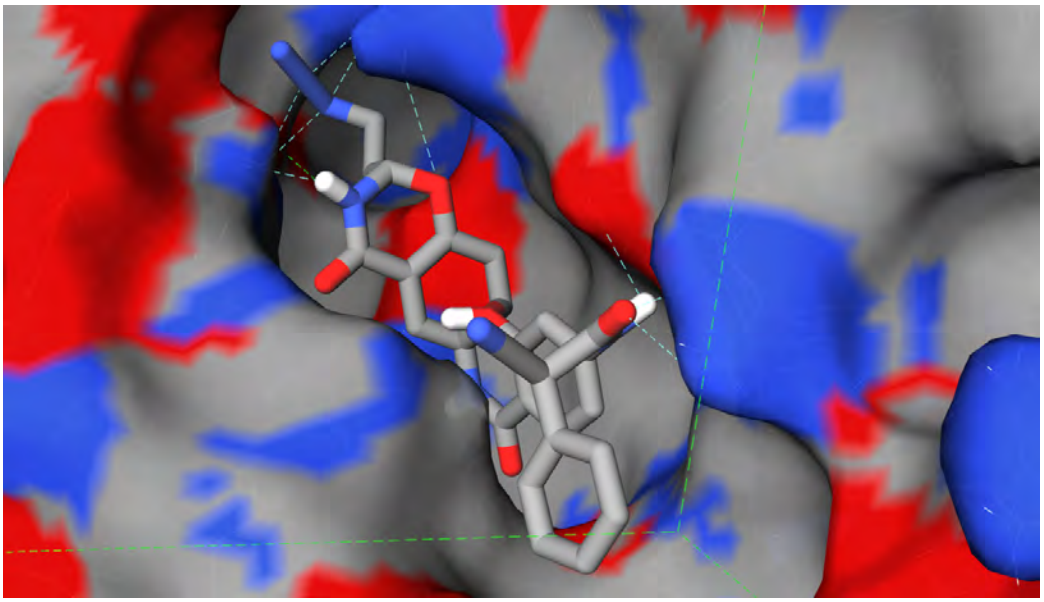


Figure 5.9: CDK2 (PDB: 1JSV) in complex of the best ligand.

5.5.2 Inhibitors of cyclin-dependent kinase 2

CDK2 is a member of the cyclin-dependent kinase family that are potential therapeutic targets for oncology. Inhibition of CDK2 may represent a therapeutic strategy for prevention of many cell cycle related diseases.

PDB entry 1JSV was used. Likewise, a number of ligands were chosen randomly from the fragment library to act as initial ligands.

iSyn was run for 11 hours and 20 minutes for 4 generations, and the best resultant compound had predicted free energy of -11.345 kcal/mol, with 9 putative hydrogen bonds (Figure 5.9). For comparison, its starting ligand had predicted free energy of only -5.607 kcal/mol.

In another run of iSyn with the same target, PDB entry

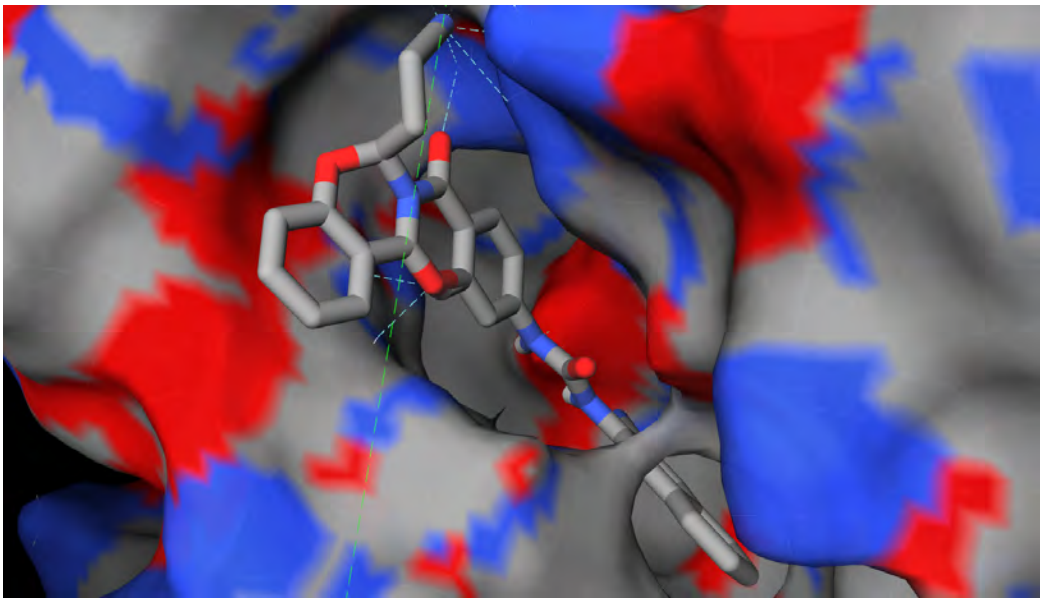


Figure 5.10: CDK2 (PDB: 1PXM) in complex of the best ligand.

1PXM was used. We would like to test whether iSyn could produce consistent results given different PDB entries of the same protein.

iSyn was run for 14 hours and 30 minutes for 5 generations, and the best resultant compound had predicted binding affinity of -14.071 kcal/mol, with 12 putative hydrogen bonds (Figure 5.10).

Obviously, the best ligand obtained in this example is better than the one obtained in the previous example. This could be due to several reasons. The major reason is most likely the stochastic nature of the evolutionary algorithm. Different optimization runs typically lead to different results. Another reason is the possible conformational differences in the two protein-ligand complex entries, which could affect intermolecular bind-

ing. A third reason is the difference in the center and size of the binding site. Nevertheless, in both examples the best ligand still had predicted free energy lower than -11 kcal/mol, indicating that iSyn managed to produce very potent ligands in both cases.

In another run of iSyn with the same target, PDB entry 1PF8 was used. Its experimentally measured binding affinity is K_i=31nM. Within just 4 hours and 10 minutes, iSyn generated the best ligand with predicted free energy of -14.4 kcal/mol, which translates to 27pM in potency. This is over one thousand times more potent than the crystal ligand.

5.6 Conclusions

Although *in silico* fragment-based drug design (FBDD) represents a promising approach to complement structure-based virtual screening, few FBDD tools show satisfactory performance in terms of achieved potency and computational resources.

In this study we have presented iSyn, our WebGL-based solution for computationally synthesizing *de novo* drug compounds with click chemistry support plus additional cutting reactions. iSyn is a methodological mixture of ligand duplication by USR [19], four new cutting reactions, efficient docking by idock [7], accurate rescoring by RF-Score-v3, fitness proportionate selection and intelligent termination in genetic algorithm, and WebGL visualization with molecular surface. Various test cases on TbREL1 and CDK2 have proved its strength in finding candi-

date drug compounds within a reasonable time. We hope that the iSyn can pragmatically assist medicinal chemists in optimizing candidate compounds and designing novel drugs.

5.7 Availability

iSyn is written in C++, Python, HTML5 and JavaScript. It is free and open source, available at http://istar.cse.cuhk.edu.hk/iSyn.tgz. It has been tested successfully on both Linux and Windows.

5.8 Future works

There are some major weak points about iSyn, though. iSyn requires ligands in PDB format to perform the genetic operators, but it requires PDBQT format to perform docking and scoring. So every generated ligand must undergo PDB-to-PDBQT conversion, which incurs substantial overhead. Moreover, one user recently found a bug in iSyn that after a cutting reaction, iSyn discarded the large fragment and retained the small one, which was subsequently discarded again.

We are developing a new project called igrow to directly manipulate ligands in PDBQT format. A further attempt would be to incorporate igrow into idock to better speed up the docking process. It is also inspiring to design multitarget ligands [177, 188].

□ **End of chapter.**

Chapter 6

RF::Cyscore: binding affinity prediction

State-of-the-art protein-ligand docking methods are generally constrained by the traditionally low accuracy of their scoring functions, which are for binding affinity prediction and thus vital for discriminating between active and inactive compounds. Despite intensive research over the years, classical scoring functions have reached a plateau in their predictive performance. They assume a predetermined additive functional form for some sophisticated numerical features, and use standard multivariate linear regression (MLR) on experimental data to derive the coefficients.

In this study we show that such a simple functional form is detrimental for the predictive performance of a scoring function, and replacing linear regression by machine learning techniques like random forest (RF) can improve predictive performance. We investigated the conditions of applying RF under various contexts and found that given sufficient training samples RF

managed to comprehensively capture the non-linearity between structural features and measured binding affinities. Incorporating more structural features and training with more samples could both boost RF performance. In addition, we analyzed the importance of structural features to binding affinity prediction using the RF variable importance tool. Lastly, we used Cyscore, a top performing empirical scoring function, as a baseline for comparison study.

In conclusion, machine-learning scoring functions are fundamentally different from classical scoring functions because the former circumvents the fixed functional form relating structural features with binding affinities. RF, but not MLR, can effectively exploit more structural features and more training samples, leading to higher predictive performance. The future availability of more X-ray crystal structures will further widen the performance gap between RF-based and MLR-based scoring functions. This further stresses the importance of substituting RF for MLR in scoring function development.

This was a collaborative project with Pedro J. Ballester from Cancer Research Center of Marseille, Marseille, France. It was published in *BMC Bioinformatics* on 27 August 2014 [15]. Notably, this article has been tagged "Highly accessed" by the journal, indicating that it may be of broad interest in the community.

6.1 Background

Protein-ligand docking is a computational tool that predicts how a ligand binds to a target protein and their binding affinity. Therefore docking is useful in elucidating intermolecular interactions and enhancing the potency and selectivity of binding in subsequent phases of the modern drug design process. Docking has a wide variety of pragmatic and successful applications in structure-based virtual screening [189], drug repurposing [190], lead compound optimization [191], protein cavity identification [192], and protein function prediction [193].

Docking performs two main operations: predicting the position, orientation and conformation of a ligand when docked to the protein's binding site, and predicting the binding strength. The former operation is known as pose generation and the latter is known as scoring. State-of-the-art docking tools, AutoDock Vina [8] and idock [7] for instance, work reasonably well at pose generation with a redocking success rate of over 50% [9] on the benchmarks of both PDBbind v2012 and v2011 [136–138] and the CSAR NRC HiQ Set 24Sept2010 [142, 143]. Nonetheless, the single most critical limitation of docking is the traditionally low accuracy of the scoring functions.

Classical scoring functions are defined by using an assumed, fixed functional form for the relationship between the numerical features that characterize the protein-ligand complex and its predicted binding affinity. This functional form consists of the energetic contributions of various intermolecular interactions,

and is often additive. The overall binding affinity is calculated as a weighted sum of some physically meaningful terms, whereas their coefficients are typically derived from standard multivariate linear regression (MLR) on experimental data.

Cyscore [14], a recently published empirical scoring function, assumed that the overall protein-ligand binding free energy can be divided into four terms: hydrophobic free energy, van der Waals interaction energy, hydrogen bond interaction energy and ligand's conformational entropy. Cyscore improved the prediction of hydrophobic free energy using a novel curvature-dependent surface-area model, which was claimed to be able to distinguish convex, planar and concave surface in the calculation of hydrophobic free energy.

A recent study on a congeneric series of thrombin inhibitors concluded that free energy contributions to ligand binding at the molecular level are non-additive [194], therefore the modelling assumption of additivity models is error prone. Recent years have seen a growing number of new developments of machine-learning scoring functions, with RF-Score [10] being the first that introduced a large improvement over classical approaches. RF-Score, as its name suggests, uses Random Forest (RF) [139] to implicitly learn the functional form in an completely data-driven manner, and thus circumvents the modelling assumption imposed by previous scoring functions. RF-Score was shown to significantly outperform 16 classical scoring functions when evaluated on the commonly-used PDBbind v2007 benchmark [10]. Despite being a recent development, RF-Score has already

been successfully used to discover a large number of innovative binders against antibacterial DHQase2 targets [140]. For the purpose of prospective virtual screening, RF-Score-v3 has now been incorporated into istar [9], our large-scale online docking service available at http://istar.cse.cuhk.edu.hk/idock. A number of subsequent machine-learning scoring functions have also shown large improvements over classical approaches. These include, but are not limited to, NNScore 2.0 [54], SVR-KB and SVR-EP [195], CScore [196], SVR-Score [149], B2Bscore [197], SFCscoreRF [52], and ID-Score [55].

6.2 Motivation

Despite the superior performance of RF-Score, its generalization power has not yet been evaluated under standard cross validation and leave-cluster-out cross validation [198]. It would be interesting to see if substituting RF for MLR can improve the predictive performance of a classical scoring function with a fixed functional form.

6.3 Objective

In this study we compare the predictive performance of two regression models MLR and RF when trained with varying numbers of structural features and training samples, and investigate their application conditions and interpretability in various contexts. We use Cyscore as a baseline.

6.4 Methods

The following subsections introduce MLR and RF, three sets of features, three benchmarks, two types of cross validations, and four performance metrics.

6.4.1 Multiple Linear Regression (MLR) with Cyscore features

Cyscore is an empirical scoring function in an additive functional form of four energetic terms: hydrophobic free energy $\Delta G_{hydrophobic}$, van der Waals interaction energy ΔG_{vdw}, hydrogen bond interaction energy ΔG_{hbond} and ligand's conformational entropy $\Delta G_{entropy}$ (equation (6.1)). Their coefficients k_h, k_v, k_b and k_e and the intercept C were obtained by MLR on 247 high-quality complexes carefully selected from PDBbind v2012 refined set. The intercept value was not reported in the original publication, but was included in this study as usual [148] in order to quickly estimate the absolute binding affinity value, which is the ultimate goal in some real-life applications.

$$\Delta G_{bind} = k_h \Delta G_{hydrophobic} + k_v \Delta G_{vdw} + k_b \Delta G_{hbond} + k_e \Delta G_{entropy} + C \tag{6.1}$$

We use MLR::Cyscore to denote the scoring function built with MLR and the 4 features from Cyscore. It is noteworthy that Cyscore is a sheer MLR model, unlike AutoDock Vina [8] which is a quasi MLR model because the number of rotatable

bonds N_{rot} is in the denominator in order to penalize ligand flexibility (see [9] for the exact equation) and therefore MLR::Vina would require an additional grid search for the weight of the N_{rot} parameter. Hence this study allows a more direct comparison between MLR and RF.

6.4.2 Random Forest (RF) with Cyscore, AutoDock Vina and RF-Score features

A RF [139] is a consensus of a large number of different decision trees generated from random bootstrap sampling of the same training data. During tree construction, at each inner node RF chooses the best splitting feature that results in the highest purity gain from a normally small number (mtry) of randomly selected features instead of utilizing all input features. In the case of regression, the final output is computed as the arithmetic mean of all individual tree predictions in the RF. More details on RF construction can be found in [9, 10].

In this study, multiple RFs of the default number of 500 trees were built using values of the mtry control parameter from one to the total number of input features. The selected RF was the one that resulted in the lowest root mean square error (RMSE) on the Out-of-Bag (OOB) samples of the training set. We used just one single random seed for training because seed is not a significant impact factor of the predictive performance. Using fewer seeds also has the advantage of computationally faster training process.

Table 6.1: The three combinations of three different sets of features used to train RF models.

model	features
RF::Cyscore	4 in Cyscore
RF::CyscoreVina	4 in Cyscore, 6 in AutoDock Vina
RF::CyscoreVinaElem	4 in Cyscore, 6 in AutoDock Vina, 36 in RF-Score

In our experiments we aimed at analyzing how RF responds to varying numbers of features. So we selected three sets of features: Cyscore [14], AutoDock Vina [8] and RF-Score [10]. Cyscore comprises four numerical features: $\Delta G_{hydrophobic}$, ΔG_{vdw}, ΔG_{hbond} and $\Delta G_{entropy}$. AutoDock Vina comprises six numerical features: $Gauss_1$, $Gauss_2$, $Repulsion$, $Hydrophobic$, $HBonding$ and N_{rot}. RF-Score comprises 36 features, defined as the number of intermolecular contacts between two elemental atom types. Four atom types for proteins (C, N, O, S) and nine for ligands (C, N, O, S, P, F, Cl, Br, I) were selected so as to produce a dense set of features while considering all the heavy atom types commonly observed in protein-ligand complexes. Table 6.1 summarizes the three combinations of these feature sets used to train RF models. Totally four models, MLR::Cyscore, RF::Cyscore, RF::CyscoreVina and RF::CyscoreVinaElem, were evaluated in this study.

6.4.3 PDBbind v2007 and v2012 benchmarks

The PDBbind [136–138] benchmark is arguably the most widely used for binding affinity prediction. It is a diverse collection of experimentally resolved protein-ligand complexes, assembled

through a systematic mining of the yearly releases of the entire PDB (Protein Data Bank) [22, 144]. For each complex, the experimentally measured binding affinity, in terms of either dissociation constant Kd or inhibition constant Ki, was manually collected from its primary literature reference. The complexes with a resolution of $\leq$2.5Å and with the ligand comprising only nine common heavy atom types (C, N, O, F, P, S, Cl, Br, I) were filtered to constitute the refined set. These complexes were then clustered by protein sequence identity with a cutoff of 90%, and for each of the resulting clusters with at least five complexes, the three complexes with the highest, median and lowest binding affinity were selected to constitute the core set. Due to the structural diversity of the core set, it is a common practice to use the core set as a test set and the remaining complexes in the refined set as a training set.

Cyscore was tested on two independent sets: PDBbind v2007 core set (N=195) and PDBbind v2012 core set (N=201), whose experimental binding affinities span 12.56 and 9.85 pKd units, respectively. Cyscore was trained on a special set of 247 complexes carefully selected from the PDBbind v2012 refined set using certain criteria [14] (e.g. structural resolution $<$ 1.8Å, binding affinity spans 1 to 11 kcal/mol, protein sequence similarity and ligand chemical composition are different from the test set), ensuring that the training complexes are of high quality and do not overlap with any of the two test sets. In this study in order to make a fair comparison to Cyscore we used exactly the same training and test sets.

Moreover, in view of the fact that 16 classical scoring functions have already been evaluated [148] on PDBbind v2007 core set and the top performing of them (e.g. X-Score [150]) were trained on the remaining 1105 complexes in PDBbind v2007 refined set, we also used these 1105 complexes to constitute another training set to permit a direct comparison. Using predefined training and test sets, where other scoring functions had previously been trained and tested, has the advantage of reducing the risk of using a benchmark complementary to one particular scoring function.

Similarly for the PDBbind v2012 benchmark, we used an extra training set comprising the complexes in PDBbind v2012 refined set excluding those in PDBbind v2012 core set. This led to a total of 2696 complexes. By construction, this training set does not overlap with the test set.

6.4.4 PDBbind v2013 round-robin benchmark

We propose a new benchmark with the purpose to investigate how predictive performance of the four models changes in cross validation and with different numbers of training samples. We used PDBbind v2013 refined set (N=2959), which was the latest version at the time of writing and constituted the most comprehensive and publicly available structural dataset suitable for training scoring functions.

We used 5-fold cross validation, as was used by the recently published empirical scoring function ID-Score [55], to estimate

Table 6.2: Statistics of the five partitions of PDBbind v2013 refined set.

#	complexes	lowest pKd	highest pKd
1	592	2.00	11.74
2	592	2.00	11.80
3	592	2.00	11.85
4	592	2.00	11.92
5	591	2.05	11.72

overfitting and thus generalization errors. The entire PDBbind v2013 refined set (N=2959) was decomposed into five equal partitions using uniform sampling on a round-robin basis: the entire 2959 complexes were first sorted in the ascending order of their measured binding affinity, and the complexes with the 1st, 6th, 11th, etc. lowest binding affinity belonged to the first partition, the complexes with the 2nd, 7th, 12th, etc. lowest binding affinity belonged to the second partition, and so on. This round-robin partitioning method, though not entirely random, has two advantages. On one hand, each partition is guaranteed to span the largest range of binding affinities and incorporates the largest structural diversity of different protein families. On the other hand, each partition is composed of a deterministic list of complexes, permitting reproducibility and comparisons in future studies. Table 6.2 summarizes the statistics of the five partitions. Figure 6.1 plots the binding affinity distribution of pKd values of the five partitions. Due to the round-robin sampling mechanism, the five histograms are quite balanced compared to one another.

We then used the partition on which the optimal performance was obtained (It turned out to be partition 2 (N=592). See the

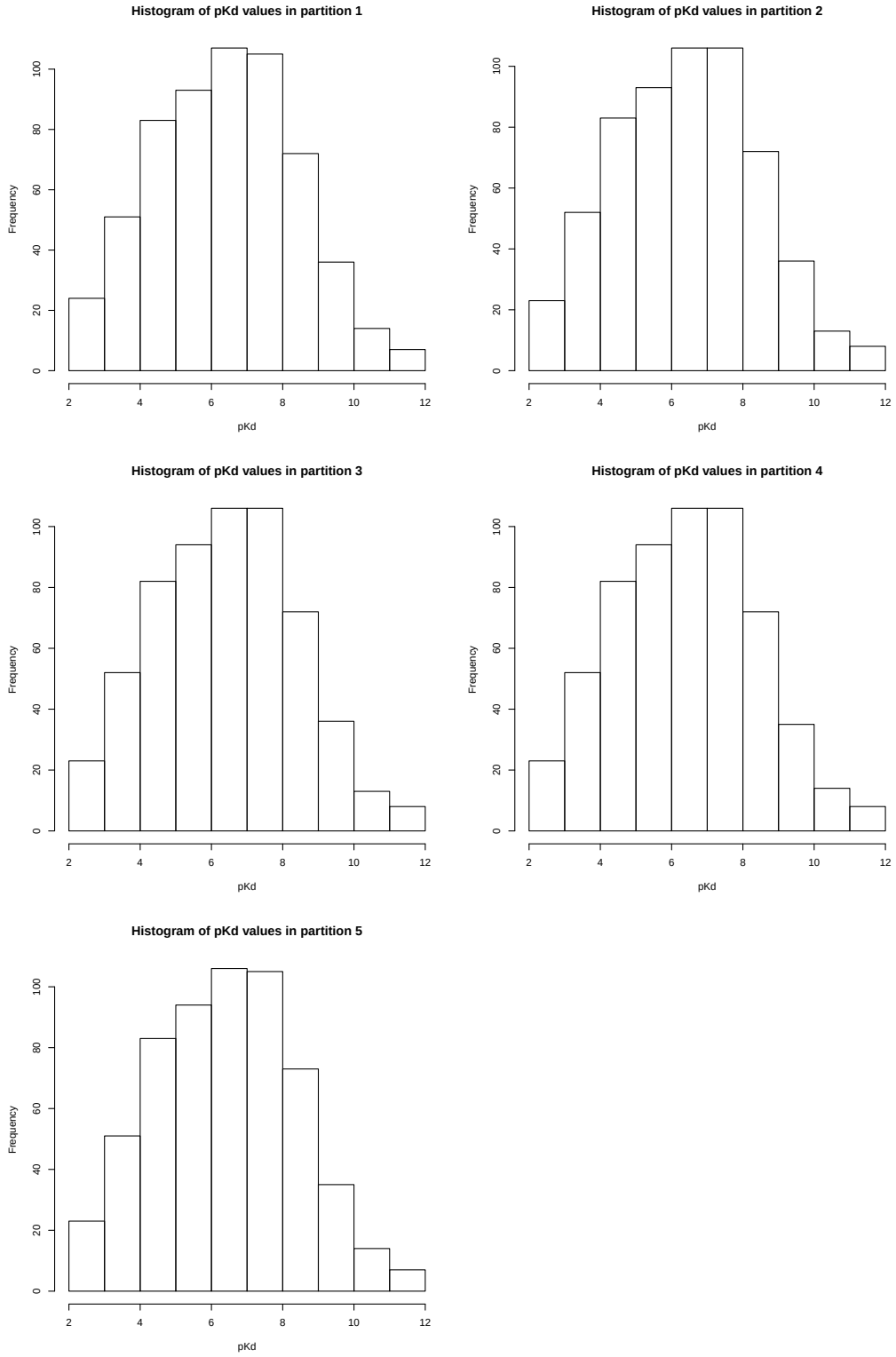


Figure 6.1: Histograms of pKd values of the five partitions of PDBbind v2013 refined set.

Table 6.3: The numbers of test samples and training samples for the PDBbind v2007, v2012 and v2013 benchmarks.

benchmark	test samples	training samples
v2007	195	247, 1105
v2012	201	247, 2696
v2013	592	592, 1184, 1776, 2367

Results section below.) as the test set in PDBbind v2013 round-robin benchmark, and used the remaining four partitions (1, 3, 4, 5) to construct four training sets of incremental sizes: the first training set comprises partition 1 (N=592), the second training set comprises partitions 1 and 3 (N=1184), the third training set comprises partitions 1, 3 and 4 (N=1776), and the fourth training set comprises partitions 1, 3, 4 and 5 (N=2367). By construction, this new benchmark helps to study how predictive performance varies with training set size. Moreover, its test set has a substantially larger number of complexes (N=592) compared to PDBbind v2007 (N=195) and v2012 (N=201) benchmarks, making this new benchmark not being a redundant duplication of the previous two benchmarks. Table 6.3 summarizes the numbers of test and training samples for the three benchmarks.

6.4.5 Leave-cluster-out cross validation (LCOCV)

Leave-cluster-out cross validation (LCOCV) [198], in contrast to standard cross validation, divides the whole set of complexes into protein families instead of random subsets. Each protein family, or each cluster, is typically determined by 90% protein sequence

identity. A total of 23 selected protein families with at least ten complexes are treated as individual clusters, labeled as A to W. Protein families with four to nine complexes are combined into cluster X. Protein families with two to three complexes are combined into cluster Y. Singletons are combined into cluster Z. Each cluster is iteratively left out of the training set and used to evaluate the predictive performance of the scoring function. The performance on each cluster can be inspected individually, and the overall performance can be estimated by averaging over all clusters.

So far LCOCV has been applied to the assessment of six scoring functions, which are RF-Score [196–198], ddPLAT+MOE [199], CScore [196], B2Bscore [197], SFCscoreRF [52] and the work by Ross et al. [200]. The first four scoring functions were evaluated on PDBbind v2009 refined set, while SFCscoreRF was on PDBbind v2010 refined set and the work by Ross et al. was on PDBbind v2011 refined set. For the purpose of comparison to these scoring functions, we used PDBbind v2009 refined set (N=1741) to perform LCOCV. We discarded the 1xr8 entry in cluster X because its ligand is far away from its protein, thereby leaving 1740 complexes. Figure 6.2 plots the binding affinity distribution of pKd values of the 26 clusters. Unlike Figure 6.1, here the histograms have discrepant shapes and magnitudes, suggesting that each cluster has its unique properties in the binding affinity range.

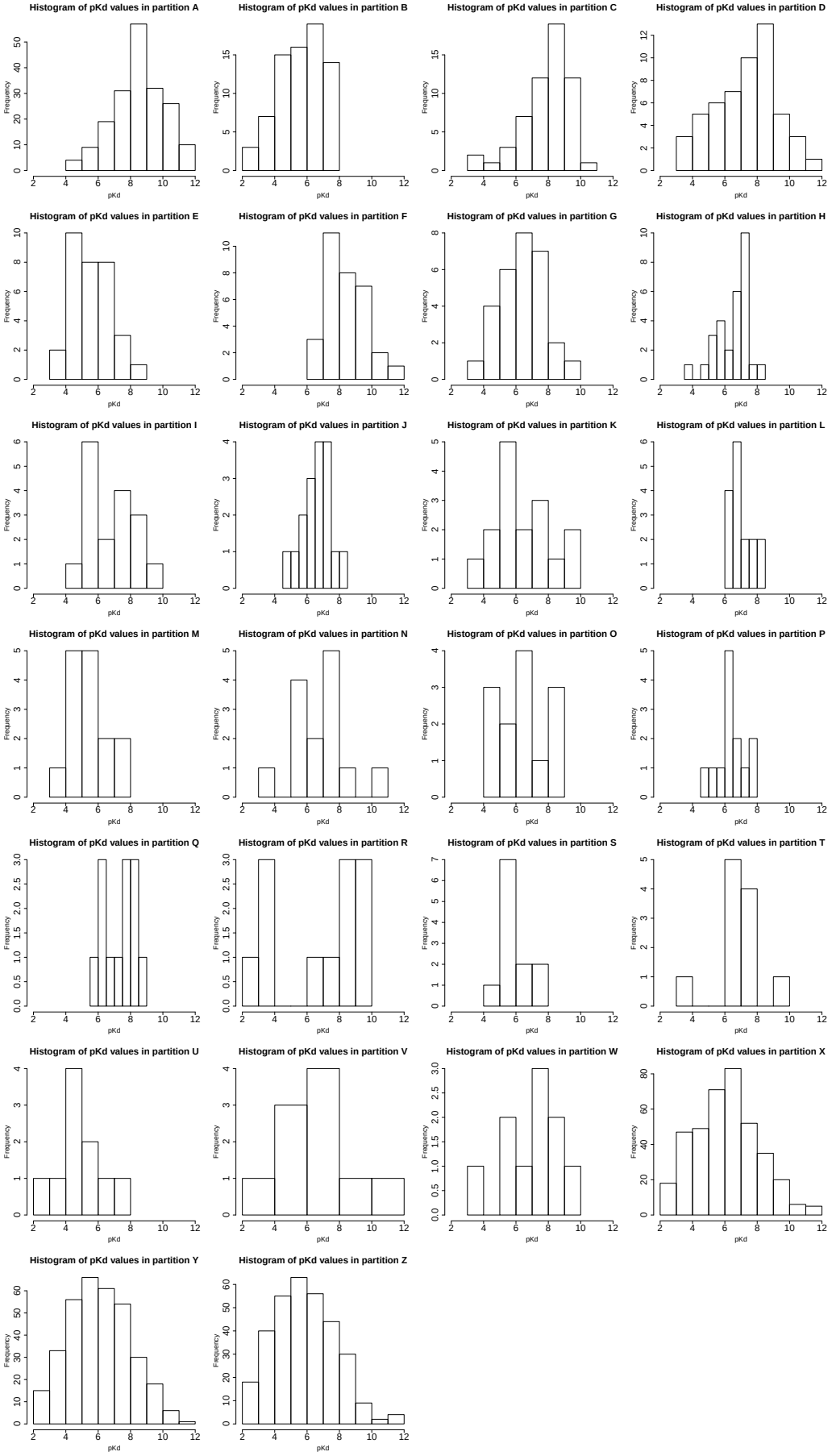


Figure 6.2: Histograms of pKd values of the 26 clusters of PDBbind v2009 refined set.

6.4.6 Performance metrics

Predictive performance was quantified through standard deviation SD in linear correlation, Pearson correlation coefficient Rp and Spearman correlation coefficient Rs between the measured and predicted binding affinities of the test set. These metrics are commonly used in the community [148]. The SD metric is essentially the residual standard error (RSE) metric used in some other studies [195]. The above three metrics are invariant under linear transformations. Changing the intercept or coefficient values in equation (6.1) affects none of these metrics. So they are mainly for comparative purpose. In some applications, however, the ultimate goal of scoring functions is to predict an absolute binding affinity value as close to the measured value as possible. Therefore we use a more realistic metric, the root mean square error RMSE between measured and predicted binding affinities without coupling a linear correlation. Lower values in RMSE and SD and higher values in Rp and Rs indicate better predictive performance.

Mathematically, equations (6.2), (6.3), (6.4) and (6.5) show the expressions of the four metrics. Given a scoring function f and the features $\mathbf{x}_i$ characterizing the ith complex out of n complexes in the test set, $p_i = f(\mathbf{x}_i)$ is the predicted binding affinity, $\{\hat{p}_i\}_{i=1}^n$ are the fitted values from the linear model between $\{y_i\}_{i=1}^n$ and $\{p_i\}_{i=1}^n$ on the test set, whereas $\{y_i^r\}_{i=1}^n$ and $\{p_i^r\}_{i=1}^n$ are the rankings of $\{y_i\}_{i=1}^n$ and $\{p_i\}_{i=1}^n$, respectively.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(p_i - y_i)^2} \tag{6.2}$$

$$SD = \sqrt{\frac{1}{n-2}\sum_{i=1}^{n}(\hat{p}_i - y_i)^2} \tag{6.3}$$

$$R_p = \frac{n\sum_{i=1}^{n} p_i y_i - \sum_{i=1}^{n} p_i \sum_{i=1}^{n} y_i}{\sqrt{(n\sum_{i=1}^{n}(p_i)^2 - (\sum_{i=1}^{n} p_i)^2)(n\sum_{i=1}^{n}(y_i)^2 - (\sum_{i=1}^{n} y_i)^2)}} \tag{6.4}$$

$$R_s = \frac{n\sum_{i=1}^{n} p_i^r y_i^r - \sum_{i=1}^{n} p_i^r \sum_{i=1}^{n} y_i^r}{\sqrt{(n\sum_{i=1}^{n}(p_i^r)^2 - (\sum_{i=1}^{n} p_i^r)^2)(n\sum_{i=1}^{n}(y_i^r)^2 - (\sum_{i=1}^{n} y_i^r)^2)}} \tag{6.5}$$

6.5 Results and discussion

6.5.1 MLR::Cyscore performance does not increase with more training samples

Figure 6.3 plots the predictive performance of MLR::Cyscore, RF::Cyscore, RF::CyscoreVina and RF::CyscoreVinaElem using different numbers of training samples. The first row is for root mean square error RMSE, the second row is for standard deviation SD in linear correlation, the third row is for Pearson correlation coefficient Rp, and the fourth row is for Spearman correlation coefficient Rs. The left column is for PDBbind v2007 benchmark (N=195), the center column is for PDBbind v2012

benchmark (N=201), and the right column is for PDBbind v2013 round-robin benchmark (N=592).

On both PDBbind v2007 and v2012 benchmarks, MLR::Cyscore performed best when it was trained on the 247 carefully selected complexes used by Cyscore [14]. Its performance dropped when more complexes were used for training. On PDBbind v2013 round-robin benchmark, MLR::Cyscore performance stayed nearly flat regardless of training set sizes.

These results show that MLR::Cyscore cannot exploit large sets of structural data given only a small set of sophisticated features. Feeding more training samples to MLR::Cyscore actually increases the difficulty in regressing the coefficients well. Generally it would be a good idea to select the training complexes that provide the best performance on a test set, as was the case of Cyscore. But in real applications the binding affinities of the test set are not known and unfortunately selection of training complexes is not performed blindly without measuring performance on test set.

6.5.2 RF performance increases with more structural features and training samples

As seen from Figure 6.3, on all the three benchmarks, given the same set of features, the RF models trained with more samples resulted in higher predictive accuracy. Likewise, given the same training samples, the RF models trained with more features resulted in higher predictive accuracy.

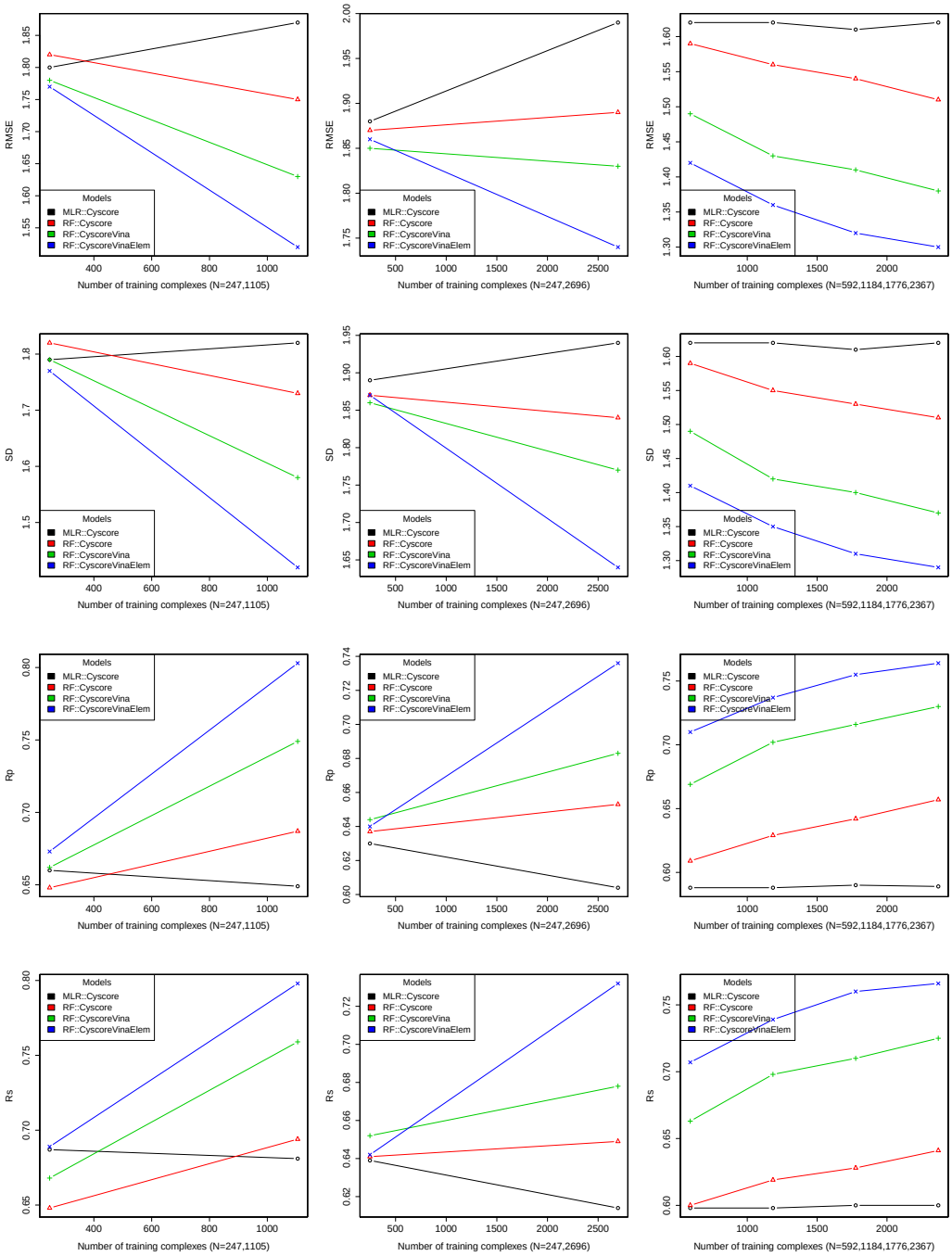


Figure 6.3: Predictive performance of MLR::Cyscore, RF::Cyscore, RF::CyscoreVina and RF::CyscoreVinaElem trained with varying numbers of samples.

These results suggest that RF is able to effectively exploiting a comprehensive set of structural features and training samples. Generally the more training samples, the more knowledge for RF to learn so as to capture the non-linearity of the structural data. Likewise, the more appropriate features, the higher probability of choosing the best splitting feature that can result in a high purity gain at non-leaf nodes during RF construction, and thus the higher chance of improved RF performance.

6.5.3 RF models perform consistently well in cross validation

Table 6.4 shows the results of 5-fold cross validation for all the four models on the five partitions of PDBbind v2013 refined set (N=2959). Interestingly, the four models all exhibited the best performance on partition 2. In terms of average performance, the relative ranking is consistent, where RF::CyscoreVinaElem (RMSE=1.35, SD=1.35, Rp=0.738, Rs=0.738) is better than RF::CyscoreVina (RMSE=1.44, SD=1.44, Rp=0.693, Rs=0.690), which is better than RF::Cyscore (RMSE=1.59, SD=1.59, Rp=0.603, Rs=0.587), which is better than MLR::Cyscore (RMSE=1.66, SD=1.66, Rp=0.556, Rs=0.559). Indeed the consistency of relative ranking holds for each individual partition. Although only partition 2 was used as the test set, we believe consistent conclusions can be drawn if any other partition is selected as the test set.

Table 6.4: Cross validation results of the four models on the five partitions of PDBbind v2013 refined set.

#	N	RMSE	SD	Rp	Rs
MLR::Cyscore					
1	592	1.66	1.66	0.560	0.555
2	592	1.62	1.62	0.589	0.600
3	592	1.69	1.70	0.531	0.529
4	592	1.68	1.68	0.542	0.557
5	591	1.65	1.65	0.559	0.553
avg		1.66	1.66	0.556	0.559
RF::Cyscore					
1	592	1.60	1.60	0.601	0.588
2	592	1.51	1.51	0.657	0.641
3	592	1.66	1.66	0.561	0.545
4	592	1.63	1.63	0.580	0.576
5	591	1.57	1.57	0.615	0.586
avg		1.59	1.59	0.603	0.587
RF::CyscoreVina					
1	592	1.41	1.41	0.708	0.709
2	592	1.38	1.37	0.730	0.725
3	592	1.49	1.49	0.668	0.665
4	592	1.51	1.51	0.657	0.661
5	591	1.42	1.42	0.701	0.692
avg		1.44	1.44	0.693	0.690
RF::CyscoreVinaElem					
1	592	1.33	1.33	0.748	0.746
2	592	1.30	1.29	0.764	0.766
3	592	1.41	1.41	0.711	0.709
4	592	1.41	1.41	0.711	0.722
5	591	1.30	1.30	0.758	0.749
avg		1.35	1.35	0.738	0.738

6.5.4 Leave-cluster-out cross validation leads to unrealistically low performance

Tables 6.5, 6.6, 6.7 and 6.8 show the results of leave-cluster-out cross validation (LCOCV) for all the four models on the 23 protein families (A to W) and 3 multi-family clusters (X to Z) of PDBbind v2009 refined set. Not unexpectedly, the observed performance is considerably heterogeneous across the different protein families. These results indeed agree with the LCOCV results of six other scoring functions from previous studies [52, 196–200]. Having analyzed the LCOCV statistics of all these ten scoring functions, we found that they all performed well in certain clusters (e.g. trypsin and β-secretase I) and poorly in some other clusters (e.g. HIV protease and factor Xa). The reasons for the large spread of performance across the different clusters are manifold, and a comprehensive analysis for each protein family is beyond the scope of this study. As pointed out in [52], eliminating all the HIV protease complexes leads to an imbalance between the training and test sets because HIV protease inhibitors are on average much larger than the ligands of the other targets. This illustrates that the LCOCV results should not be directly interpreted as performance measures on particular protein families. Moreover, the small size of many clusters and the limited range of measured binding affinity values therein render a satisfactory prediction of the ranking rather challenging.

While results on standard cross validation might be too op-

Table 6.5: Leave-cluster-out cross validation results of MLR::Cyscore.

cluster name	cluster	N	RMSE	SD	Rp	Rs
HIV protease	A	188	1.65	1.53	0.259	0.216
trypsin	B	74	1.24	1.11	0.612	0.695
carbonic anhydrase	C	57	2.47	1.35	0.473	0.343
thrombin	D	53	1.52	1.40	0.702	0.676
protein tyrosine phosphatase	E	32	1.23	1.06	0.411	0.313
factor Xa	F	32	1.18	0.96	0.604	0.634
urokinase	G	29	1.15	1.14	0.643	0.602
different similar transporters	H	29	0.96	0.96	0.285	0.122
c-AMP dependent kinase	I	17	1.32	1.15	0.537	0.537
β-glucosidase	J	17	1.03	0.78	0.383	0.316
antibodies	K	16	1.41	1.43	0.693	0.706
casein kinase II	L	16	0.75	0.58	0.538	0.358
ribonuclease	M	15	1.12	1.20	0.230	0.340
thermolysin	N	14	1.15	1.14	0.680	0.635
CDK2 kinase	O	13	1.06	0.80	0.841	0.812
glutamate receptor 2	P	13	1.08	0.85	0.070	0.096
P38 kinase	Q	13	0.55	0.57	0.834	0.896
β-secretase I	R	12	1.44	1.33	0.892	0.725
tRNA-guanine transglycosylase	S	12	0.90	0.95	0.463	0.544
endothiapepsin	T	11	1.18	1.30	0.435	0.215
α-mannosidase 2	U	10	1.67	1.63	-0.004	0.248
carboxypeptidase A	V	10	2.13	1.99	0.479	0.523
penicillopepsin	W	10	1.71	1.87	0.339	0.188
families with 4-9 complexes	X	386	1.73	1.71	0.500	0.577
families with 2-3 complexes	Y	340	1.64	1.64	0.510	0.495
singletons	Z	321	1.76	1.74	0.407	0.417
average			1.35	1.24	0.493	0.470
standard deviation			0.41	0.38	0.216	0.217

Table 6.6: Leave-cluster-out cross validation results of RF::Cyscore.

cluster name	cluster	N	RMSE	SD	Rp	Rs
HIV protease	A	188	1.70	1.51	0.310	0.201
trypsin	B	74	1.10	1.11	0.610	0.636
carbonic anhydrase	C	57	2.44	1.43	0.368	0.264
thrombin	D	53	1.50	1.44	0.680	0.611
protein tyrosine phosphatase	E	32	1.30	1.10	0.338	0.268
factor Xa	F	32	1.54	1.13	0.367	0.356
urokinase	G	29	1.10	1.14	0.642	0.645
different similar transporters	H	29	1.27	0.99	0.056	-0.040
c-AMP dependent kinase	I	17	1.16	1.11	0.582	0.602
β-glucosidase	J	17	1.04	0.76	0.444	0.365
antibodies	K	16	1.67	1.76	0.455	0.466
casein kinase II	L	16	0.76	0.58	0.535	0.330
ribonuclease	M	15	1.07	1.06	0.505	0.281
thermolysin	N	14	0.98	1.03	0.748	0.648
CDK2 kinase	O	13	1.14	1.01	0.733	0.817
glutamate receptor 2	P	13	1.09	0.85	0.120	0.097
P38 kinase	Q	13	0.76	0.66	0.762	0.757
β-secretase I	R	12	1.57	1.51	0.858	0.620
tRNA-guanine transglycosylase	S	12	1.06	1.04	0.212	0.375
endothiapepsin	T	11	1.28	1.35	0.358	0.210
α-mannosidase 2	U	10	1.65	1.62	0.116	0.188
carboxypeptidase A	V	10	1.90	1.89	0.556	0.370
penicillopepsin	W	10	1.78	1.94	0.236	0.188
families with 4-9 complexes	X	386	1.61	1.60	0.587	0.598
families with 2-3 complexes	Y	340	1.64	1.63	0.522	0.505
singletons	Z	321	1.81	1.75	0.397	0.395
average			1.38	1.27	0.465	0.414
standard deviation			0.38	0.37	0.209	0.212

Table 6.7: Leave-cluster-out cross validation results of RF::CyscoreVina.

cluster name	cluster	N	RMSE	SD	Rp	Rs
HIV protease	A	188	1.76	1.56	0.182	0.105
trypsin	B	74	0.96	0.97	0.723	0.700
carbonic anhydrase	C	57	2.60	1.37	0.448	0.372
thrombin	D	53	1.47	1.45	0.675	0.675
protein tyrosine phosphatase	E	32	1.36	0.98	0.538	0.542
factor Xa	F	32	1.53	1.02	0.533	0.498
urokinase	G	29	1.25	1.27	0.516	0.436
different similar transporters	H	29	1.10	0.98	0.188	0.077
c-AMP dependent kinase	I	17	0.94	0.91	0.748	0.664
β-glucosidase	J	17	0.92	0.72	0.518	0.443
antibodies	K	16	1.47	1.51	0.645	0.643
casein kinase II	L	16	0.90	0.60	0.493	0.322
ribonuclease	M	15	1.11	0.99	0.595	0.481
thermolysin	N	14	1.04	1.12	0.696	0.565
CDK2 kinase	O	13	1.14	1.02	0.729	0.661
glutamate receptor 2	P	13	1.08	0.85	0.116	0.121
P38 kinase	Q	13	0.95	0.62	0.799	0.764
β-secretase I	R	12	1.54	1.51	0.860	0.687
tRNA-guanine transglycosylase	S	12	0.87	0.95	0.457	0.403
endothiapepsin	T	11	1.35	1.36	0.345	0.215
α-mannosidase 2	U	10	1.73	1.62	0.089	0.176
carboxypeptidase A	V	10	1.82	1.76	0.632	0.467
penicillopepsin	W	10	1.81	1.96	0.183	0.030
families with 4-9 complexes	X	386	1.58	1.56	0.610	0.612
families with 2-3 complexes	Y	340	1.55	1.55	0.583	0.580
singletons	Z	321	1.70	1.68	0.476	0.467
average			1.37	1.23	0.515	0.450
standard deviation			0.39	0.36	0.211	0.211

Table 6.8: Leave-cluster-out cross validation results of RF::CyscoreVinaElem.

cluster name	cluster	N	RMSE	SD	Rp	Rs
HIV protease	A	188	1.77	1.56	0.166	0.129
trypsin	B	74	0.93	0.93	0.751	0.715
carbonic anhydrase	C	57	2.33	1.35	0.481	0.234
thrombin	D	53	1.46	1.40	0.699	0.680
protein tyrosine phosphatase	E	32	1.23	0.89	0.643	0.615
factor Xa	F	32	1.61	1.07	0.470	0.470
urokinase	G	29	1.05	1.06	0.699	0.624
different similar transporters	H	29	1.01	0.93	0.354	0.123
c-AMP dependent kinase	I	17	1.06	0.91	0.747	0.644
β-glucosidase	J	17	1.05	0.68	0.597	0.649
antibodies	K	16	1.36	1.33	0.739	0.777
casein kinase II	L	16	0.97	0.61	0.454	0.309
ribonuclease	M	15	1.23	1.03	0.551	0.493
thermolysin	N	14	0.97	1.05	0.738	0.636
CDK2 kinase	O	13	1.12	1.14	0.640	0.525
glutamate receptor 2	P	13	1.00	0.84	0.123	0.016
P38 kinase	Q	13	0.59	0.51	0.870	0.896
β-secretase I	R	12	1.43	1.31	0.895	0.687
tRNA-guanine transglycosylase	S	12	0.87	0.95	0.457	0.522
endothiapepsin	T	11	1.36	1.27	0.480	0.210
α-mannosidase 2	U	10	1.83	1.63	0.053	0.103
carboxypeptidase A	V	10	1.77	1.54	0.734	0.685
penicillopepsin	W	10	1.91	1.99	0.078	-0.030
families with 4-9 complexes	X	386	1.54	1.53	0.630	0.632
families with 2-3 complexes	Y	340	1.51	1.52	0.608	0.595
singletons	Z	321	1.67	1.65	0.503	0.507
average			1.33	1.18	0.545	0.479
standard deviation			0.39	0.35	0.228	0.251

timistic, results on leave-cluster-out cross validation might be too pessimistic. Here we want to emphasize that LCOCV is only suitable for estimating the performance of a generic scoring function on a truly new target protein that does not belong to a cluster represented by any of the proteins in the training set, but this constitutes a very rare scenario in real applications because it is uncommon for a target protein not to have high sequence similarity to any other protein in a large and diverse training set. In fact, such type of complexes should never be eliminated from a training set. Instead, the training set composition should reflect as closely as possible the actual complexes on which the scoring function is to be applied in order for the machine-learning models to learn the patterns from the experimental data. LCOCV is consequently inappropriate to evaluate generic scoring functions, as previously argued [201].

6.5.5 Machine-learning scoring functions are significantly more accurate than classical scoring functions

Table 6.9 compares Cyscore, RF::Cyscore, RF::CyscoreVina and RF::CyscoreVinaElem against 21 other scoring functions on PDBbind v2007 core set (N=195). The scoring functions are sorted in the descending order of Rp. RF::CyscoreVinaElem ranks the highest in terms of Rp, Rs and SD. The statistics for the other 21 scoring functions are collected from [9, 50, 52]. It is worth noting that the top four scoring functions are all trained with RF.

Table 6.9: Predictive performance of 25 scoring functions evaluated on PDBbind v2007 core set.

Scoring function	Rp	Rs	SD
RF::CyscoreVinaElem	0.803	0.798	1.42
RF-Score::Elem-v2	0.803	0.797	1.54
SFCscoreRF	0.779	0.788	1.56
RF-Score	0.774	0.762	1.59
ID-Score	0.753	0.779	1.63
RF::CyscoreVina	0.749	0.759	1.58
SVR-Score	0.726	0.739	1.70
RF::Cyscore	0.687	0.694	1.73
Cyscore	0.660	0.687	1.79
X-Score::HMScore	0.644	0.705	1.83
DrugScoreCSD	0.569	0.627	1.96
SYBYL::ChemScore	0.555	0.585	1.98
DS::PLP1	0.545	0.588	2.00
GOLD::ASP	0.534	0.577	2.02
SYBYL::G-Score	0.492	0.536	2.08
DS::LUDI3	0.487	0.478	2.09
DS::LigScore2	0.464	0.507	2.12
GlideScore-XP	0.457	0.435	2.14
DS::PMF	0.445	0.448	2.14
GOLD::ChemScore	0.441	0.452	2.15
SYBYL::D-Score	0.392	0.447	2.19
DS::Jain	0.316	0.346	2.24
GOLD::GoldScore	0.295	0.322	2.29
SYBYL::PMF-Score	0.268	0.273	2.29
SYBYL::F-Score	0.216	0.243	2.35

6.5.6 Substituting RF for MLR and incorporating more features and training samples strongly improves Cyscore

Figure 6.4 compares the predictive performance of Cyscore and RF::CyscoreVinaElem. The top row is for Cyscore and the bottom row is for RF::CyscoreVinaElem. The left column is for PDBbind v2007 benchmark (N=195), with RF::CyscoreVinaElem trained on 1105 complexes. The center column is for PDBbind v2012 benchmark (N=201), with RF::CyscoreVinaElem trained on 2696 complexes. The right column is for PDBbind v2013 round-robin benchmark (N=592), with RF::CyscoreVinaElem trained on 2367 complexes. As seen, RF::CyscoreVinaElem improved Cyscore by -0.28 in RMSE, -0.37 in SD, +0.143 in Rp and +0.111 in Rs on the PDBbind v2007 benchmark, by -0.14 in RMSE, -0.25 in SD, +0.106 in Rp and +0.093 in Rs on the PDBbind v2012 benchmark, and by -0.40 in RMSE, -0.29 in SD, +0.187 in Rp and +0.184 in Rs on the PDBbind v2013 round-robin benchmark.

These results show that RF::CyscoreVinaElem performed consistently better than Cyscore on all the three benchmarks. It is important to note that, in each benchmark, both scoring functions used the same non-overlapping training and test sets. Taken together, these results suggest that one can develop a much more accurate scoring function out of an existing one simply by changing the regression model from MLR to RF and incorporating more structural features and training samples.

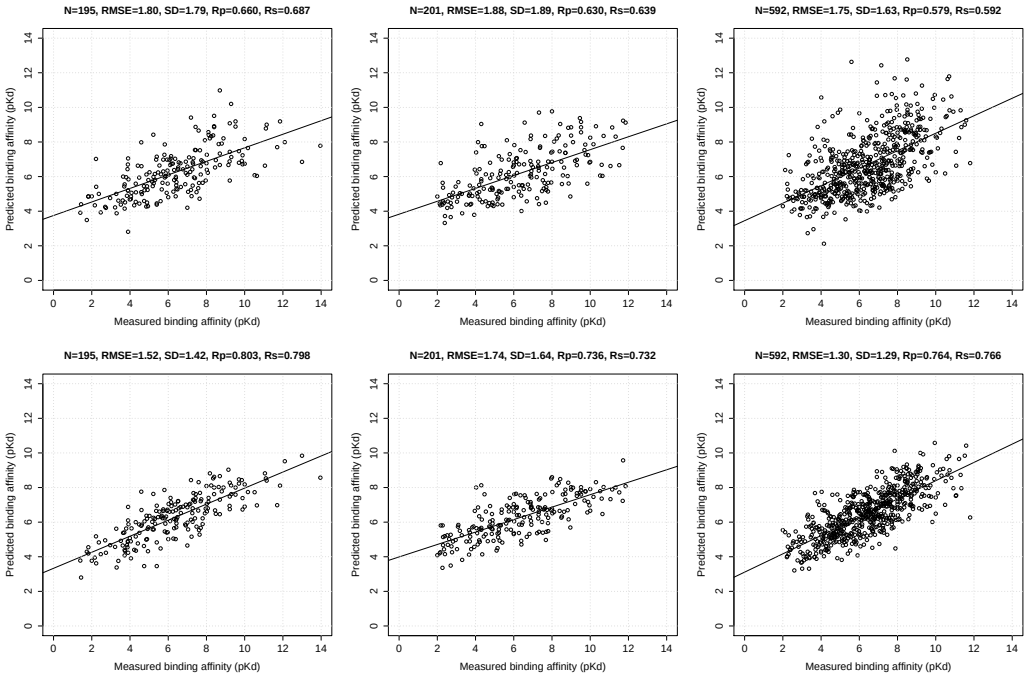


Figure 6.4: Correlation plots of predicted binding affinities against measured ones.

6.5.7 Sensitivity analysis of the RF model can estimate feature importance

RF-based scoring functions, unlike classical scoring functions, can barely be explicitly expressed as a mathematical equation like equation (6.1). Therefore it is useful to employ the variable importance tool of RF to estimate the importance of each feature by randomly permuting its training values, and the feature leading to the largest variation in the predicted binding affinity on the OOB data can be regarded as the most important for a particular training set. Figure 6.5 plots the percentage of increase in mean square error (%IncMSE) observed when each of the 4 Cyscore features used to train RF was noised up. The four features are hydrophobic free energy (Hydrophobic), van der Waals interaction energy (Vdw), hydrogen bond interaction energy (HBond) and ligand's conformational entropy (Ent). The %IncMSE value of a particular feature was computed as the percentage of increase in mean square error observed in OOB prediction when that features was randomly permuted. All the 4 features turned out to be important (%IncMSE>20), with van der Waals interaction energy (Vdw) and hydrophobic free energy (Hydrophobic) being relatively more important (%IncMSE>40). Correctly estimating variable importance can assist in feature selection and in understanding ligand binding.

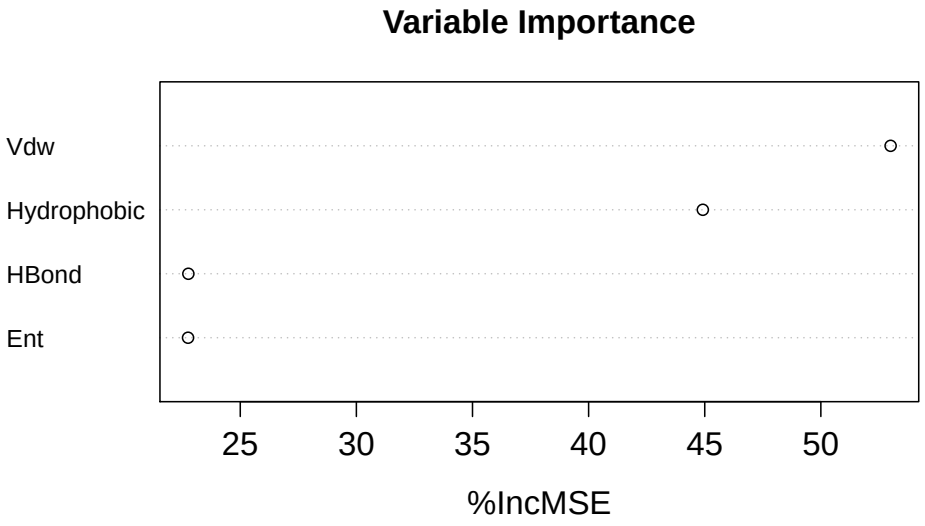


Figure 6.5: RF::Cyscore feature importance estimated on internal OOB data of the 1105 complexes from PDBbind v2007 refined set.

6.6 Conclusions

We have demonstrated that the multiple linear regression (MLR) model used in many scoring functions like Cyscore does not improve its performance in the presence of abundant training samples. This is a especially significant drawback for MLR-based scoring functions because they cannot benefit from the increasing availability of future experimental data. On the other hand, RF-based scoring functions can comprehensively capture the non-linear nature in the data and thus assimilate data significantly better than MLR-based scoring functions. Most importantly, feeding more training samples to RF can increase its predictive performance. Under this circumstance, improvements with dataset size can only be gained with the appropriate re-

gression model. Changing the regression model of Cyscore from MLR to RF and expanding the feature set and the sample set can significantly increase the predictive accuracy. The performance gap between MLR-based and RF-based scoring functions will be further widened by the future availability of more and more X-ray crystal structures.

Classical empirical scoring functions typically rely on complicated energetic contributions that must be carefully devised from intermolecular interactions, whereas RF-based scoring functions can also effectively exploit features as simple as occurrence count of intermolecular contacts. It has also been shown in a previous study that functional group contributions in protein-ligand binding are non-additive. This means novel features can be difficult to be incorporated into an existing MLR model. In this study we have shown that using more structural features appropriately can also substantially boost the predictive accuracy of RF, as can be seen in the comparison between RF::CyscoreVinaElem and RF::Cyscore. This further stresses the importance of substituting RF for MLR in scoring function development.

6.7 Future works

The PHOENIX [202] scoring function uses calorimetry data to decompose the change of binding free energy into change of enthalpy and change of entropy. It is an empirical scoring function that uses shape and volume descriptors to independently model

enthalpic and entropic contributions. It is interesting to see how the predictive performance changes if RF is utilized for regression.

□ **End of chapter.**

Chapter 7

RF-Score-v3: binding affinity prediction

There is a growing body of evidence showing that machine learning regression results in much more accurate structure-based prediction of protein-ligand binding affinity. Such prediction is a requirement for docking methods in that it is the basis for discriminating between active and inactive molecules or optimising the potency of a ligand against a target. However, despite their proven advantages, machine-learning scoring functions are still not widely applied. This seems to be due to insufficient understanding of their properties and the lack of user-friendly software implementing them.

Here we present a study where the accuracy of AutoDock Vina, arguably the most commonly-used docking software, is strongly improved by following a machine learning approach. We also analyse the factors that are responsible for this improvement and their generality. Most importantly, with the help of a proposed benchmark, we demonstrate that this im-

provement will be larger as more data becomes available for training Random Forest models, as regression models implying additive functional forms do not improve with more training data. We discuss how the latter opens the door to new opportunities in scoring function development. In order to facilitate the translation of this advance to enhance structure-based molecular design, we provide software to directly re-score Vina-generated poses and thus strongly improve their predicted binding affinity. The rescoring software is freely available at http://istar.cse.cuhk.edu.hk/rf-score-3.tgz.

This was a collaborative project with Pedro J. Ballester from Cancer Research Center of Marseille, Marseille, France. It was published in *Proceedings of the 11th International Meeting on Computational Intelligence Methods for Bioinformatics and Biostatistics (CIBB)* on 26 June 2014 [17], and in *Molecular Informatics* on 12 February 2015 [16].

7.1 Background

Molecular docking is a key computational technique in structural bioinformatics and structure-based molecular design. Docking predicts the preferred conformations and binding strength of a ligand molecule, typically a small organic molecule, as bound to a protein pocket. Such prediction is necessary to discriminate between molecules that bind and those that do not bind to a target of interest (i.e. those molecules with high affinity for the target and those with an affinity so low that does not

permit stable binding). Docking is not only useful to anticipate whether a ligand binds tightly to a target, but also to understand how it binds. The latter can be helpful to improve the potency and selectivity of binding. Docking is often utilized to identify a molecule that binds tightly to the target, so that a small concentration of the molecule is sufficient to modulate its biochemical function.

Docking applications include, but are not limited to, structure-based virtual screening [189, 203, 204], drug lead optimisation [191], polypharmacology prediction [205, 206], drug repositioning [190], binding pocket prediction [192, 207], human variation prediction [208], protein function prediction [193] and target druggability assessment [209].

Operationally, docking has two stages: predicting the position, orientation and conformation of a molecule when docked to the target's binding site (pose generation), and predicting how strongly the docked pose of such putative ligand binds to the target (scoring). The single most important limitation of docking is the traditionally low accuracy of the scoring functions that predict the strength of binding.

Classical scoring functions assume a predetermined theory-inspired functional form for the relationship between the numerical features that describe the complex and its predicted binding affinity. There are three types of classical scoring functions: force-field [210–212], empirical [213–216] and knowledge-based [217–220]. Each type follows a different philosophical approach to scoring function development, as explained elsewhere, e.g.

[221]. However, it is important to note that these three types are mathematically equivalent in that a predetermined functional form is imposed to vertebrate the scoring function. In almost all classical scoring functions, the assumed functional form is additive. Furthermore, like empirical scoring functions, most modern force-field and knowledge-based scoring functions weight their constituent terms by fitting experimental binding data [221]. Figure 7.1 illustrates the mathematical equivalence of these three popular classical scoring functions as sums of data-weighted energetic contributions to binding. K_j is the total number of protein atoms of type j and L_i is the total number of ligand atoms of type i in the considered complex. An atom type is defined according to atomic number and in some cases also using the calculated protonation state (e.g. hydrogen bond donors). The additive terms may additionally impose interatomic distance and angle constraints between neighbouring atoms of the considered types (e.g. hydrogen bonding terms).

Recently, machine-learning scoring functions have been shown [10] to be much more accurate than classical scoring functions at binding affinity prediction. This improvement is due to two factors. The first is the circumvention of the assumed functional form of classical scoring functions, which is learnt instead in an entirely data-driven manner in machine-learning scoring functions. This was to be expected as it is well known that individual free energy terms may not be additive [194, 222]. Second, research on classical scoring functions has focused on increasingly detailed modelling of contributions to binding, but

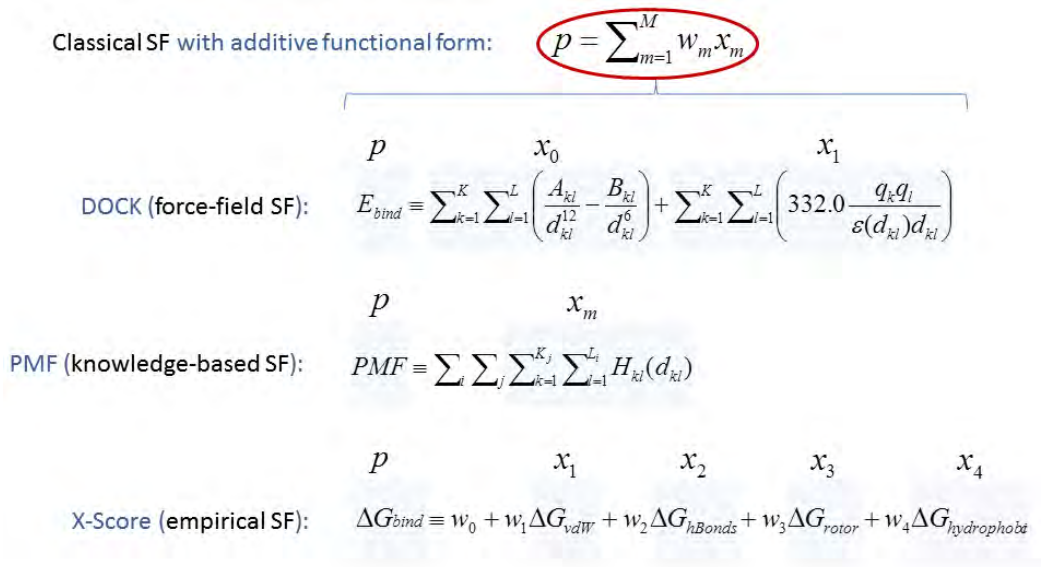


Figure 7.1: Mathematical equivalence of classical scoring functions as sums of data-weighted energetic contributions to binding.

it has now been established that a more precise description of protein-ligand complexes does not generally lead to more accurate prediction of binding affinity [50]. This study has resulted in an expected set of structural descriptors leading to improved performance when allied with a sufficiently flexible regression model.

Machine-learning scoring functions have been misclassified as knowledge-based scoring functions [14, 223] or empirical scoring functions [55], but these are fundamentally different from either type because of not imposing a fixed functional form on the relationship between structural and binding data. This distinction between machine-learning and classical scoring functions has important consequences in practice, as it will be analysed.

Despite being a recent development, there are already suc-

cessful prospective applications of machine-learning scoring functions. RF-Score [10] has recently been used [140] to discover a large number of innovative binders of antibacterial DHQase2 targets. To facilitate its use, RF-Score has been incorporated into a large-scale docking web server for prospective virtual screening, available at http://istar.cse.cuhk.edu.hk/idock [9]. On the other hand, a machine-learning scoring function called MD-SVR has been generated and applied [224] to guiding the optimisation of known Akt1 kinase inhibitors. The derivatives highlighted by MD-SVR were synthesised and all of them exhibited moderate to good inhibitory activities.

7.2 Motivation

The innovative development of RF-Score has however raised a few concerns. For example, the use of oversimplified features in the original version of RF-Score has been pointed out as problematic [225], although no empirical evidence was provided in support of this claim and this version actually achieved high hit rates in prospective virtual screening [140]. The superior performance of RF-Score was highlighted by [198], which nevertheless attributed it to the characteristics of the most widely-used benchmark. This was subsequently demonstrated not to be the case [201]. Still, there seem to be some concerns that the applicability domain of machine-learning scoring functions would be somehow more restrictive than that of classical scoring functions. Lastly, [14] claimed that the application of machine-

learning scoring functions is limited by their tendency to overfit training data and their alleged difficulty in providing an immediate physical interpretation of the results. We were motivated by the above concerns and thus aimed to provide responses supported by solid numerical experiments.

7.3 Objective

In this chapter, we show that one can construct a machine-learning scoring function from a classical scoring function to have the same applicability domain and interpretability capabilities while greatly improving its ability to predict binding affinity. This will be shown with AutoDock Vina [8] as the classical scoring function because it is arguably very popular. Furthermore, we will also address the remaining criticisms supported by numerical experiments. Besides, the growing importance of machine-learning scoring functions will be demonstrated in the context of a purposely-built new benchmark by analysing how their performances improve with the increase of structural and binding data used for training. Finally, we will provide free software to rescore protein-ligand complexes, either crystal or docked.

7.4 Methods and materials

This section introduces four scoring functions building upon AutoDock Vina, two benchmarks to evaluate performance of

these scoring functions and the performance metrics that will be used to this end.

7.4.1 Model 1 - AutoDock Vina

The AutoDock series [8, 32, 69] is the most cited docking software by the research community, with over 8,000 citations to date between these three publications, according to Google Scholar. As a completely new counterpart of AutoDock 4 [32], AutoDock Vina [8] substantially improved the average accuracy of the binding mode predictions, while running two orders of magnitude faster with multithreading. Vina was an exciting development, not only because of its remarkable pose generation performance in terms of both effectiveness and efficiency, but also because it is an open source tool and is among the most accurate classical scoring functions for binding affinity prediction.

Like all classical scoring functions, Vina assumes a predetermined functional form. In this case, Vina's score for the kth conformer, e_k, is calculated as:

$$e_k = \frac{e_{k,inter} + e_{k,intra} - e_{1,intra}}{1 + w_6 N_{rot}} \tag{7.1}$$

Now because studies on binding affinity prediction are benchmarked on co-crystallised ligands to avoid confounding factors, there is only one conformer per molecule ($k = 1$) and thus the intramolecular contribution cancels out giving:

$$e_1 = \frac{e_{1,inter}}{1 + w_6 N_{rot}} \tag{7.2}$$

where

$$\begin{aligned} e_{1,inter} &= w_1 \cdot Gauss1_1 \\ &+ w_2 \cdot Gauss2_1 \\ &+ w_3 \cdot Repulsion_1 \\ &+ w_4 \cdot Hydrophobic_1 \\ &+ w_5 \cdot HBonding_1 \end{aligned} \tag{7.3}$$

$$\mathbf{w} = (-0.035579, -0.005156, 0.840245, -0.035069, -0.587439, 0.05846) \tag{7.4}$$

e_1 is the predicted free energy of binding reported by the Vina software when scoring the structure of a protein-ligand complex. The values for the six weights were found by Ordinary Least Squares (OLS) using a nonlinear optimisation algorithm as it has been the case in related force-field scoring functions [226], although this process was not detailed in the original publication [8]. The training data was PDBbind v2007 refined set (N=1300). N_{rot} is the number of rotatable bonds. Unlike other classical scoring functions, Vina is not exactly a sum of energy terms because $w_6 \neq 0$, although it is quasi-linear since $1+w_6 N_{rot}$ takes values close to 1 for most protein-ligand complexes. As usual, e.g. [9], the predicted free energy of binding in kcal/mol units is converted into pKd with $pK_d = -0.73349480509 e_1$ in order to compare to binding affinities (pK_d or pK_i). Expressions and

further details for the five $e_{k,inter}$ terms can be found in [8, 9].

7.4.2 Model 2 - MLR::Vina

This is a multiple linear regression (MLR) model using the six unweighted Vina terms as features. The use of MLR as the regression model implies an additive functional form and hence MLR::Vina is a classical scoring function. It adopts the philosophy of empirical scoring functions.

In order to make the problem amenable to MLR, we made a grid search on the w_6 weight and thereafter ran MLR on the remaining five weights. Specifically, we sampled 101 values for w_6 from 0 to 1 with a step size of 0.01. Interesting we found that the w_6 values of the best models were always between 0.005 and 0.020. Then we again sampled 16 values for w_6 in this range with step size 0.001, and used the best of them in terms of the lowest RMSE (Root Mean Square Error) on the training set.

7.4.3 Model 3 - RF::Vina

While Vina's ability to predict binding affinity is among the best provided by classical scoring functions, it is still limited by the assumption of a functional form. To investigate the impact of this modelling assumption, we used Random Forest (RF) [139] to implicitly learn the functional form from the data. Other machine learning techniques can of course be applied to this problem, e.g. SVR (Support Vector Regression) [149], although this is out of the scope of the study.

A RF is an ensemble of many different decision trees randomly generated from the same training data [139]. RF trains its constituent trees using the CART algorithm [141]. Instead of using all features, RF selects the best data split at each node of the tree from a typically small number (mtry) of randomly chosen features. In regression problems, the RF prediction is given by arithmetic mean of all the individual tree predictions in the forest.

Here we built a RF model with the six Vina features using the default number of trees (500) and values of the mtry control parameter from 1 to all 6 features. The selected model was that with the mtry value providing the lowest RMSE on a subset of training data known as the OOB (Out of Bag) data. This process was repeated ten times with ten different random seeds because RF is stochastic. The predictive performance was reported for the RF with the best seed that led to the lowest RMSE on the test set. Further details on RF model building in this context can be found in [10].

7.4.4 Model 4 - RF::VinaElem

This is the model described in the previous subsection with an expanded set of 42 features once the 36 RF-Score features are added to the six Vina features. For a given random seed, a RF for each mtry value from 1 to 42 was built and that with the lowest RMSE on OOB data was selected as the scoring function. Like in the training process of model 3, the same ten seeds were

used, and the predictive performance was reported for the RF with the best seed that resulted in the lowest RMSE on the test set.

To calculate RF-Score features, atom types were selected so as to generate features that are as dense as possible, while considering all the heavy atoms commonly observed in PDB complexes (C, N, O, F, P, S, Cl, Br, I). As the number of protein-ligand contacts is constant for a particular complex, the more atom types are considered, the sparser the resulting features will be. Therefore, we selected a minimal set of atom types by considering atomic number only. Furthermore, a smaller set of interaction features has the additional advantage of leading to computationally faster scoring functions.

RF-Score features are defined as the occurrence count of intermolecular contacts between elemental atom types i and j, as shown in equations (7.5) and (7.6), where d_{kl} is the Euclidean distance between the kth protein atom of type j and the lth ligand atom of type i calculated from a structure; K_j is the total number of protein atoms of type j ($\#\{j\} = 9$) and L_i is the total number of ligand atoms of type i ($\#\{i\} = 4$) in the considered complex; $\mathcal{H}$ is the Heaviside step function that counts contacts within a d_{cutoff} neighbourhood. For example, $x_{7,8}$ is the number of occurrences of protein oxygen atoms hypothetically interacting with ligand nitrogen atoms within a chosen neighbourhood. Full details on RF-Score features are available at [10, 149].

$$x_{ij} = \sum_{k=1}^{K_j} \sum_{l=1}^{L_i} \mathcal{H}(d_{cutoff} - d_{kl}) \tag{7.5}$$

$$\mathbf{x} = \{x_{ij}\} \in N^{36} \tag{7.6}$$

7.4.5 The PDBbind benchmark

Using predefined training and test sets, where other scoring functions had previously been tested, has the advantage of minimising the risk of using a benchmark complementary to the presented scoring function. There are many examples of benchmarks to validate generic scoring functions [227–230].

Here we use the PDBbind benchmark [148], arguably the most widely used for binding affinity prediction of diverse complexes. This benchmark is based on the 2007 version of the PDBbind database, which contains a particularly diverse collection of 1300 protein-ligand complexes with their corresponding binding affinities, assembled through a systematic mining of the entire PDB (Protein Data Bank) [22, 144].

The PDBbind benchmark essentially consists of testing the predictions of scoring functions on the 2007 core set, which comprises 195 diverse complexes with measured binding affinities spanning more than 12 orders of magnitude, while training in the remaining 1105 complexes in the refined set. In this way, a set of protein-ligand complexes with measured binding affinity can be processed to give two non-overlapping data sets, where each complex is represented by its feature vector $\mathbf{x}_i$ and its bind-

ing affinity y_i, which includes both pKd and pKi measurements, henceforth referred to as just pKd for simplicity:

$$D_{train} = \{y_i, \mathbf{x}_i\}_{i=1}^{1105} \tag{7.7}$$

$$D_{test} = \{y_i, \mathbf{x}_i\}_{i=1106}^{1300} \tag{7.8}$$

$$y = pK_d = -\log_{10} K_d \tag{7.9}$$

This benchmark has the advantage of permitting a direct comparison against the performance of 16 classical scoring functions that had previously been benchmarked on the same test set [148]. These 16 classical scoring functions include five scoring functions in the Discovery Studio software version 2.0 from Accelrys: LigScore [216], PLP [217], PMF [218], Jain [231] and LUDI [213], five scoring functions (D-Score, PMF-Score, G-Score, Chem-Score, and F-Score) in the SYBYL software version 7.2 from Tripos, GlideScore [215] in the Schrödinger software version 8.0 from Schrödinger, three scoring functions in the GOLD software version 3.2 from the CCDC: GoldScore [232], ChemScore [214] and ASP [219], and two standalone scoring functions released by academic groups: DrugScore [220, 233] and X-Score version 1.2 [150]. Several of these scoring functions had different versions or multiple options, including LigScore (LigScore1 and LigScore2), PLP (PLP1 and PLP2), and LUDI (LUDI1, LUDI2, and LUDI3) in Discovery Studio; GlideScore (GlideScore-SP and GlideScore-XP) in the Schrödinger software;

DrugScore (Drug-ScorePDB and DrugScoreCSD); and X-Score (HPScore, HMScore, and HSScore). For simplicity, the authors who tested all these scoring functions on the PDBbind benchmark [148] only reported the best performance of the version/options of each scoring function. Furthermore, we added to the comparison other classical scoring functions that have subsequently been tested on this benchmark: IMP::RankScore [234] in [50], HYDE2.0::HbondsHydrophobic [235], PHOENIX [202], Cyscore [14], HotLig [236] and DSX^{CSD} [237]. Many machine-learning scoring functions have also been tested on this benchmark, which are however not relevant for the goals of this study and hence not included.

7.4.6 The 2013 blind benchmark

We propose a new benchmark mimicking a blind test to provide a more realistic validation than the PDBbind benchmark, where higher performance is to be expected due to the protocol that generates this partition [201]. The new test set comprises all the structures in the 2013 release of the PDBbind refined set that were not already in the 2012 release, i.e. the new protein-ligand complexes added in 2013, whereas the 2012 refined set is used for training. This is hence conducted as a blind test in that only data available until a certain year is used to build the scoring function that predicts the binding affinities of 2013 complexes as if these had not been measured yet. The PDBbind v2013 refined set (N=2959) and the PDBbind v2012 refined set

(N=2897) have 2576 complexes in common. In the 2013 refined set, the 3rv4 protein consists of two Cs atoms which Vina does not support, so this complex was discarded. Eventually the test set has 2959-2576-1=382 complexes.

In addition, we define three more training sets to account for the structural and binding data available in the public domain at three previous times. These three previous PDBbind releases were selected so that there is approximately the same number of complexes between consecutive releases. In this way, one can use these four partitions sharing the same test set to study how scoring function performance varies with training data set size. We selected PDBbind v2002 (N=800), v2007 (N=1300) and v2010 (N=2061) refined sets. In the v2002 refined set, the 1tha protein failed PDB-to-PDBQT conversion, and the 1lkk, 1mfi, 7std, 1cet, 2std, 1els, 1c3x ligands failed PDB-to-PDBQT conversion. Eventually this training set has 800-8=792 complexes. In the v2010 refined set, the 2bo4 protein failed PDB-to-PDBQT conversion, and the 1xr8 ligand is far away from its protein. Eventually this training set has 2061-2=2059 complexes.

Table 7.1 shows the data partitions. Partition 1 is the PDBbind benchmark. Refined02 means the 2002 release of the PDBbind refined set, whereas refined07\core07 means the complexes left in refined07 after removing those in core07. By construction of each partition, there are no complexes in common between any training set and test set pair. In other words, there is no overlap between both sets and hence each test set complex is new data not seen in model training. Note that only models 2,

Table 7.1: Data set partitions of the PDBbind and the 2013 blind benchmarks.

Partition	Training set	N	Test set	N
1	refined07\core07	1105	core07	195
2	refined02	792	refined13\refined12	382
3	refined07	1300	refined13\refined12	382
4	refined10	2059	refined13\refined12	382
5	refined12	2897	refined13\refined12	382

3 and 4 were re-trained on each of the five training sets, whereas model 1, AutoDock Vina, was used out of the box without re-training.

One may argue that an additional partition is necessary, where all training complexes similar to any of the test complexes in some way are removed. Nevertheless, in practice, a test set will contain really few complexes of this type. Therefore, we would be depriving the scoring functions of the most relevant training data, which would actually be available, without good reason, leading to unrealistically low performance for all scoring functions. This point has already been discussed elsewhere [15, 201].

7.4.7 Performance measures

As usual [148], performance will be measured by the Standard Deviation (SD), Root Mean Square Error (RMSE), Pearson correlation (Rp) and Spearman rank-correlation (Rs) between predicted and measured binding affinity. SD is included to permit comparison to previously-tested scoring functions on this benchmark. RMSE, on the other hand, reflects the ability of the scor-

ing function to report an accurate binding affinity value. Rs shows how well it can rank bound ligands according to binding strength. Rp simply shows how linear the correlation is and thus it is a less relevant indicator of the quality of the prediction. The mathematical expressions of these four metrics can be found in [15].

7.5 Results and discussion

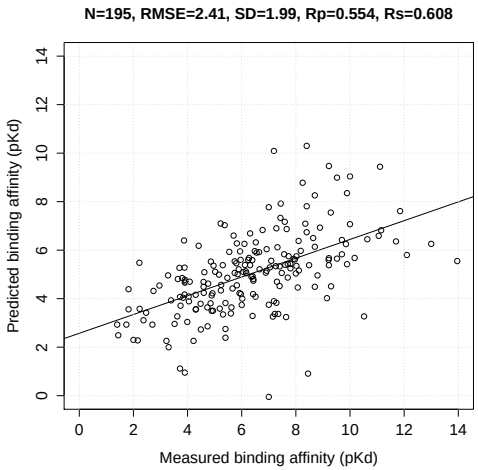
Figures 7.2 and 7.3 show the predictive performance of the four models on the PDBbind benchmark (partition 1 in Table 7.1) and the 2013 blind benchmark (partition 5 in Table 7.1), respectively.

7.5.1 MLR is better at calibrating the additive functional form of Vina's scoring function

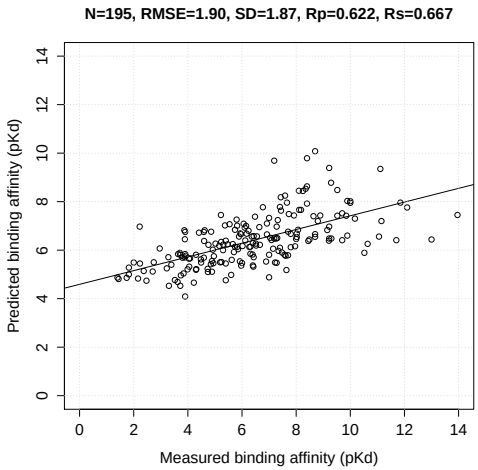
Figures 7.2 and 7.3 show that MLR::Vina provided a test set performance with significantly lower error and higher correlation than Vina on both benchmarks. This means that MLR is more suitable to calibrate Vina's scoring function than the originally used nonlinear optimisation algorithm.

7.5.2 Vina's assumed functional form is detrimental for its performance

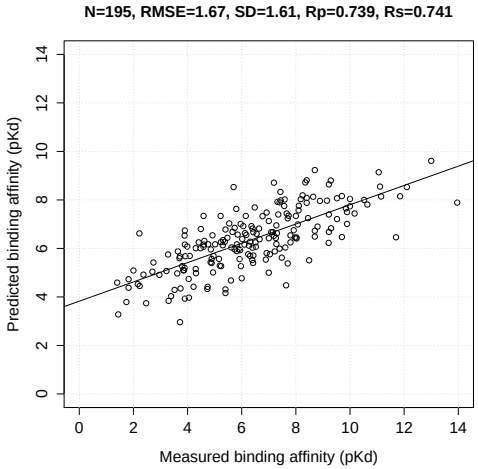
Both the linear (model 2) and nonlinear (model 1) optimisation approaches to training Vina assume a quasi-additive func-



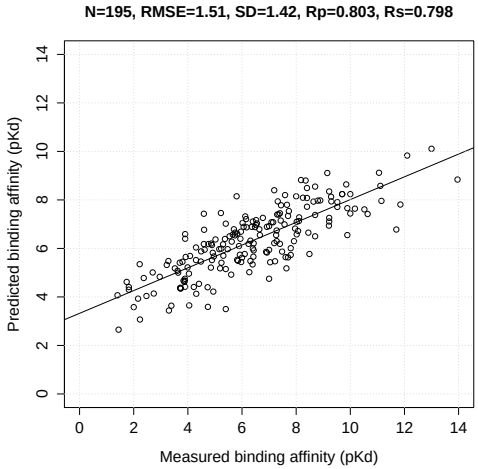
(a) AutoDock Vina.



(b) MLR::Vina.

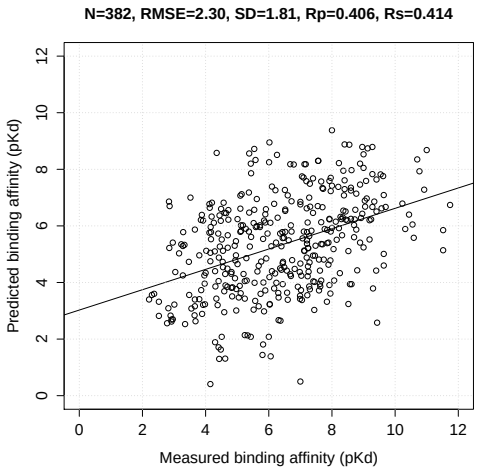


(c) RF::Vina.

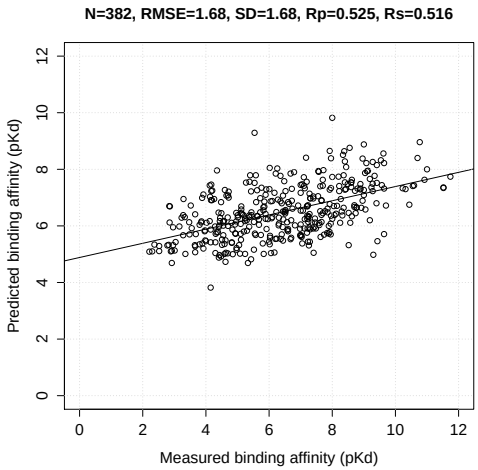


(d) RF::VinaElem.

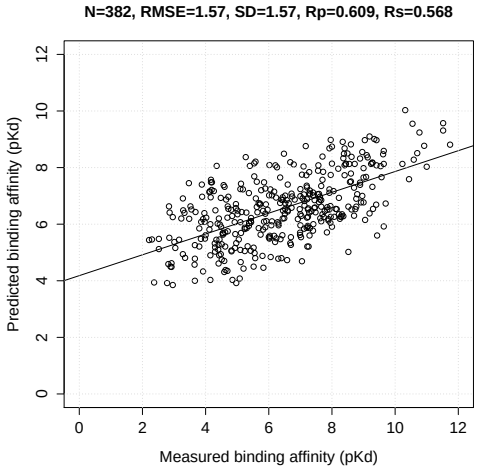
Figure 7.2: Performance on the 195 test set complexes in the PDBbind benchmark.



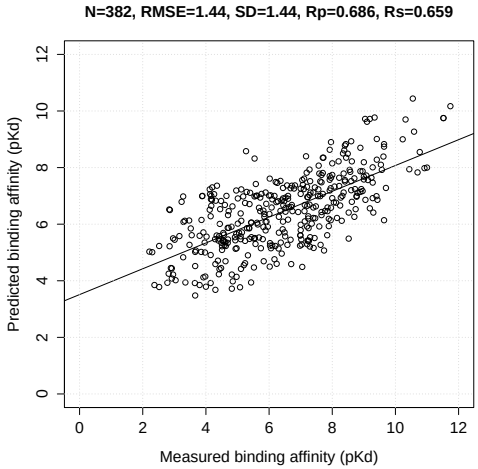
(a) AutoDock Vina.



(b) MLR::Vina.



(c) RF::Vina.



(d) RF::VinaElem.

Figure 7.3: Performance on the 382 test set complexes in the 2013 blind benchmark.

tional form. By looking at Figures 7.2 and 7.3, it is clear that model 3 performed much better than models 1 and 2. Note that model 3 uses exactly the same features as the other two models, and it uses exactly the same training data as model 2. The only difference between these models is that model 3 implicitly constructs the functional form from the data using RF for regression, whereas the other two Vina models assume a priori form for how the features are combined to form the scoring function. Therefore, these results demonstrate that this performance improvement is entirely due to the avoidance of this commonly-used modelling assumption.

7.5.3 Incorporating ligand properties increases performance further

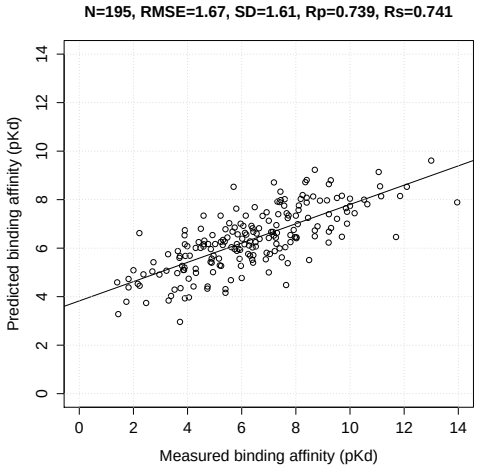
The N_{rot} feature, unlike the remaining five fixed Vina features, which encode properties of the protein-ligand complex, is exclusively a property of the ligand. It is the number of rotatable bonds, effectively an estimation of the flexibility of the ligand. When model 3 was run with five features (all but N_{rot}), test set error increased from a best RMSE of 1.67 to 1.74, and Rs correlation dropped from 0.741 to 0.706 on the PDBbind benchmark, as shown in Figure 7.4. Similar performance degradation was also observed on the PDBbind v2013 blind benchmark. This result shows that it is advantageous to add N_{rot} as a model feature. More broadly, this suggests that incorporating ligand properties into the model, such as those that are routinely used

in ligand-based QSAR models, may enhance performance further. Likewise, features encoding protein properties could also extend the capabilities of generic scoring functions.

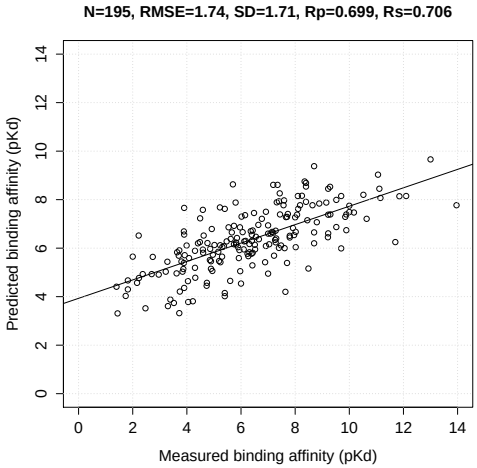
7.5.4 The impact of overfitting on RF performance

It has been recently claimed [14] that the tendency of machine-learning scoring functions to overfit training data is a weak point limiting their application. This is a surprisingly common misconception, whereby a less overfitted model is regarded as necessarily better than a more overfitted model. The latter implicitly assumes that the impact of overfitting will be the same for different classes of regression models. Nevertheless, some models, e.g. RF, are robust to overfitting in the sense that this has a low impact on its generalization ability.

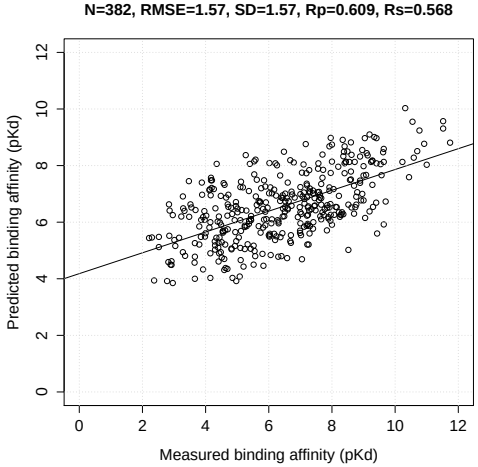
To quantify the impact of overfitting, we trained MLR::Vina and RF::Vina on the same 1105 complexes and use them to predict the binding affinity of the 195 test set complexes in partition 1 in Table 7.1. MLR::Vina performed better on the training set than on the test set (SD=1.73 vs SD=1.87, respectively), which suggests that classical scoring functions only have a small degree of overfitting. In contrast, RF::Vina's performance was much better on the training set than on the test set (SD=0.60 vs SD=1.61, respectively), which evidences that RF significantly overfits training data. However, the large performance gain of RF::Vina over MLR::Vina on the test set (SD=1.61 vs SD=1.87, respectively) makes clear that RF::Vina is substantially more



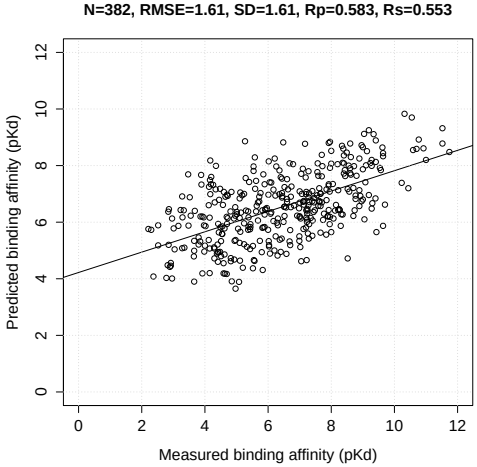
(a) RF::Vina with N_{rot} on v2007.



(b) RF::Vina without N_{rot} on v2007.



(c) RF::Vina with N_{rot} on v2013.



(d) RF::Vina without N_{rot} on v2013.

Figure 7.4: Performance of RF::Vina including and excluding the N_{rot} feature on the PDBbind v2007 and v2013 blind benchmarks.

accurate than MLR::Vina regardless of overfitting. Therefore, overfitting cannot be used to anticipate the relative performance of two different models on a test set and hence it is necessary to measure the true impact of overfitting on the compared models.

Next, we used partition 5 from Table 7.1 to provide further evidence that overfitting is not a weak point limiting the application of RF-based scoring functions. The training set is composed of 2897 complexes and test set is composed of 382 complexes, which as that from partition 1 also contains complexes that are very different among themselves. The test set was subdivided into five equally-sized new sets and the operation was repeated ten times to provide 50 different smaller test sets with 76 or 77 complexes each. Thereafter, we evaluated MLR::Vina and RF::VinaElem on each test set and plotted the resulting SD errors in Figure 7.5. If overfitting was a problem with RF, a highly variable difference in performance between both models would have been observed, with the less overfitted model being often better than the more overfitted model. By contrast, not once MLR::Vina outperformed RF::VinaElem.

7.5.5 Improvement of AutoDock Vina using RF

RF::VinaElem is the product of two improvements over Vina: using RF on the six Vina features to circumvent the need of a functional form (model 3) and combining the latter with an expanded set of 42 features incorporating the 36 RF-Score features (model 4). Figure 7.2 clearly shows that RF::VinaElem greatly

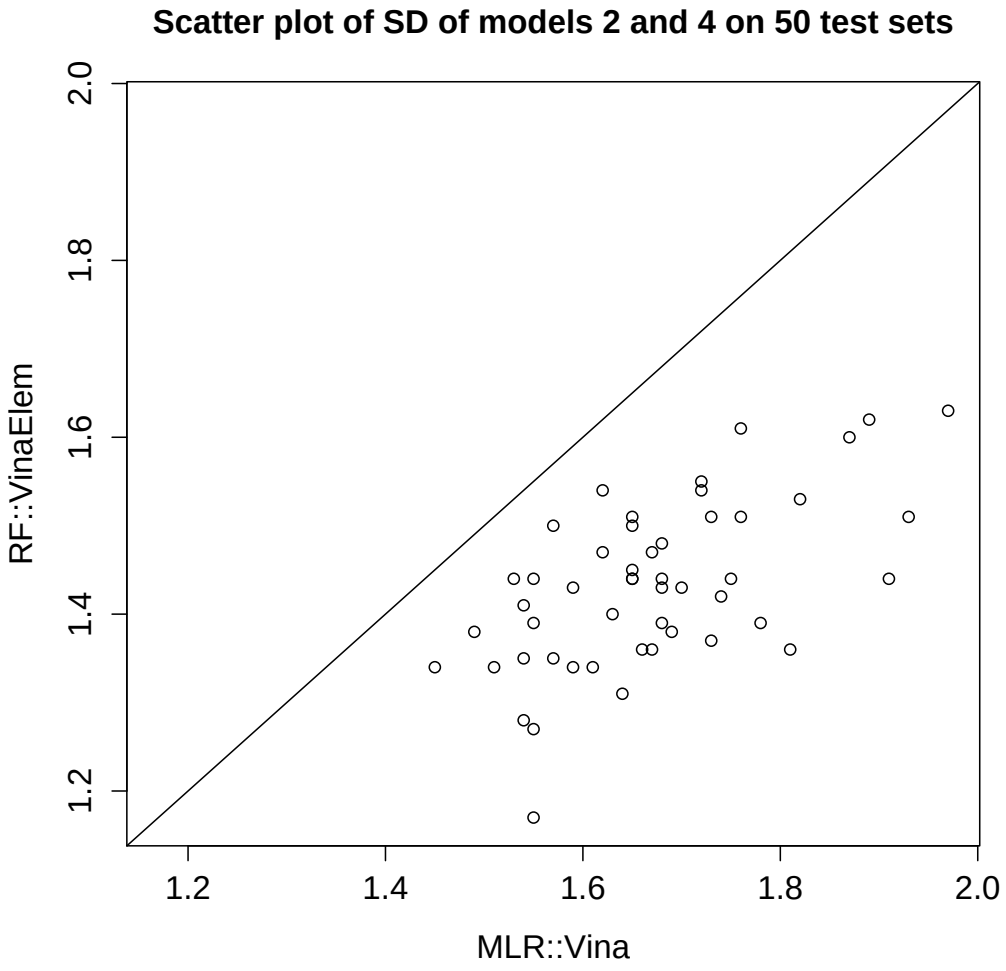


Figure 7.5: MLR::Vina and RF::VinaElem, both trained on the same 2897 complexes, compared on 50 test sets by their SD errors.

improved Vina by -0.90 in RMSE, -0.57 in SD, +0.249 in Rp and +0.190 in Rs. In comparison, the NHA baseline obtained RMSE=2.15, SD=2.15, Rp=0.431, Rs=0.510, and the MWT baseline obtained RMSE=2.16, SD=2.17, Rp=0.418, Rs=0.496.

Figure 7.3 also shows the same conclusion, with RF::VinaElem also achieving a substantial improvement over Vina of -0.86 in RMSE, -0.37 in SD, +0.280 in Rp and +0.245 in Rs. While there is a significant decrease in Rp and Rs for both scoring functions with respect to the PDBbind benchmark (compare Figures 7.2 and 7.3), the relative performance of both scoring functions is similar on both benchmarks as it was anticipated [201]. In comparison, the NHA baseline obtained RMSE=1.90, SD=1.89, Rp=0.295, Rs=0.363, and the MWT baseline obtained RMSE=1.90, SD=1.90, Rp=0.269, Rs=0.330.

On the other hand, it is well known that there is a considerable correlation between ligand size and binding affinity. Thus, simple models such as the MWT baseline have been used to put in perspective the performance of scoring functions. Table 7.2 shows that many classical scoring functions are close to or even below this simple baseline. By contrast, RF::VinaElem obtained an Rp of 0.803 on the PDBbind benchmark, which almost doubles that of MWT (0.418). On the 2013 blind benchmark, RF::VinaElem obtained a Rp of 0.686 whereas MWT's was just 0.269. This is not surprising as RF::VinaElem has been designed to learn the relationship between intermolecular interactions and binding affinity from structural data, not ligand properties, which were solely used in the NHA and MWT

baselines.

Overall, it is remarkable that RF::VinaElem achieved an error of just 1.44 pKd units, or 1.96 kcal/mol equivalently, in a blind benchmark comprising such a diverse independent test set (Figure 7.3). In comparison, Vina's RMSE=2.30 translates to 3.14 kcal/mol. It would be interesting to see how well other classical scoring functions perform on this benchmark, such as those in Table 7.2 and even theoretically more accurate techniques such as free energy calculations [238].

7.5.6 Machine-learning scoring functions are remarkably more accurate than empirical scoring functions

Table 7.2 compares the performance of RF::VinaElem against that of 21 classical scoring functions and two naive baselines, NHA and MWT. NHA is simply a linear regression model with the number of heavy atoms of the ligand as only variable. MWT uses the molecular weight of the ligand as the variable instead. Some other classical scoring functions only reported some of the performance measures, and hence cannot be included in the full comparison in Table 7.2. These are DSX^{CSD}::All [237] with Rp=0.609, and HotLig [236] with Rs=0.609.

While it has been claimed [14] that empirical scoring functions have similar accuracy than machine-learning scoring functions, the results in Table 7.2 clearly demonstrate that this is not the case. Indeed, the improvement introduced by RF::VinaElem

Table 7.2: Performance of 22 scoring functions and 2 naive baselines on the PDBbind benchmark.

Scoring function	Rp	Rs	SD
RF::VinaElem	0.803	0.798	1.42
Cyscore	0.660	0.687	1.79
X-Score::HMScore	0.644	0.705	1.83
HYDE2.0::HbondsHydrophobic	0.620	0.669	1.89
DrugScoreCSD	0.569	0.627	1.96
SYBYL::ChemScore	0.555	0.585	1.98
AutoDock Vina	0.554	0.608	1.99
DS::PLP1	0.545	0.588	2.00
GOLD::ASP	0.534	0.577	2.02
SYBYL::G-Score	0.492	0.536	2.08
DS::LUDI3	0.487	0.478	2.09
DS::LigScore2	0.464	0.507	2.12
GlideScore-XP	0.457	0.435	2.14
DS::PMF	0.445	0.448	2.14
GOLD::ChemScore	0.441	0.452	2.15
NHA baseline	0.431	0.517	2.15
PHOENIX	0.616	0.644	2.16
MWT baseline	0.418	0.496	2.17
SYBYL::D-Score	0.392	0.447	2.19
DS::Jain	0.316	0.346	2.24
IMP::RankScore	0.322	0.348	2.25
GOLD::GoldScore	0.295	0.322	2.29
SYBYL::PMF-Score	0.268	0.273	2.29
SYBYL::F-Score	0.216	0.243	2.35

over the best classical scoring functions is much larger than what has recently been considered acceptable for publication. For instance, Cyscore was shown to outperform the best classical scoring function, X-Score::HMScore, by only 0.04 in SD error [14]. By contrast, RF::VinaElem improves Cyscore and X-Score::HMScore by 0.37 and 0.41 in this study, respectively. These are 9 and 10 times larger SD error reduction.

7.5.7 Machine-learning scoring functions assimilate data better than empirical scoring functions

This subsection analyses the reasons why machine-learning scoring functions perform so well in predicting the binding affinities of diverse protein-ligand complexes. It also investigates how this performance improvement over classical scoring functions is expected to widen with the future availability of more training data. As explained, the 382 complexes appeared in 2013 were predicted with models trained on data up to 2002 (792 complexes in partition 2), 2007 (1300 complexes in partition 3), 2010 (2059 complexes in partition 4) and 2012 (2897 complexes in partition 5).

Figure 7.6 shows how the performance in predicting binding affinity varies with training data size. RF-based scoring functions, i.e. models 3 and 4, are represented by boxplots to show how their performance varies using 10 different random seeds for training on the same data. Model 1 is off-the-shelf software and model 2's MLR is deterministic, so they are not stochastic and

only a single performance value was obtained from a particular training data set. The comparison between models 2 and 3 is particularly striking. The performance of model 2, effectively a classical scoring function, does not improve with training data set size. By contrast, model 3, whose only difference with model 2 is in using RF instead MLR as the regression model, greatly improves with more data. This demonstrates that the circumvention of an additive functional form is one of the reasons.

The second reason why machine-learning scoring functions perform so well is that RF is capable of effectively exploiting a more comprehensive set of structural features. This can be seen by comparing the performance of models 3 and 4, which only differ in that model 4 uses 36 features in addition to the 6 features used by model 3. Not only the difference in performance is substantial but grows as more data is available for training. This observation increases again the importance of using RF in the future.

The same conclusions are reached from this complementary view of performance: a non-parametric regression model performs better because it circumvents modelling assumptions and is capable of exploiting richer structural descriptions of the complex. For the training set with the lowest number of complexes, model 2 outperforms model 3, indicating that the additivity assumption of classical scoring functions was the best approach back in the days when few structures were available to calibrate the scoring function. It is also noteworthy that model 1 performs worse than model 2, which suggests that the nonlinear OLS used

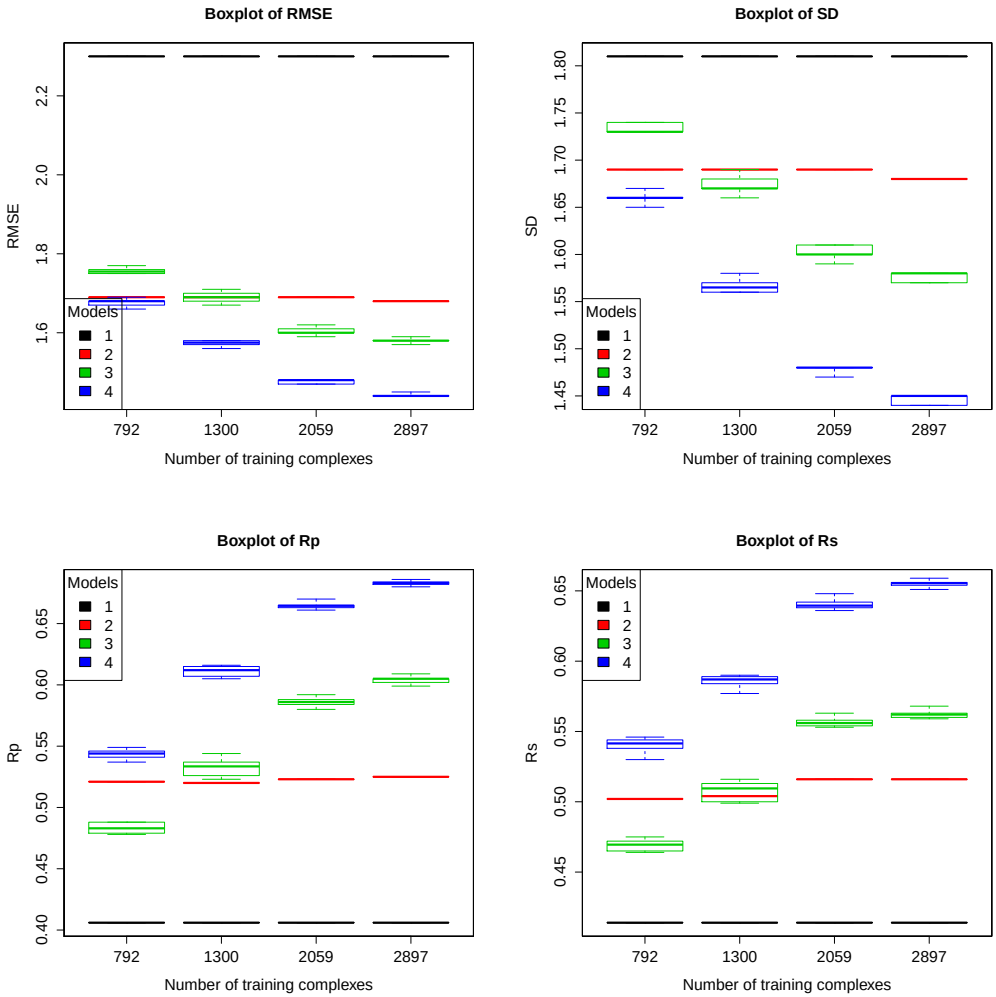


Figure 7.6: Performance in predicting binding affinity on the 382 new complexes in 2013 using training sets formed by the complexes known in 2002, 2007, 2010 and 2012.

in Vina is not as suitable as MLR, at least for this crucial aspect of docking, i.e. scoring complexes. Note that model 1 represents the situation where the Vina's scoring function was used out of the box without retraining for each set, although here we have seen that improvements with data set size can only be gained by using the appropriate regression model.

7.5.8 Machine-learning scoring functions can also be used to interpret docking results

In addition to predicting binding affinity, the magnitude of the terms or features of a scoring function for a docking pose can be used to find out which the most important contributions to binding are. There are a number of approaches for this sensitivity analysis. In a chemical series, one can look at how the value of each feature correlates with measured binding affinity, the more important for binding being those obtaining a higher correlation. For a particular docking pose, one can multiply each feature with its weight to obtain the energetic contribution to binding of each feature. Because knowledge-based scoring functions typically have no weights, it has been claimed [14] that these can barely provide immediate physical interpretation of the results. In reality, one could also evaluate the scoring function for that pose with each feature set to zero in turn, with the features resulting in the largest variation in the predicted pKd value being the most important. This can also be easily implemented in machine-learning scoring functions to understand

ligand binding.

Another potential issue is that the set of features may not be directly related to intermolecular interactions. This is not the case of model 3, which has the same six directly interpretable features as model 2 representing an empirical scoring function. Nevertheless, model 4 incorporates 36 additional features which are not directly interpretable. Before assessing whether this is a drawback, we should remember what interpretability is ultimately useful for. Often, the optimisation of ligand potency is carried out by synthesising derivatives that preserve important favourable interactions and reduce unfavourable interactions according to such interpretation. However, extracting knowledge from a docking pose using a less accurate scoring function to use the derived knowledge to optimise ligand potency is apparently less accurate than simply using the most accurate scoring function to score all possible derivatives in order to synthesise those that are predicted to be more potent. Therefore, rather than a drawback, we believe that circumventing the interpretability stage can be an advantage in structure-based optimisation.

7.5.9 The applicability domain of the developed scoring functions

The applicability domain of a regression model is given by the set of training data points and how they are represented. In consequence, models 2 and 3 have the same applicability domain and thus are expected to work well on the same types of

small molecules and proteins. Model 4 is trained and tested on the same data sets and thus should have a similar applicability domain than models 2 and 3. However, because model 4 incorporates an additional set of features, so the data representation is different and thus the applicability domain is not exactly the same. It is important to note that these features encode neither the chemical structure of the ligand nor the sequence information of the protein. Therefore, there is no reason to think that its applicability domain is more restricted to the chemotypes and protein families in the training set any more than other scoring functions trained on the same data are.

7.6 Conclusions

We have seen that one can greatly improve Vina by circumventing its assumed functional form using RF as the regression model and expanding the set of features describing the complexes. The resulting machine-learning scoring functions have either the same or very similar applicability domain by construction. Furthermore, we have explained how these scoring functions could be also used to understand binding. However, we have also argued that extracting knowledge from the description that a scoring function provides of a docking pose is a suboptimal way to improve ligand binding, as the direct application of a more accurate scoring function on ligand derivatives should perform better. We have demonstrated that the tendency of RF-based scoring functions to overfit training data is

not a limitation but instead a trait of these regression models which are robust to overfitting. Finally, we have also suggested that incorporating ligand- and protein-only properties into the scoring function is a promising path to future improvements.

Another big contribution of this study is the release of free software implementing RF::VinaElem so that it can be directly used by the large number of Vina users, and literally by all users after converting the file format to PDBQT using OpenBabel [187] or AutoDock Tools [32]. Given the large number of Vina users and the large increase in scoring accuracy achieved, we have trained the best of our models on the most comprehensive set of high-quality complexes and implemented it as easy-to-use free software that directly re-scores Vina-generated poses. Specifically, we refer to RF::VinaElem as RF-Score-v3, and we have trained two RF models respectively on the 2959 complexes from PDBbind v2013 refined set and on the 3444 complexes from PDBbind v2014 refined set. RF-Score-v3, given its accuracy at ranking complexes, should generally perform well on structure-based drug lead optimization.

Because classical scoring functions generally use MLR as the regression model and we have shown its inability to improve with larger sizes of structural data, we expect that the performance of any of these will be boosted by following the same procedure we have applied here to Vina. We therefore suggest developers modify their scoring functions accordingly so that users can enjoy a much higher predictive accuracy. Using non-parametric machine learning remains a largely unexplored

approach to developing scoring functions. For example, the incorporation of ligand-only, protein-only and alternative inter-molecular features, e.g. [239], is still to be fully investigated.

The proposed PDBbind 2013 benchmark, effectively a blind test using four time-stamped training sets, has revealed that the performance difference between classical and machine-learning scoring functions will be larger as more structural data becomes publicly available in the future. These machine-learning scoring functions could include the very large number of experimentally determined structures that are continuously generated by the pharmaceutical industry and the academic institutes. Confidentiality is not be a problem because only the inter-molecular features and binding affinities of these structures are required to train scoring functions, from which it is impossible to decode the identity of either the targets or the bound molecules. It is important to note that this is a new opportunity because, as we have shown here, the regression model adopted by classical scoring functions would not be able to exploit new flood of data.

As usual, e.g. [148], the performance of generic scoring functions has been assessed by measuring their ability to predict the binding affinities of diverse protein-ligand complexes. Given its accuracy at this task, RF-Score-v3 should generally perform well on structure-based drug lead optimization applications. We have however not yet investigated its ability to discriminate between binders and non-binders in virtual screening (VS) settings, as it is important to first study binding affinity prediction in isolation so as to avoid additional confounding factors such as true

binders that might not bind to the assumed binding conformation or pocket as well as assumed non-binders that might be actually binding. This problem, known as enrichment, belongs to another dedicated study, as it has previously been the case [54, 240], mainly because additional research is required to find an optimal configuration of the scoring function, which might involve different features and training strategies. Since these issues are hence out the scope of this study, we are not making any claim about how RF-Score-v3 will compare to classical scoring functions on VS benchmarks. However, we expect that it will excel at VS because: 1) excellent prospective results have already been achieved with the less accurate RF-Score-v1 [140], 2) pose generation error has typically a low impact on binding affinity prediction [9], 3) accurate ranking by the affinity of true binders is a necessary condition for top VS performance, and 4) non-binders are nothing but extremely weak binders whose low affinity should be best predicted by RF-Score-v3. In fact, machine-learning scoring functions have already demonstrated substantial improvements over classical scoring functions on VS benchmarks [241].

7.7 Availability

RF-Score-v3 is free and open source under Apache License 2.0. The source code is available at https://github.com/HongjianLi/RF-Score. The precompiled 64-bit executables for Linux and Windows, a README file for operating instructions, a prebuilt RF

file, a sample protein file and a sample ligand file in PDBQT format are available at http://istar.cse.cuhk.edu.hk/rf-score-3.tgz.

7.8 Future works

The current study can be extended in multiple aspects. From the perspective of training data compositions, leave-cluster-out cross validation (LCOCV) [198] is becoming a popular validation method in evaluating scoring function performance of predicting binding affinity of truly new protein targets. [242] compares confirmed inactive and randomly selected compounds as negative training examples in support vector machine-based virtual screening. [243] analyzes the influence of negative training set size on machine learning-based virtual screening.

From the perspective of new benchmarks, the well known PDBbind benchmark has recently been updated to CASF-2013 [244], and 20 scoring functions, most of which are implemented in mainstream commercial software, have been evaluated in terms of "scoring power" (binding affinity prediction), "ranking power" (relative ranking prediction), "docking power" (binding pose prediction), and "screening power" (discrimination of true binders from random molecules) [245].

From the perspective of features, the DRAGON 6 software [246], available at http://www.talete.mi.it/, can calculate 4885 molecular descriptors. PaDEL-descriptor [247] and QuBiLS-MIDAS [248] are free software for molecular descriptors computation. It is also possible to borrow the ideas of UFSRAT

[249] and USRCAT [20] and generate features from subsets of atoms.

From the perspective of regression models, the random generalized linear model (RGLM) has recently been proposed [250] as a highly accurate and interpretable ensemble predictor, and subsequently evaluated in predicting the status of COPD (chronic obstructive pulmonary disease) [251].

From the perspective of applications, it is interesting to see an analogous scoring function tailored to the enrichment problem of discriminating between binders and non-binders, as well as scoring functions specific to protein families [252]. Moreover, like T-PioDock [253], machine-learning scoring functions can be applied to protein-protein docking.

□ **End of chapter.**

Chapter 8

RF-Score-v4: pose generation error

In prospective virtual screening, accurate prediction of binding affinity of docked poses is crucial for ranking compounds. However, many existing studies focus on scoring crystal complexes only, without considering the impact of pose generation error. Therefore the high accuracy claimed in those studies would potentially lead to degradation of predictive performance when their methods are applied to scoring docked poses.

In this study we investigate the impact of pose generation error on the predictive performance of both classical and machine-learning scoring functions. We also study their capability of predicting the near-native pose that is most conformationally closest to the crystal pose. Our results show that pose generation error affects the accuracy of scoring functions, which is well anticipated. To minimize this negative impact, re-training the scoring functions on docked poses instead of crystal poses can be a straightforward solution. On the other hand, we find that

although machine-learning scoring functions are generally good at binding affinity prediction, they do not perform as well as classical scoring functions on native pose prediction. This indicates that predictions of binding affinity and native pose are two different tasks and no single scoring function performs optimally for both tasks.

This was a collaborative project with Pedro J. Ballester from Cancer Research Center of Marseille, Marseille, France. It was published in *Proceedings of the 11th International Meeting on Computational Intelligence Methods for Bioinformatics and Biostatistics (CIBB)* on 26 June 2014 [18].

8.1 Background

Protein-ligand docking predicts the binding conformation (pose) of a ligand when bound to a target protein to form a stable complex, as well as their binding affinity. The former process is known as pose generation and the latter is known as scoring. The current limitation is in accurate scoring. In the previous chapter, we have already seen that RF-Score-v3 accurately predicted the binding affinities of crystal protein-ligand complexes.

8.2 Motivation

Although there have been quite some studies [15–17] on scoring, they only concentrate on crystal complexes in order to avoid confounding factors introduced by pose generation, therefore their

methods and conclusions are only applicable to scoring crystal complexes. However, scoring of the docked poses of a molecule is required when the experimentally determined pose is not available. This is the common case in prospective virtual screening, such as our istar web service. Hence accurate prediction of binding affinity of a docked pose can be practically meaningful in sorting candidate compounds in a database. Although RF-Score-v3 has been vigorously validated on crystal poses, its predictive power on docked poses is yet to be investigated.

8.3 Objective

Here we study the impact of pose generation error on classical and machine-learning scoring functions. Furthermore, we investigate which of these scoring functions is the most suitable for predicting the near-native pose, i.e. the docked pose most similar to the crystal pose. This kind of capability is referred to as "docking power" in some other studies [245].

This study can be regarded as an extension to the previous chapter [16, 17]. The same models, materials and metrics were reused, with some slight adjustments specifically for the purpose of this study. Likewise, the numerical experiments were performed with AutoDock Vina [8] as the classical scoring function because it is one of the most popular docking software, and RF-Score [10] as the machine-learning scoring function because it has been vigorously studied in multiple aspects [9, 50, 140]. Note that although this chapter and the previous chapter share

some similarities in terms of methods and materials, their applications are fundamentally different.

8.4 Methods

We reused the four models, the two benchmarks, and the four performance measures described in the previous chapter. Here we only highlight the differences that were made particularly for investigating the impact of pose generation error. For this purpose, new experiments were designed to generate and measure docked poses.

8.4.1 Model 1 - AutoDock Vina

Vina's score for the kth pose of a molecule is given by the predicted free energy of binding to the target protein and computed in Vina as:

$$e_k = \frac{e_{k,inter} + e_{k,intra} - e_{1,intra}}{1 + w_6 N_{rot}} \tag{8.1}$$

Unlike the previous chapter, where $k = 1$ because only the crystal pose was considered, in this study we aim at docked poses and thus k can be an arbitrary value. Therefore $e_{k,intra}$ and $e_{1,intra}$ cannot be cancelled out. As a result, five more features from $e_{k,intra}$ were incorporated in models 2, 3 and 4, constituting a feature vector of 11 elements.

8.4.2 Model 2 - MLR::Vina

This is a multiple linear regression (MLR) model using the 11 unweighted Vina terms as features. Similarly, in order to make the problem amenable to MLR, we made a grid search on the w_6 weight and thereafter ran MLR on the remaining ten weights. The sampling range for w_6 was extended to [0.000 to 0.030] with a step size of 0.001 because of more variability when multiple docked poses were considered.

8.4.3 Model 3 - RF::Vina

This is a random forest (RF) model with the 11 Vina features using the default 500 trees and mtry values from 1 to 11. The selected model was the one that provided the lowest RMSE (Root Mean Square Error) on the OOB (Out of Bag) data. This process was repeated ten times with ten different random seeds because RF is stochastic.

8.4.4 Model 4 - RF::VinaElem

This is an extension to RF::Vina and incorporates the 36 RF-Score features. Hence, it was built with 47 features. This process was also repeated ten times with ten different random seeds.

8.4.5 The PDBbind benchmark

We reused the PDBbind v2007 benchmark because the four models have been evaluated on it in the previous chapter, permitting a direct comparison. Briefly, the test set comprises

195 diverse complexes with measured binding affinities spanning more than 12 orders of magnitude, whereas the training set comprises 1105 non-overlapping complexes.

8.4.6 The 2013 blind benchmark

We reused the PDBbind v2013 blind benchmark because the four models have been evaluated on it in the previous chapter, allowing a direct comparison. Briefly, the test set comprises 382 complexes newly added in the 2013 release, whereas the training set comprises 2897 complexes from PDBbind v2012 refined set.

8.4.7 Performance measures

We reused the four performance measures: root mean square error RMSE, standard deviation SD, Pearson correlation coefficient Rp and Spearman correlation coefficient Rs between predicted and measured binding affinity. Note that RMSE was not calculated in a linear correlation, while SD was.

8.4.8 Experimental design

To generate docked poses, each ligand in the two benchmarks was docked into the binding site of its target protein using Vina. This process is known as redocking. As usual [9], the search space was defined first by finding the smallest cubic box that covers the entire ligand and then by extending the box in X, Y, Z dimensions by 10Å. Redocking a ligand resulted in up to nine docked poses output by Vina.

Here we define two schemes to refer to different poses from which the features are extracted. In scheme 1, the chosen pose is the crystal pose. In scheme 2, the chosen pose is the docked pose with the best Vina score, i.e. the one with the lowest Vina score in terms of estimated free energy. We trained the four models on both crystal and docked poses (in both schemes), and tested them also on both crystal and docked poses (in both schemes). Hereafter whenever we mention the docked pose, we implicitly refer to the one with the best Vina score, if not specified explicitly.

8.5 Results

After redocking by Vina, we used root mean square deviation (RMSD) to quantify the pose generation error, i.e. how different the 3D geometry of the redocked pose is from the corresponding crystal pose of the same ligand molecule. A RMSD value of 2Å was used as a publicly accepted positive control for correct bound structure prediction. 101 out of the 195 ligands (52%) in the PDBbind v2007 benchmark and 219 out of the 382 ligands (57%) in the PDBbind v2013 blind benchmark had their best-scoring docked pose with RMSD $<$ 2Å. When all the docked poses (up to nine) were considered, these redocking success rates increased to 76% and 81%, respectively. These results are consistent with those obtained in [9], where Vina managed to predict a conformation sufficiently close to that of the co-crystallized ligand as the first conformation in over half of the

Table 8.1: Performance of the four models trained on crystal and docked poses and tested also on crystal and docked poses on the PDBbind v2007 benchmark.

Model	Training	Test	RMSE	SD	Rp	Rs
1	Crystal	Crystal	2.41	1.99	0.554	0.608
2	Crystal	Crystal	1.88	1.85	0.630	0.680
3	Crystal	Crystal	1.66	1.59	0.744	0.752
4	Crystal	Crystal	1.52	1.42	0.803	0.799
1	Crystal	Docked	2.02	1.98	0.557	0.597
2	Crystal	Docked	1.90	1.87	0.622	0.670
3	Crystal	Docked	1.76	1.72	0.693	0.710
4	Crystal	Docked	1.60	1.52	0.772	0.771
2	Docked	Crystal	1.91	1.88	0.618	0.648
3	Docked	Crystal	1.74	1.69	0.705	0.716
4	Docked	Crystal	1.58	1.45	0.794	0.790
2	Docked	Docked	1.86	1.83	0.640	0.667
3	Docked	Docked	1.69	1.63	0.730	0.730
4	Docked	Docked	1.55	1.45	0.795	0.789

cases.

Tables 8.1 and 8.2 enumerate the predictive performance of the four models trained on crystal and docked poses and tested also on crystal and docked poses on the PDBbind v2007 benchmark and the PDBbind v2013 blind benchmark, respectively. Figures 8.1 and 8.2 plot the same results graphically, where trn-1 means the model was trained in scheme 1, i.e. on crystal poses, and trn-2 means the model was trained in scheme 2, i.e. on docked poses. Likewise, tst-1 and tst-2 mean the model was tested on crystal and docked poses, respectively. Note that model 1 was trained on crystal poses and used out of the box without re-training, so its results of trn-1 are simply repeated for trn-2.

From these results on both benchmarks, several interesting

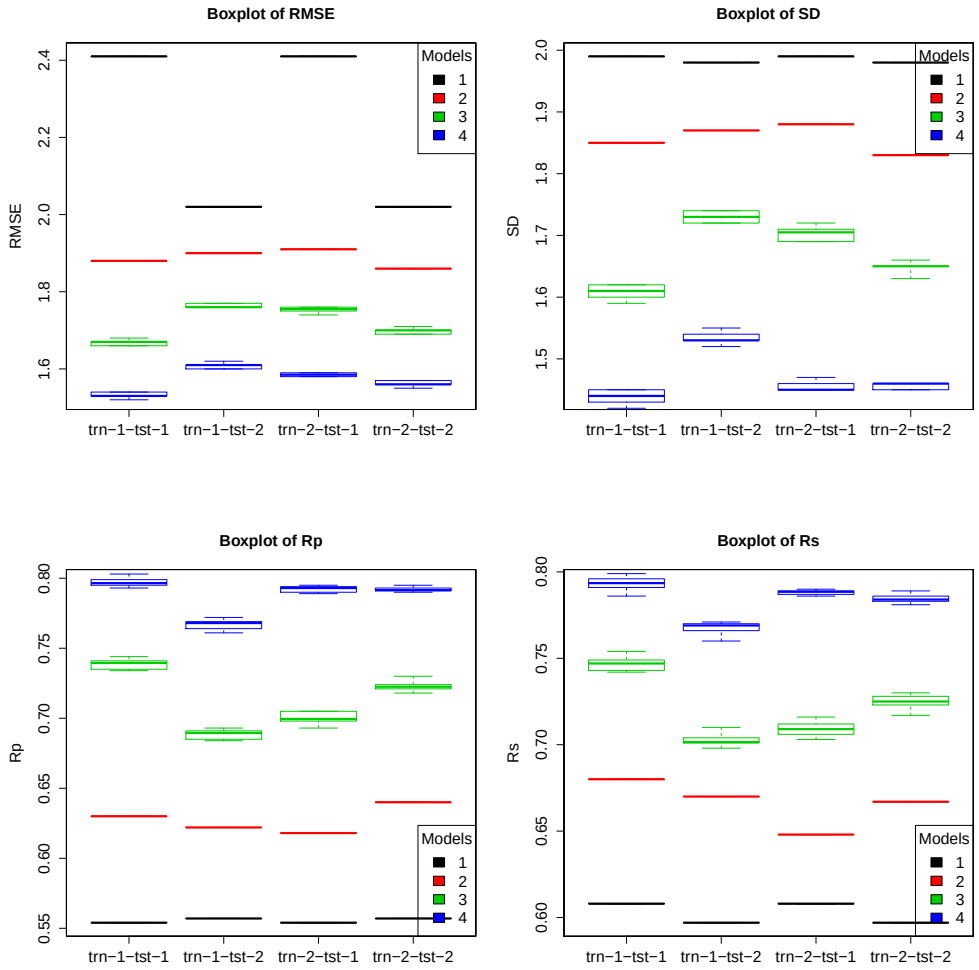


Figure 8.1: Performance of the four models trained on crystal and docked poses and tested also on crystal and docked poses on the PDBbind v2007 benchmark.

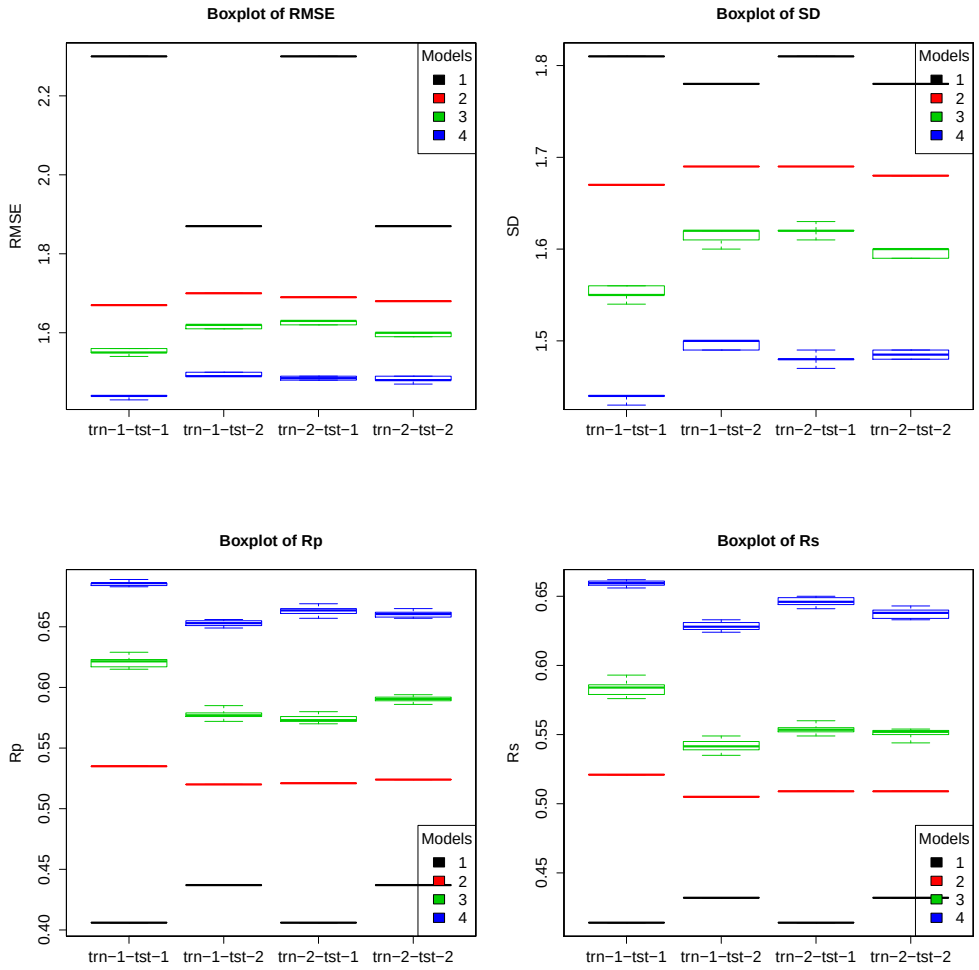


Figure 8.2: Performance of the four models trained on crystal and docked poses and tested also on crystal and docked poses on the PDBbind v2013 blind benchmark.

Table 8.2: Performance of the four models trained on crystal and docked poses and tested also on crystal and docked poses on the PDBbind v2013 blind benchmark.

Model	Training	Test	RMSE	SD	Rp	Rs
1	Crystal	Crystal	2.30	1.81	0.406	0.414
2	Crystal	Crystal	1.67	1.67	0.535	0.521
3	Crystal	Crystal	1.54	1.54	0.629	0.593
4	Crystal	Crystal	1.43	1.43	0.689	0.662
1	Crystal	Docked	1.87	1.78	0.437	0.432
2	Crystal	Docked	1.70	1.69	0.520	0.505
3	Crystal	Docked	1.61	1.60	0.585	0.549
4	Crystal	Docked	1.49	1.49	0.656	0.633
2	Docked	Crystal	1.69	1.69	0.521	0.509
3	Docked	Crystal	1.62	1.61	0.580	0.560
4	Docked	Crystal	1.48	1.47	0.669	0.650
2	Docked	Docked	1.68	1.68	0.524	0.509
3	Docked	Docked	1.59	1.59	0.594	0.553
4	Docked	Docked	1.47	1.48	0.665	0.643

phenomena are observed. First, for model 1, its performance tested on docked poses was always better than its performance tested on crystal poses, except for the Rs performance on the PDBbind v2007 benchmark. Vina performing better on docked poses is likely to be due to the fact that docked poses are by construction optima of the objective function spanned by the Vina score, which may favor prediction of docked poses over unoptimized crystal poses.

Second, for models 2, 3 and 4 trained on crystal poses, their performance tested on docked poses was always worse than their performance tested on crystal poses. This is well anticipated because of the impact of pose generation error.

Third, for models 2, 3 and 4 tested on docked poses, their performance was better when they were trained on docked poses

than their counterparts trained on crystal poses. This implies that a simple and quick solution to improving performance on docked poses is to re-train the model on docked poses instead of on crystal poses.

Fourth, for models 2, 3 and 4 tested on crystal poses, the models trained on docked poses did not outperform their counterparts trained on crystal poses. This is also well anticipated due to the impact of pose generation error, and suggests that it is not feasible to improve the predictive performance on crystal poses by using docked poses for training.

Fifth, regardless of the training or test schemes, model 4 consistently outperformed model 3, which in turn outperformed model 2, which in turn outperformed model 1. It is remarkable that the best scoring function, RF::VinaElem, when trained on docked poses, achieved the highest performance in the literature on the PDBbind v2007 benchmark in the more common application of re-scoring docked poses. Here we denote this version of RF::VinaElem as RF-Score-v4 specifically for the purpose of binding affinity prediction given a docked pose. Importantly, since Vina and RF::Vina used the same features and were trained on the same data, RF::Vina performed much better in predicting binding affinity than the widely-used Vina while having the same applicability domain.

Next, we assessed the ability of each of the four models to predict the near-native pose from the up to nine docked poses output by Vina (Table 8.3). In other words, we would like to see if a model could correctly assign the best score to the particular

Table 8.3: Performance of the four models in near-native pose prediction.

	PDBbind v2007 benchmark		PDBbind v2013 blind benchmark	
Model	#	%	#	%
1	94	48	208	54
2	59	30	142	37
3	53	27	119	31
4	59	30	141	37

docked pose having the lowest RMSD to the crystal pose out of at most nine docked poses. Interestingly, results show that Vina, although being the least accurate predictor of binding strength, turned out to be the best at predicting which docked pose is geometrically the closest to the crystal pose. This is probably due to the fact that, as explained previously, the best-scoring docked pose was resulted from the optimization of Vina's scoring function during redocking. In contrast, the presented machine-learning scoring functions, while excelling at binding affinity prediction, performed much worse than Vina at native pose prediction. This indicates that these two tasks, binding affinity prediction and native pose prediction, cannot be optimally covered by a single scoring function.

8.6 Conclusions

This study has demonstrated that errors in pose generation generally introduce a degradation in the accuracy of scoring. One straightforward approach to enhancing predictive accuracy on docked poses is to re-train the scoring function also on docked poses. Furthermore, RF-Score-v4, essentially RF::VinaElem trained

on docked poses, obtained the highest predictive performance on two PDBbind benchmarks in the common scenario where one has to predict the binding affinity of docked poses instead of those for crystal poses, usually because a crystal structure of the ligand is unavailable. Nevertheless, we observed that the presented machine-learning scoring functions did not perform as well as Vina in predicting the near native pose of a ligand. This could be due to the confounding factor that the docked poses were all generated and optimized by Vina. It is out of the scope of this study to investigate the generalization of these conclusions to other machine learning methods such as support vector regression, but we expect them to yield similar conclusions.

8.7 Future works

The redocking experiment was carried out by Vina in this study. It is of great interest to repeat the experiment with idock to see if the same conclusions still hold. If so, we can substitute RF-Score-v4 for RF-Score-v3 in our istar web platform for large-scale prospective virtual screening.

□ **End of chapter.**

Chapter 9

USR@istar: ultrafast shape recognition

Finding compounds structurally similar to a query ligand has been an important but daunting problem for a long time. The USR (Ultrafast Shape Recognition) algorithm represents a whole new alignment-free method that encodes the shape information semantically and permits superfast screening of a large molecular database. A few extensions to USR improve the original method from various perspectives and three of them, UFSRAT, EDULISS and SwissTargetPrediction, are also available as web servers. However, UFSRAT and EDULISS are unable to discriminate between long, chain-like molecules, and their calculated distributions are not meaningful when some pharmacophoric features are rarer than others. SwissTargetPrediction uses a small, well annotated, bioactive compound database and is generally for target fishing.

For prospective virtual screening purposes, in this study we have implemented USR and its extension USRCAT (USR with

Credo Atom Types) on top of our istar web platform. We reused the large molecular database of more than 23 million diverse ligands originally accompanied with idock, and exploited three levels of parallelism with a novel implementation of sum of absolute differences using AVX (Advanced Vector Extensions) to accelerate similarity score calculation. Our USR@istar supports a query ligand in either SDF, MOL2, XYZ, PDB or PDBQT format, and interfaces with our iview WebGL visualizer for interactive visualization of high-score hits. USR@istar is freely available at http://istar.cse.cuhk.edu.hk/usr. In the future we will exploit caching and indexing algorithms to further speed up USR matching and implement geometrical and functional clustering.

To thoroughly benchmark USR@istar, we selected 19 query ligands with different molecular sizes. Not unexpectedly, ranking by USR or USRCAT score yielded different output, particularly if the query ligand was large. When the query ligand had more than four rotatable bonds, both methods failed to recover the query ligand in a different pose in the output due to the large torsional diversity implied. Surprisingly, input file format was found to affect the classification of atoms into predefined pharmacophoric subsets. We also observed that loading database features in advance significantly reduced the matching runtime from 167 seconds to just 30 seconds on average for a query.

We believe USR@istar, which circumvents the requirement of macromolecular structure, would be an excellent supplement to our existing idock@istar.

This is an ongoing collaborative project with Pedro J. Ballester from Cancer Research Center of Marseille, Marseille, France.

9.1 Background

Molecular shape has been widely acknowledged as a key factor for biological activity and is thus regarded as a very important pattern for drug searching. Searching a molecular database for compounds that most closely resemble the shape of a given query molecule, be it a known inhibitor of a target protein, a natural product, or even a patented compound, finds pragmatic applications in ligand-based virtual screening [140, 254–258] and target fishing [259–262]. Therefore molecular similarity search can assist in discovering structurally novel active compounds, or in identifying potential interacting target of bioactive ligands, which is useful for understanding the polypharmacology and safety profile of existing drugs. Furthermore, this approach can be applied to other scientific disciplines such as performing similarity comparisons between proteins or designing content-based Internet search engines for 3D geometrical objects [263].

The molecular shape similarity can be quantified by methods based on structural alignment [264–267] or shape recognition [19, 20, 268]. Structural alignment is also known as molecular superposition, and requires precise geometric comparison, which is often computationally demanding. Shape recognition, on the other hand, encodes shape information into a numerical feature vector, which can be subsequently used to compute a similarity

score between two molecules very efficiently.

USR (Ultrafast Shape Recognition) [19] was the very first non-superposition method for molecular shape comparison, and demonstrated superior computational performance at least three orders of magnitude faster than previously existing alignment-based methods. USR has another major advantage of being invariant to spatial rotation and translation, and hence circumvents the problematic requirement of aligning molecules. USR defines the shape of a molecule independently and for every molecule uses a fixed set of 12 descriptors derived from the first 3 statistical moments of distributions of interatomic distances between atoms and 4 purposely-selected centroids. This encoding scheme ensures that every molecule has a unique location in the 12-dimensional chemical space spanned by the used descriptors, and consequently enables finding and visualizing clusters of molecules with similar shapes [254, 263]. Selecting the most representative molecule of each cluster can avoid repeating expensive biological tests on similar molecules [263]. The ability of USR as a standalone method was studied to identify molecules sharing common biological activities through retrospective [254] and prospective [140, 255–258] virtual screening experiments. Prospectively, USR was applied to the discovery of inhibitors of arylamine NATs (N-acetyltransferases) [255], DHQase2 (dehydroquinase type 2) [140], PAD4 (protein arginine deiminase type 4) [256], p53-MDM2 (murine double minute 2) [257], and PRL-3 (phosphatase of regenerating liver) [258]. USR was also used for deduplication in a virtual screening campaign [269] and

in our iSyn [12, 13] *de novo* ligand design software.

Since USR was devised in 2007, there have been quite a few extensions [20, 249, 260, 261, 268, 270–274] to augment the original method. [270] presented a hybrid approach composed of USR and the topological MACCS key descriptors, which are binary in nature and encode the presence or absence of 166 predefined structural fragments. It used the first four unbalanced moments of each distribution of atomic distances and incorporated additional chemical information through 2D structural similarity. Random Forest [139] was used for multi-class classification. Incorporating an additional central moment, the kurtosis, was found to significantly improve the performance. The addition of the fifth central moment, however, did not improve the performance sufficiently to justify the increased computational expense.

UFSRAT [249] addressed the lack of discrimination between compounds having similar shape but distinct pharmacophoric features by subdividing atoms into four subsets which are heavy, hydrophobic, hydrogen bond acceptor or donor atoms, according to their atom types. For each subset, the four centroids were calculated, and so were the 12 USR descriptors. Therefore 48 descriptors were resulted. This was to ensure that similar compounds are able to make the same type of interactions within biological systems as the query ligand. UFSRAT was prospectively applied to the discovery of inhibitors of 11β-HSD1 (hydroxysteroid dehydrogenase type 1) [275]. UFSRAT is available as a web server at http://opus.bch.ed.ac.uk/ufsrat/,

which contains 28 databases with the largest one containing 4,853,000 conformers. UFSRAT is also employed for geometrical similarity searches in the EDULISS database [271] available at http://eduliss.bch.ed.ac.uk/, which comprises over 5 million commercially available compounds.

CSR [272] and USR:OptIso [273] attempted to tackle the lack of discrimination between chiral compounds. Their novel idea was to position the centroids in such a way that they clearly distinguish between enantiomers, i.e. optical isomers. They both used cross product because it is an operator that transforms equivariantly under rotations and translations, but not under reflections. The two methods differed in selecting the centroids and in replacing or supplementing the new optical isomerism descriptor [273]. CSR [272] was tested on the DUD (Directory of Decoys) dataset [276], where a significant improvement in enrichment was found over USR. USR:OptIso [273] was shown to be helpful for analyzing molecules with stereogenic centers, atropisomerism, and in the clustering of conformers generated by systematic bond rotation.

ElectroShape [268, 274] extended the CSR [272] method by encoding electrostatics and liphophilicity through additional dimensions and centroids. In [274], the partial charge was represented as a fourth coordinate, with atoms being identified by points in four-dimensional space. ElectroShape was validated using release 2 of the DUD dataset [276], and showed a near doubling in enrichment over USR and CSR. Different implementations of partial charge were also revealed to affect the

enrichment performance significantly. The addition of a fourth statistical moment, as was done in [270], improved USR and CSR but not ElectroShape, suggesting that adding extra information might not necessarily improve enrichment but could dilute the information already included. In [268], ElectroShape was further extended by using atomic lipophilicity as an additional molecular property, with atoms being identified by points in five-dimensional space. This version of ElectroShape showed a clear improvement in performance, indicating that adding extra independent atomic properties makes shape-based enrichments even better.

USRCAT [20] extended the UFSRAT [249] method by identifying five subsets of atoms with the help of the SMARTS patterns used for atom typing in the CREDO database [277, 278]. The five subsets were chosen to be heavy, hydrophobic, aromatic, hydrogen bond acceptor or donor atoms. Aromaticity was added to USRCAT as a pharmacophoric subset because USR was unable to discriminate between long, chain-like molecules such as certain heteropeptides and long alkylchains in particular. Unlike UFSRAT [249], USRCAT [20] derived the four centroids from heavy atom coordinates and used them to calculate the distributions for all the five subset moments to improve screening performance. USRCAT was shown to outperform the traditional USR method in a retrospective virtual screening benchmark on the DUD-E (Directory of Decoys, Enhanced) dataset [279]. The highest enrichment factors were only achieved if the LEC (Lowest Energy Conformer) of an active was

used as a query and if the LECs were included in the target set, but this observation could not be generalized. DUD-E was found to be not ideal to benchmark the virtual screening performance of global shape similarity algorithms such as USR and its variants due to the large variations in molecular size of the active ligands.

A recent study [260] used a reference set of 224,412 molecules active on 1,700 human proteins and showed that accurate target prediction can be achieved by using a multiple logistic regression to combine different measures of chemical similarity based on both chemical structure and molecular shape, with the former using FP2 fingerprints and the latter using ElectroShape [268]. This hybrid method was subsequently encapsulated into the SwissTargetPrediction [261] web server, freely available at http://www.swisstargetprediction.ch/, to identify new targets for uncharacterized molecules or secondary targets for known molecules. With data collected from the ChEMBL database version 16 [280], the molecular library was expanded to 280,000 compounds active on 2,686 targets of the organisms of human, mouse, rat, cow and horse. Mapping predictions by homology within and between different species, a powerful approach to translate results obtained in model organisms to human, were enabled for close paralogs and orthologs.

Table 9.1 summarizes USR and its variants.

Table 9.1: Summary of USR-like methods.

method	novelty	references
USR	encoding shape by atomic distribution	[19]
USR+MACCS	incorporating chemical similarity	[270]
UFSRAT	subdiving atoms into pharmacophoric subsets	[249, 271]
CSR	repositioning reference locations	[272]
USR:OptIso	repositioning reference locations	[273]
ElectroShape	expanding coordinate dimension	[268, 274]
USRCAT	subdiving atoms into pharmacophoric subsets	[20]
SwissTargetPrediction	mixing ElectroShape and chemical similarity	[260, 261]

9.2 Motivation

Among the USR variants mentioned above, only three of them [249, 261, 271] have been made available as web servers together with different databases for different purposes. The UFSRAT [249] and EDULISS [271] web servers both employ the UFSRAT [249] method for ligand similarity search. However, as pointed out in [20], this method is incapable of discriminating between long, chain-like molecules such as certain heteropeptides and long alkylchains because aromaticity is not considered as a pharmacophoric subset; besides, calculating the four centroids for each set of atoms individually is problematic because either the parameters cannot be calculated at all or the underlying distance distributions are not with respect to the overall shape of a molecule and not meaningful when some pharmacophoric features are rarer than others. The UFSRAT [249] web server restricts the input query ligand to be one molecule in SDF format only, and does not support online visualization. The EDULISS [271] web server requires drawing a query structure in a Java

molecular editor, which is being disabled on more and more systems due to security concerns. The SwissTargetPrediction [261] web server, on the other hand, comprises well-annotated active compounds and is primarily used for predicting the target proteins of bioactive small molecules but not for prospective virtual screening purposes.

9.3 Objective

In this project we aimed to provide an istar-based [9] web service for large-scale prospective virtual screening using USR-like methods. We chose to employ USR [19] and USRCAT [20] because they have demonstrated pragmatic usefulness in prospective [140, 255–258] and retrospective [20] virtual screening experiments, respectively, and their Python source code is freely available for studying the precise implementation and porting to other programming languages such as C++ and JavaScript.

Our USR@istar has several distinctive features. First, it uses both USR and USRCAT to search a large database comprising 23 million compounds collected from ZINC [27, 28]. Second, it utilizes three levels of parallelism, both coarse grained and fine grained, to accelerate job execution. Third, it supports a query ligand in one job in five formats, and interfaces with the iview [11] WebGL visualizer to display results in an interactive manner.

9.4 Methods

This section first reviews the general methods of USR [19] and USRCAT [20], and then introduces their specific implementations on our istar platform [9].

9.4.1 USR and USRCAT

USR is based on the observation that the shape of a molecule is uniquely determined by the relative position of its atoms, which is in turn determined by the set of all interatomic distances. This convenient representation is independent of molecular orientation or position, and thus eliminates any need for alignment or translation. The interatomic distances are heavily constrained by the forces that hold the atoms together, and hence they contain more than sufficient information to accurately describe molecular shape. So it is possible to use a set of atomic distances from only a small number of strategic reference locations uniquely defined in every molecule, and meanwhile retain the discriminative power necessary to distinguish between molecules.

The four reference locations are selected to be the molecular centroid (ctd), the closest atom to ctd (cst), the farthest atom to ctd (fct), and the farthest atom to fct (ftf). These locations represent the center of the molecule and its extremes, and are thus well separated. In this way molecular shape is described by four distributions of atomic distances, where the number of atomic distances is proportional to the number of atoms. In

order to compare molecules with different number of atoms, the first three moments of these distributions are computed and used to encode the shape information instead. These moments have semantics indeed. For instance, the 1st, 2nd and 3rd moments of distribution of atomic distances to the molecular centroid (ctd) capture the size, variance and skewness of the molecule, respectively. Selecting the first three moments provides an excellent compromise between the efficiency and the effectiveness of the method. Finally the shape similarity score of two molecules is calculated through the sum of absolute differences of their respective moments.

In this study the first three moments are computed in the same way as in ElectroShape [274], which is slightly different than the way used in USR [19, 254, 255] and USRCAT [20]. Mathematically, for a distribution of atomic distances $\{d_k\}_{k=1}^{n}$ to one of the four reference locations (ctd, cst, fct, ftf), where n is the number of atoms, the first three moments are semantically the mean, the standard deviation, and the cube root of the third central moment, respectively. Their exact expressions are shown in equations (9.1), (9.2) and (9.3). The roots are intended to provide all moments with linear space dimension in Å, unlike the skewness, for instance, which is unitless. This computation allows the distributions to contain only one sample, in which case the 2nd and 3rd moments will be zeros, i.e. $\mu_2 = \mu_3 = 0$ when $n = 1$. Likewise, when a distribution contains only two samples, the 3rd moment will be zero, i.e. $\mu_3 = 0$ when $n = 2$.

$$\mu_1 = \frac{1}{n}\sum_{k=1}^{n} d_k \tag{9.1}$$

$$\mu_2 = \sqrt[2]{\frac{1}{n}\sum_{k=1}^{n} (d_k - \mu_1)^2} \tag{9.2}$$

$$\mu_3 = \sqrt[3]{\frac{1}{n}\sum_{k=1}^{n} (d_k - \mu_1)^3} \tag{9.3}$$

After a molecule is encoded into a 12-element moment vector $\mathbf{M} = (\mu_1^{ctd}, \mu_2^{ctd}, \mu_3^{ctd}, \mu_1^{cst}, \mu_2^{cst}, \mu_3^{cst}, \mu_1^{fct}, \mu_2^{fct}, \mu_3^{fct}, \mu_1^{ftf}, \mu_2^{ftf}, \mu_3^{ftf})$, the dissimilarity score of two molecules $\mathbf{M}^i$ and $\mathbf{M}^j$ can be defined as the sum of absolute differences of their respective moments, much like the city block distance. Thereafter, this dissimilarity is monotonically inverted using equation (9.4) so as to get transformed to a normalized similarity score, where the minimum shape similarity is represented by score 0 and the maximum similarity is represented by score 1. Any other inverse monotonic function, such as cosine or L2-norm inversion, can do this transformation if it preserves the ranking order. Equation (9.4) is favored because it is simple, fast and interpretable, and has fixed upper and lower bounds.

$$S(\mathbf{M}^i, \mathbf{M}^j) = (1 + \frac{1}{12}\sum_{k=1}^{12} |\mathbf{M}_k^i - \mathbf{M}_k^j|)^{-1} \in [0, 1] \tag{9.4}$$

USR [19] is highly extensible via positioning reference loca-

tions [272, 273], incorporating higher orders of moments [270, 274], expanding coordinate dimensions [268, 274], mixing chemical component similarity [260, 261, 270], or subdividing atoms into subsets [20, 249]. USRCAT [20] extends USR [19] by separately identifying five pharmacophoric subsets of atoms, which are heavy, hydrophobic, aromatic, hydrogen bond acceptor or donor atoms. Consequently the resulting moment vector is expanded from 12 descriptors to 60, with the first 12 being identical to USR moments. In the case of an empty subset, for example if no hydrogen bond donors are found, the corresponding elements in the moment vector are set to zero. The four reference locations are uniformly derived from heavy atom coordinates and are thus meaningful with respect to the overall shape of a molecule. The five sets of 12 moments are individually scaled by the factors ow for all atoms, hw for hydrophobic atoms, rw for aromatic atoms, aw for hydrogen bond acceptors and dw for hydrogen bond donors, as shown in equation (9.5). Without a prior knowledge, the values of the five scaling factors are all defaulted to one. USRCAT degenerates to USR when $ow = 1$ and $hw = rw = aw = dw = 0$.

$$
\begin{aligned}
S(\mathbf{M}^i, \mathbf{M}^j) &= (1 \\
&+ ow \times \frac{1}{12} \sum_{k=1}^{12} |\mathbf{M}_k^i - \mathbf{M}_k^j| \\
&+ hw \times \frac{1}{12} \sum_{k=13}^{24} |\mathbf{M}_k^i - \mathbf{M}_k^j| \\
&+ rw \times \frac{1}{12} \sum_{k=25}^{36} |\mathbf{M}_k^i - \mathbf{M}_k^j| \\
&+ aw \times \frac{1}{12} \sum_{k=37}^{48} |\mathbf{M}_k^i - \mathbf{M}_k^j| \\
&+ dw \times \frac{1}{12} \sum_{k=49}^{60} |\mathbf{M}_k^i - \mathbf{M}_k^j|)^{-1} \\
&\in [0, 1] \qquad (9.5)
\end{aligned}
$$

9.4.2 USR and USRCAT on istar

Like in idock@istar for prospective structure-based virtual screening, we used the same database, which comprises 23,129,083 ligands collected from the All Clean Subset of ZINC [27, 28].

It is helpful to include more than one conformer per compound in the database since flexible molecules can adopt different shapes. Hence the more of these conformations included in the database, the less likely it is to miss molecules with the desired pattern. Each small organic molecule could have an average of about 200 conformations [254], or up to 292 conformations [263]. The conformers of a particular molecule are in

general geometrically distinct and have low potential energy, as conformers with high internal energy are in principle less likely to exist in nature.

There are numerous 2D-to-3D conversion tools that can generate 3D molecular conformations from a considered 2D chemical structure, such as Cyndi [281, 282] and OMEGA [283]. A study [284] examined the performance of four freely available small molecule conformer generation tools, Balloon [285], Confab [286], Frog2 [287], and RDKit (http://www.rdkit.org/), alongside a commercial tool, MOE (http://www.chemcomp.com/), and found that RDKit and Confab were statistically better than other methods at generating low RMSD (Root Mean Square Deviation) conformers to the known structure, and RDKit resulted as the second fastest method after Frog2. These positive results for RDKit in terms of accuracy and speed make it a valid free alternative to commercial, closed source, proprietary software.

However, even though the conformers generated by RDKit can be ensured to be at least a certain RMSD threshold apart from each other by setting an appropriate parameter, they are not guaranteed to be of low energy, so it is suggested by the manual to energy minimize them using RDKit's implementation of the Universal Force Field (UFF). After energy minimization, unfortunately, some conformers could fall into the same local energy minimum and again become structurally similar to each other with RMSD below the threshold. To resolve this conformational diversity problem, the study [284] also described a postprocessing algorithm to discriminate and keep only con-

formers which are both energy minimized and a certain RMSD threshold apart. In the postprocessing algorithm, the energy minimized conformers are first sorted by increasing energy value and the lowest energy conformer is retained, and then for each of the remaining conformers, it will be discarded when its RMSD from any conformers that have been retained is smaller than a fixed threshold d_{min}, or it will be retained otherwise. The study [284] also suggested optimal values of the number of conformers to generate (n_{conf} in equation (9.6), where n_{rot} is the number of rotatable bonds) and the d_{min} value of 0.35Å. Using RDKit in combination with the postprocessing algorithm, one can quickly build a diverse and representative set of conformers.

$$n_{conf} = \begin{cases} 50 & \text{if } n_{rot} \leq 7 \\ 200 & \text{if } 8 \leq n_{rot} \leq 12 \\ 330 & \text{otherwise} \end{cases} \tag{9.6}$$

Due to the superior accuracy and speed of RDKit, in this study we proposed to use RDKit together with the postprocessing algorithm to generate conformers. Precisely, we would use RDKit version 2014.09.2 and program against its C++ API instead of directly using the example Python script, and adopt the optimal parameter values suggested in [284]. Conformer generation is yet to be implemented in the near future considering the great efforts required, such as sufficient hard disk space.

Suppose after generating approximately 10 conformers on average for each molecule, the database would contain 230 million

conformers, each of which has a USRCAT moment vector of 60 elements of double precision floating point type, which requires 64 bits, or 8 bytes for storage. This sums up to $8B*60*230M = 108GB$ for the size of the USRCAT descriptors of all conformers in the entire database.

On one hand, given a server with sufficient capacity of memory, it is possible to preload all descriptors to enable fast screening. Mathematically, suppose t_{read} is the reading time of a moment vector of a conformer in the database, t_{score} is the scoring time of two moment vectors, n_{conf} is the number of conformers in the database, n_{query} is the number of query ligands of a single job, and n_{job} is the number of jobs, the total screening time t would be

$$t = t_{read} \times n_{conf} + t_{score} \times n_{conf} \times n_{query} \times n_{job} \tag{9.7}$$

On the other hand, on a server with insufficient memory to preload all descriptors, it is possible to load the descriptors chunk by chunk whenever a new job gets executed. Suppose only $m_{conf} << n_{conf}$ moment vectors fit in the memory, the chunk-by-chunk approach is to loop for n_{conf}/m_{conf} times, each time loading a different chunk with m_{conf} moment vectors and then making the n_{query} queries on this chunk. In this way, the total screening time t would be

$$\begin{aligned} t &= (t_{read} \times m_{conf} + t_{score} \times m_{conf} \times n_{query}) \times (n_{conf}/m_{conf}) \times n_{job} \\ &= t_{read} \times n_{conf} \times n_{job} + t_{score} \times n_{conf} \times n_{query} \times n_{job} \end{aligned} \quad (9.8)$$

As seen from the above screening time analysis, there are apparently several levels of parallelism to exploit. On the server side, there are millions of conformers, each of which has a moment vector of 60 elements. On the client side, there are multiple job submissions, each of which can contain multiple queries. In the chunk-by-chunk approach, the scoring of the current chunk and the reading of the next chunk can even be pipelined.

Considering the fundamental architecture of our istar platform [9] as well as the nature of the problem, we decided to exploit three levels of parallelism: multiple jobs are broadcast to multiple daemons running on multiple servers; multiple queries are broadcast to multiple CPU cores of a server; multiple descriptors are broadcast to multiple vector registers of a CPU core. The first two levels of parallelism are coarse grained and easy to implement, whereas the third level is relatively fine grained and requires the support of, for instance, AVX (Advanced Vector Extensions).

AVX are extensions to the x86 instruction set and expand the width of the SIMD (Single Instruction, Multiple Data) register file to 256 bits. Such a 256-bit register can hold four USR or USRCAT descriptors of 64-bit double precision floating point type. In view of the fact that a USR or USRCAT moment vector

can be expressed as $\mathbf{M} = (M_1, M_2, M_3, M_4, \ldots, M_n)$ for indexing purpose, where $n = 12$ for USR and $n = 60$ for USRCAT, this moment vector can be decomposed into groups of four elements, which are processed using AVX in a SIMD fashion as shown in Figure 9.1. Specifically, to compute the USR or USRCAT scores between two moment vectors $\mathbf{M}^i$ and $\mathbf{M}^j$ in equation (9.4), the sum of absolute differences of their respective moments must be first calculated. This can be done ideally using AVX in four steps: firstly, the four elements are subtracted; secondly, the most significant bits of the four elements are set to zeros; thirdly, the four elements undergo a horizontal addition; and lastly, the first and third elements are summed up. Note that the above four steps compute the sum of absolute differences of four respective moments only, so they are wrapped inside a loop where the number of iterations is 3 for USR and 15 for USRCAT so as to compute the sum of absolute differences of all respective moments. The calculations of USR and USRCAT scores can be merged in one loop because the first 3 iterations for USRCAT are literally also for USR (equation (9.5)). Since the loop count is constant and both the moment vectors are indexed constantly, unrolling is applicable here, resulting in a longer sequence of instructions but faster execution due to the circumvention of the use of loop iterators.

Our daemon supports a query ligand in either SDF, MOL2, XYZ, PDB or PDBQT format in one single job. Sample files in these formats are provided in the USR@istar web page. The parser on the server side is achieved by the C++ API of Open-

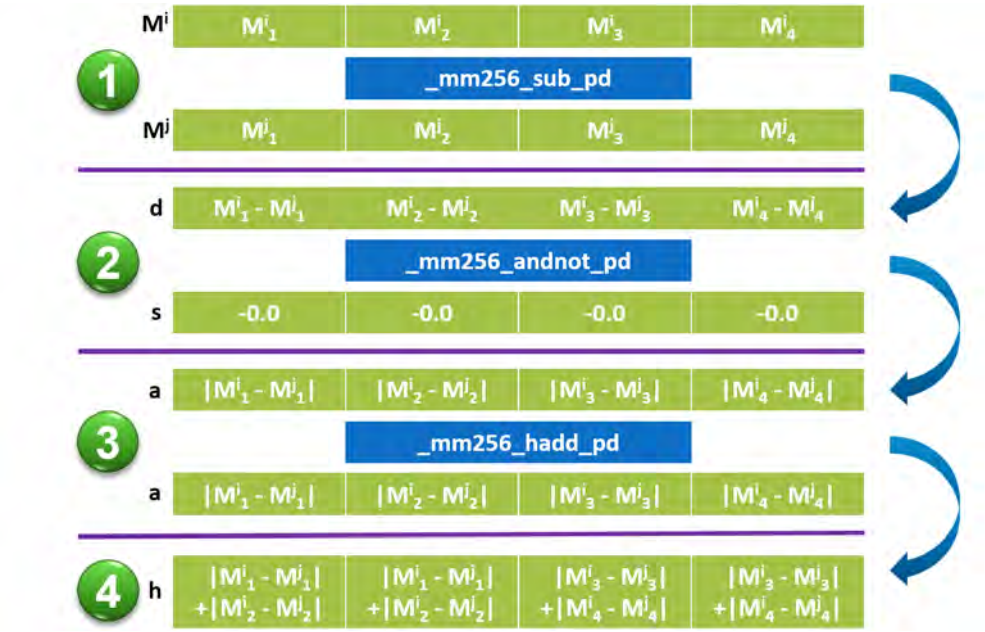


Figure 9.1: AVX instructions used to compute USR or USRCAT scores.

Babel v2.3.2 [187], while the parser on the client side is our in-house JavaScript code. Moreover, our client side user interface features our WebGL visualizer iview [11] to realize interactive visualization of high-score hits.

In the current implementation, no indexing or clustering has been done to accelerate searching. Supporting geometrical and functional clustering would be an exciting future direction.

9.5 Results and discussion

Our USR@istar requires a query ligand in 3D format as input. Thereafter, the daemon program running the background searches the entire database of 23 million compounds for structurally similar ones by USR or USRCAT score, and finally out-

Table 9.2: Molecular properties of the 19 query ligands.

ZINC ID	HA	MWT	HBD	HBA	NRB
06827693	4	60.008	0	3	0
03641271	7	103.101	4	4	0
00000882	10	135.130	3	5	0
03594299	13	176.243	4	3	1
01760831	16	206.244	0	1	0
00000163	19	277.771	0	2	2
00000931	22	314.796	2	4	1
00000706	25	332.335	0	5	0
00577115	28	382.449	1	1	5
03784182	31	411.521	0	3	4
00537755	35	476.591	2	4	7
33359785	39	536.653	4	10	3
53073961	42	588.552	1	6	10
34801951	46	650.698	4	12	7
29416466	49	671.863	7	11	13
08101127	53	751.991	0	8	14
08101051	57	806.987	4	14	9
85536932	60	835.944	3	15	15
96006018	65	914.187	3	14	6

puts the top 1000 matches. Hence, to benchmark the searching capability of USR@istar given different input, we selected 19 query ligands with different numbers of heavy atoms, spanning from 4 to 65 with a step size of 3 or 4 (Table 9.2. HA: heavy atoms. MWT: molecular weight in Daltons. HBD: hydrogen bond donors. HBA: hydrogen bond acceptors. NRB: rotatable bonds). These query ligands had a molecular weight as low as 60 Daltons and as high as 914 Daltons, covering nearly all possible input.

In the following subsections, we analyzed the output results of these 19 query ligands in order to see how the USR or USRCAT score would affect compound ranking. We then studied

how the input file format would impair pharmacophoric subset classification. Finally we benchmarked the execution time and found the performance bottleneck.

9.5.1 Matching ligands with different molecular sizes

Table 9.3 lists the top five matching compounds of the 19 selected query ligands with different molecular sizes in terms of number of heavy atoms. For clarity, the output results for different query ligands are separated by horizontal lines. Note that these query ligands were in their docked pose in PDBQT format generated by idock [9] when docked against cyclin-dependent kinase 2 (CDK2), a critical protein involved in the regulation of cell cycle transition, which will be detailed in Chapter 10. In prospective applications, the advantage of using a docked pose as input is obvious because the matching output compounds are likely to exhibit geometrically similar shape, which could retain or even enhance the intermolecular interactions and thus the putative binding affinity.

It is important to note that there is no ground truth for ranking output compounds similar to a query. The term similarity is somewhat subjective given that it might have inconsistent, or even contradictive definitions under different applications. Nevertheless, in many existing definitions, similarity can be quantified, for instance, by Tanimoto coefficient, which measures chemical similarity, or by USR score, which measures geometrical similarity.

Not unexpectedly, sorting by USR or USRCAT score yielded different output in all the 19 cases. For instance, in the case of ZINC33359785 as query, the output compound ZINC35770975, which had the highest USR score of 0.93307793, had a USRCAT score of just 0.35822866, whereas the highest USRCAT score was 0.71794128 obtained by ZINC79055171. Similarly, in the case of ZINC00577115 as input, the output compound ZINC00577115, which had the highest USRCAT score of 0.75399074, had a USR score of just 0.61248204, whereas the highest USR score was 0.95057414 obtained by ZINC68658377.

In 13 of the 19 cases, the top five matches by USR score were totally different from the top five matches by USRCAT score. This suggests that a matching compound which has a high USR score does not necessarily have a high USRCAT score, and vice versa. Particularly, in 12 of these 13 cases, the query ligand had at least 25 heavy atoms. This may indicate that USR and USRCAT tend to prioritize distinct compounds when the query ligand is large. This result was to be expected because the larger the query ligand, the larger the chemical and structural diversity of potential matching compounds.

Table 9.3: Top 5 matches of 19 query ligands

input ZINC ID	sorting method	output rank	output ZINC ID	USR score	USRCAT score
06827693	USR	1	06827693	0.99989084	0.93387099
06827693	USR	2	08214514	0.94435271	0.85958990
06827693	USR	3	08101126	0.91583558	0.90090227
06827693	USR	4	05224164	0.91323638	0.83597828
06827693	USR	5	08034818	0.91124311	0.80893830
06827693	USRCAT	1	06827693	0.99989084	0.93387099
06827693	USRCAT	2	08101126	0.91583558	0.90090227
06827693	USRCAT	3	01846598	0.82101487	0.86076010
06827693	USRCAT	4	08214514	0.94435271	0.85958990
06827693	USRCAT	5	04658552	0.77735039	0.85427671
03641271	USR	1	19737051	0.98779656	0.82477674
03641271	USR	2	34689286	0.98720924	0.82948216
03641271	USR	3	04262127	0.98577986	0.83173477
03641271	USR	4	05226936	0.98436293	0.77590893
03641271	USR	5	05226942	0.98419217	0.77576925
03641271	USRCAT	1	04417022	0.90277096	0.95866631
03641271	USRCAT	2	32296878	0.89128565	0.92383824
03641271	USRCAT	3	64033578	0.91094937	0.91829554
03641271	USRCAT	4	04658602	0.89124510	0.91780849
03641271	USRCAT	5	01666720	0.89345170	0.91692833

Table 9.3 – *Continued from previous page*

input ZINC ID	sorting method	output rank	output ZINC ID	USR score	USRCAT score
00000882	USR	1	00000882	0.99992602	0.99993121
00000882	USR	2	26995176	0.99597115	0.80645548
00000882	USR	3	72231352	0.99543539	0.82250290
00000882	USR	4	82410931	0.99508429	0.81322159
00000882	USR	5	01615910	0.99476961	0.85551400
00000882	USRCAT	1	00000882	0.99992602	0.99993121
00000882	USRCAT	2	08616408	0.99327569	0.98073897
00000882	USRCAT	3	03652180	0.98439837	0.93952496
00000882	USRCAT	4	18153302	0.97495298	0.93319108
00000882	USRCAT	5	13516924	0.99128475	0.92503209
03594299	USR	1	03594299	0.99978454	0.99983624
03594299	USR	2	86639023	0.94249238	0.56539305
03594299	USR	3	23093521	0.94248914	0.67187858
03594299	USR	4	63169716	0.94098603	0.56936544
03594299	USR	5	23093522	0.93999049	0.67166081
03594299	USRCAT	1	03594299	0.99978454	0.99983624
03594299	USRCAT	2	22219232	0.89835884	0.86306679
03594299	USRCAT	3	86051862	0.89218389	0.85055658
03594299	USRCAT	4	38532749	0.81227537	0.84324650
03594299	USRCAT	5	19796009	0.84131930	0.84309403

Table 9.3 – *Continued from previous page*

input ZINC ID	sorting method	output rank	output ZINC ID	USR score	USRCAT score
01760831	USR	1	01760831	0.99987949	0.99989913
01760831	USR	2	01709154	0.98699243	0.75851222
01760831	USR	3	00971553	0.98535056	0.81526150
01760831	USR	4	67730206	0.98262917	0.75827336
01760831	USR	5	01681580	0.97753801	0.94530750
01760831	USRCAT	1	01760831	0.99987949	0.99989913
01760831	USRCAT	2	01681580	0.97753801	0.94530750
01760831	USRCAT	3	02510715	0.89837623	0.91571639
01760831	USRCAT	4	01555275	0.88135318	0.90937875
01760831	USRCAT	5	39255067	0.89111121	0.90652744
00000163	USR	1	00000163	0.96426028	0.95981456
00000163	USR	2	95003095	0.96059429	0.46005568
00000163	USR	3	94998736	0.95953442	0.45932022
00000163	USR	4	93855398	0.95647154	0.56896132
00000163	USR	5	94468575	0.94929677	0.52144044
00000163	USRCAT	1	00000163	0.96426028	0.95981456
00000163	USRCAT	2	12336856	0.88870156	0.87618989
00000163	USRCAT	3	72196442	0.85417800	0.86199583
00000163	USRCAT	4	03830577	0.89135269	0.84555906
00000163	USRCAT	5	71767467	0.79282467	0.84492369

Table 9.3 – *Continued from previous page*

input ZINC ID	sorting method	output rank	output ZINC ID	USR score	USRCAT score
00000931	USR	1	33693318	0.96997742	0.78246622
00000931	USR	2	00000931	0.96952854	0.98807569
00000931	USR	3	66774020	0.96744933	0.69240859
00000931	USR	4	66773996	0.96728672	0.70565997
00000931	USR	5	66546099	0.96669849	0.55832601
00000931	USRCAT	1	00000931	0.96952854	0.98807569
00000931	USRCAT	2	05284765	0.89184694	0.84417333
00000931	USRCAT	3	02193124	0.93922432	0.83616846
00000931	USRCAT	4	36020480	0.88785134	0.82437481
00000931	USRCAT	5	05011019	0.90642740	0.82042605
00000706	USR	1	65405747	0.95996452	0.44442734
00000706	USR	2	40647363	0.95682962	0.50111399
00000706	USR	3	93828780	0.95053992	0.40343405
00000706	USR	4	72255543	0.95023716	0.54229924
00000706	USR	5	00408247	0.94828659	0.51394696
00000706	USRCAT	1	80448655	0.87617333	0.83086022
00000706	USRCAT	2	78886065	0.77618234	0.82713956
00000706	USRCAT	3	93775002	0.73152053	0.82414472
00000706	USRCAT	4	78886066	0.76913264	0.82370563
00000706	USRCAT	5	78700739	0.89428224	0.81951128

Table 9.3 – *Continued from previous page*

input ZINC ID	sorting method	output rank	output ZINC ID	USR score	USRCAT score
00577115	USR	1	68658377	0.95057414	0.39005121
00577115	USR	2	69389189	0.94579839	0.35499414
00577115	USR	3	89143595	0.94491604	0.46345807
00577115	USR	4	78399755	0.94491250	0.45419415
00577115	USR	5	83027211	0.94480002	0.34078650
00577115	USRCAT	1	22657173	0.61248204	0.75399074
00577115	USRCAT	2	01550499	0.56107197	0.74185012
00577115	USRCAT	3	11678289	0.55772743	0.73271990
00577115	USRCAT	4	11662947	0.82521725	0.72450314
00577115	USRCAT	5	08474264	0.84212073	0.72433086
03784182	USR	1	67249641	0.94940522	0.45316801
03784182	USR	2	79160658	0.94444038	0.37397556
03784182	USR	3	66798033	0.94393213	0.41624736
03784182	USR	4	78080377	0.94104867	0.41947726
03784182	USR	5	44148945	0.93967079	0.39586261
03784182	USRCAT	1	03784182	0.77235330	0.83985811
03784182	USRCAT	2	71789431	0.74342618	0.80805432
03784182	USRCAT	3	93106652	0.73942307	0.66665953
03784182	USRCAT	4	93106649	0.73848385	0.66623719
03784182	USRCAT	5	90800223	0.64094384	0.66486802

Table 9.3 – *Continued from previous page*

input ZINC ID	sorting method	output rank	output ZINC ID	USR score	USRCAT score
00537755	USR	1	33077240	0.92816370	0.46838130
00537755	USR	2	71880803	0.92703459	0.44320682
00537755	USR	3	10246617	0.92144736	0.47914766
00537755	USR	4	34759207	0.91737385	0.43123165
00537755	USR	5	03223873	0.91734806	0.50177117
00537755	USRCAT	1	65623842	0.82911423	0.73050138
00537755	USRCAT	2	65623840	0.82920422	0.73043065
00537755	USRCAT	3	65623844	0.84434115	0.72502994
00537755	USRCAT	4	65623838	0.84429533	0.72498612
00537755	USRCAT	5	91738997	0.80098542	0.72414875
33359785	USR	1	35770975	0.93307793	0.35822866
33359785	USR	2	72004193	0.93114334	0.38182298
33359785	USR	3	33435896	0.92965520	0.33894654
33359785	USR	4	09254874	0.92781816	0.40359101
33359785	USR	5	12807677	0.92642912	0.35213376
33359785	USRCAT	1	79055171	0.84112415	0.71794128
33359785	USRCAT	2	71923062	0.73844740	0.71181372
33359785	USRCAT	3	77341431	0.75924711	0.70930088
33359785	USRCAT	4	77341441	0.75921732	0.70927376
33359785	USRCAT	5	69046249	0.70287350	0.70695212

Table 9.3 – *Continued from previous page*

input ZINC ID	sorting method	output rank	output ZINC ID	USR score	USRCAT score
53073961	USR	1	03280771	0.89422462	0.55998626
53073961	USR	2	12509920	0.89370657	0.48547850
53073961	USR	3	34899971	0.89334569	0.59114533
53073961	USR	4	15730136	0.89292055	0.47162568
53073961	USR	5	09451077	0.89287846	0.40216378
53073961	USRCAT	1	20677274	0.82865029	0.73844702
53073961	USRCAT	2	08916625	0.80980780	0.72773540
53073961	USRCAT	3	02110104	0.80092701	0.72556402
53073961	USRCAT	4	20685629	0.84494018	0.71625164
53073961	USRCAT	5	53064465	0.81539807	0.71344270
34801951	USR	1	09813531	0.86662080	0.49260706
34801951	USR	2	10120418	0.85012639	0.48564947
34801951	USR	3	22633339	0.84690548	0.32109534
34801951	USR	4	09942653	0.83253562	0.41993991
34801951	USR	5	09016590	0.83239022	0.44123198
34801951	USRCAT	1	09730878	0.78365470	0.65931372
34801951	USRCAT	2	02952637	0.69990120	0.63865229
34801951	USRCAT	3	03066292	0.79377130	0.63427123
34801951	USRCAT	4	63600618	0.78591726	0.62978556
34801951	USRCAT	5	02171971	0.81658623	0.62773804

Table 9.3 – *Continued from previous page*

input ZINC ID	sorting method	output rank	output ZINC ID	USR score	USRCAT score
29416466	USR	1	63514587	0.91665219	0.46731155
29416466	USR	2	40292843	0.91468903	0.47087614
29416466	USR	3	40293083	0.91077708	0.45683873
29416466	USR	4	22052805	0.90502904	0.50124716
29416466	USR	5	22052817	0.90013261	0.56632489
29416466	USRCAT	1	13389101	0.77024906	0.69218386
29416466	USRCAT	2	67741752	0.71934043	0.68237936
29416466	USRCAT	3	67769739	0.71092297	0.67849718
29416466	USRCAT	4	27016688	0.75586732	0.66575518
29416466	USRCAT	5	27016682	0.73442306	0.66531265
08101127	USR	1	40253466	0.82049130	0.45823208
08101127	USR	2	40251780	0.81389665	0.44709474
08101127	USR	3	40253347	0.81270120	0.45688630
08101127	USR	4	40252264	0.81007384	0.43212674
08101127	USR	5	40252260	0.80935988	0.42814493
08101127	USRCAT	1	40204804	0.70216986	0.61769941
08101127	USRCAT	2	40203918	0.72134056	0.61467790
08101127	USRCAT	3	40198912	0.75040459	0.61092978
08101127	USRCAT	4	19788980	0.67787440	0.60600169
08101127	USRCAT	5	40203919	0.72535625	0.60578330

Table 9.3 – *Continued from previous page*

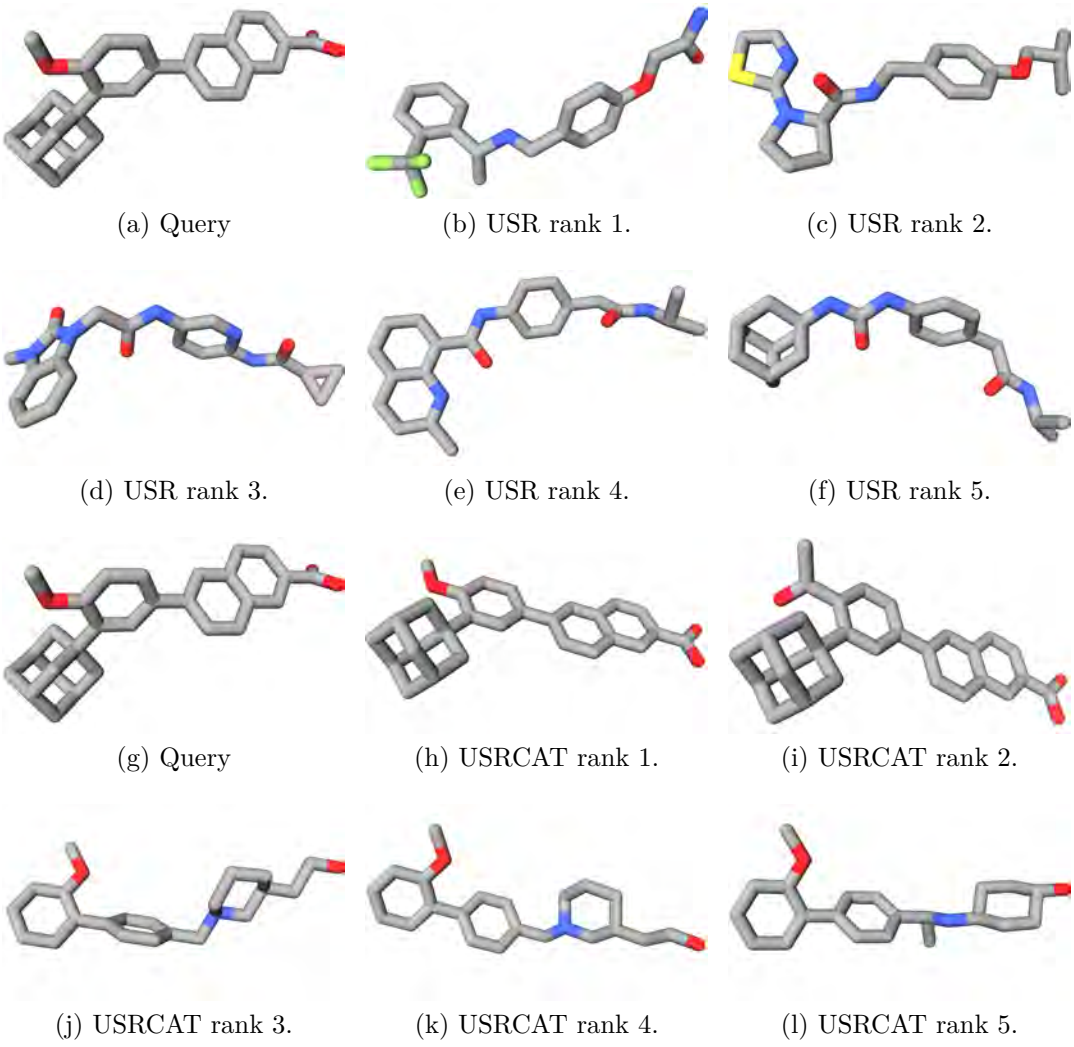
input ZINC ID	sorting method	output rank	output ZINC ID	USR score	USRCAT score
08101051	USR	1	34906112	0.81662557	0.38601073
08101051	USR	2	12790633	0.81216532	0.38263829
08101051	USR	3	34905308	0.81088357	0.37345643
08101051	USR	4	34906092	0.80499926	0.37499018
08101051	USR	5	22623297	0.80282387	0.42624504
08101051	USRCAT	1	91742856	0.62507638	0.63946406
08101051	USRCAT	2	63634534	0.68412199	0.63451306
08101051	USRCAT	3	63634530	0.66760438	0.63182588
08101051	USRCAT	4	12530576	0.59249267	0.62763900
08101051	USRCAT	5	08872621	0.63481592	0.62422077
85536932	USR	1	38491099	0.87113381	0.54465979
85536932	USR	2	33767419	0.86880460	0.51486740
85536932	USR	3	12304689	0.86865297	0.56751118
85536932	USR	4	00903819	0.85511867	0.44597602
85536932	USR	5	12706271	0.85465720	0.44716140
85536932	USRCAT	1	63386406	0.72871316	0.62348129
85536932	USRCAT	2	14232016	0.70572357	0.61247207
85536932	USRCAT	3	38573815	0.72269606	0.61188087
85536932	USRCAT	4	09713729	0.78048080	0.60555170
85536932	USRCAT	5	26998174	0.74923801	0.59966269

Table 9.3 – *Continued from previous page*

input ZINC ID	sorting method	output rank	output ZINC ID	USR score	USRCAT score
96006018	USR	1	40200504	0.83259667	0.31658414
96006018	USR	2	35402146	0.79559059	0.35874467
96006018	USR	3	10462892	0.78614715	0.34836142
96006018	USR	4	09693915	0.78488061	0.33538157
96006018	USR	5	40206470	0.78111193	0.30224723
96006018	USRCAT	1	77320087	0.65583149	0.61433137
96006018	USRCAT	2	31392738	0.64614083	0.58921551
96006018	USRCAT	3	04536260	0.60279466	0.57727754
96006018	USRCAT	4	89960170	0.47860515	0.57257619
96006018	USRCAT	5	66730926	0.48114189	0.56721396

Figures 9.2 and 9.3 plot the 3D poses of two query ligands, ZINC03784182 and ZINC00537755, respectively, as well as their top 5 matching compounds using iview [11]. These two query ligands were selected for visualization because their molecular weight (Table 9.2) is in the range of candidate leads, so they are of high chance of being selected as query ligands in real case studies. Note that the output compounds were not structurally aligned to the query.

Next, we examined the capability of ranking high the particular compound in the output with an identical ZINC ID as the input query. Apparently, such recovery test requires that the query compound must be present in the target database, although it can be in a different pose. Out of the 19 query ligands,



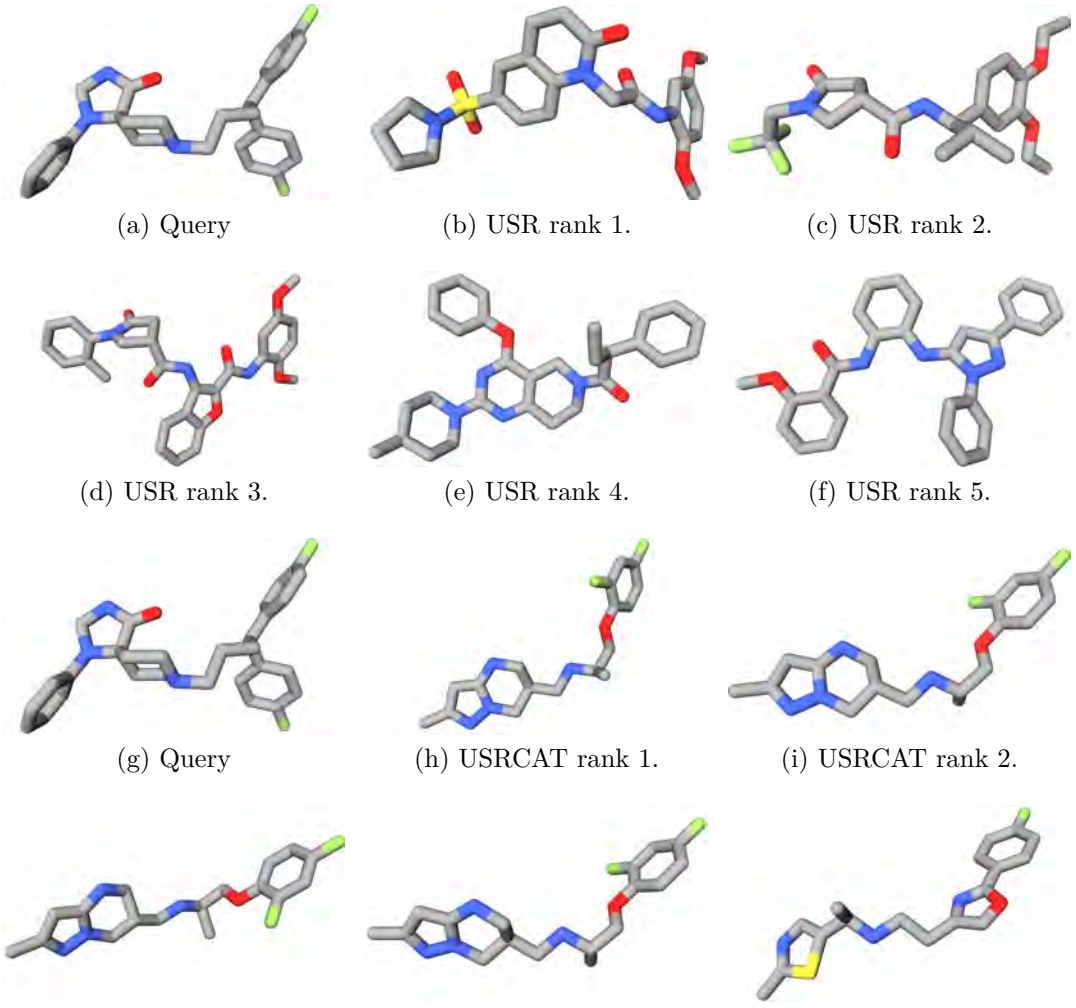
(a) Query (b) USR rank 1. (c) USR rank 2.

(d) USR rank 3. (e) USR rank 4. (f) USR rank 5.

(g) Query (h) USRCAT rank 1. (i) USRCAT rank 2.

(j) USRCAT rank 3. (k) USRCAT rank 4. (l) USRCAT rank 5.

Figure 9.2: Top 5 matching compounds for ZINC03784182 (a & g) using USR (b to f) or USRCAT (h to l).



(a) Query (b) USR rank 1. (c) USR rank 2.

(d) USR rank 3. (e) USR rank 4. (f) USR rank 5.

(g) Query (h) USRCAT rank 1. (i) USRCAT rank 2.

(j) USRCAT rank 3. (k) USRCAT rank 4. (l) USRCAT rank 5.

Figure 9.3: Top 5 matching compounds for ZINC00537755 (a & g) using USR (b to f) or USRCAT (h to l).

9 satisfied this requirement (Table 9.4). Note again that the query ligands were in their docked pose while the compounds in the molecular database were in their low energy pose, therefore the corresponding compound with an identical ZINC ID as the input query was not always guaranteed to be ranked the first in the output. Interestingly, the recovery rate seemed to correlate with NRB (number of rotatable bonds). When NRB was zero, both USR and USRCAT ranked the input compound the highest in the output. When NRB was 1 or 2, the recovery rate started to drop slightly for USR. When NRB was equal to or beyond 4, both methods had difficulty in prioritizing the input compound in the output. This observation was to be expected because USR and USRCAT are dependent on torsions, though independent of spatial position and orientation. When a compound has a NRB of zero, there is only one possible conformation, so the docked pose of the query must be conformationally equivalent to the original pose present in the molecular database. On the other hand, when a compound has a large NRB, there is a high chance that the docked pose and the original pose differ remarkably in their torsions, and so are their encoded features.

9.5.2 Impact of file format on pharmacophoric subset classification

It was anticipated that the same query ligand should yield identical output regardless of its input file format, be it sdf, mol2 or pdbqt. Surprisingly, we found that this was not necessarily the

Table 9.4: Output ranking of the same input compound in a different pose.

input ZINC ID	NRB	USR rank of the same input compound	USRCAT rank of the same input compound
06827693	0	1	1
00000882	0	1	1
01760831	0	1	1
03594299	1	1	1
00000931	1	2	1
00000163	2	1	1
03784182	4	>1000	1
00577115	5	>1000	>1000
00537755	7	>1000	>1000

case. We used as an example ZINC00537755 in its low energy pose (Figure 9.4) rather than in its docked pose in order to retrieve a USR score and a USRCAT score of exactly one. When the query ligand was in pdbqt format, the same format used to encode the molecular database, the corresponding output compound had USR and USRCAT scores of exactly one and was hence ranked the first. Nevertheless, for the same query, if the format was changed to sdf or mol2, the corresponding output compound turned out to have a USR score of 0.99984619 and a USRCAT score of just 0.81023940, though it was still ranked the first (Table 9.5).

The large deviation in USRCAT score from an expected value of one attracted our attention. After careful investigations, we were surprised to find that the N3 atom, covalently connected to a polar hydrogen and therefore supposed to be a hydrogen bond donor (Figure 9.4), failed to be recognized so in the SMARTS matching operation by OpenBabel [187] when the query was in pdbqt format. This certainly had a great impact on the score of

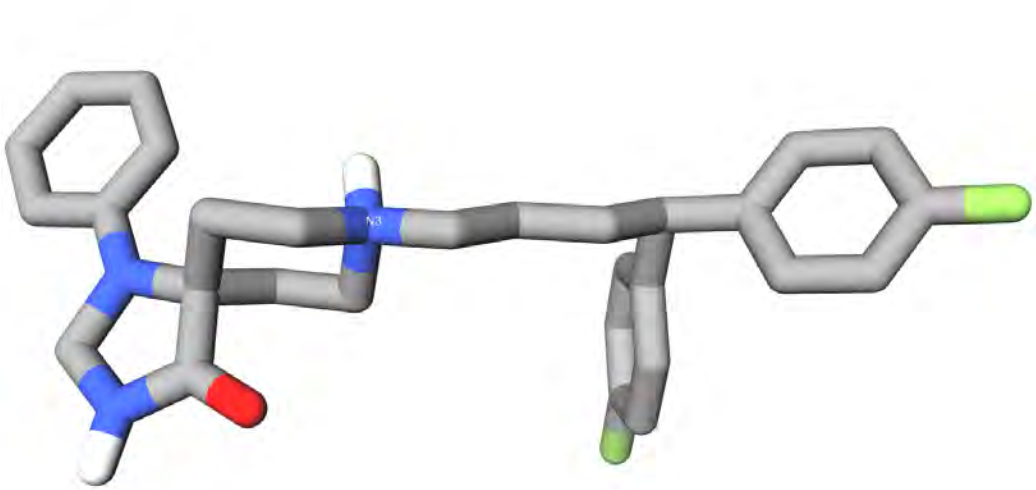


Figure 9.4: ZINC00537755 with the N3 atom labeled.

Table 9.5: Top 5 matches of ZINC00537755 in pdbqt or sdf/mol2 format.

input ZINC ID	input format	output rank	output ZINC ID	USR score	USRCAT score
00537755	pdbqt	1	00537755	1.00000000	1.00000000
00537755	pdbqt	2	91686384	0.81962803	0.66995207
00537755	pdbqt	3	09254146	0.80012011	0.66812359
00537755	pdbqt	4	22804726	0.76476137	0.66203173
00537755	pdbqt	5	22919990	0.69273531	0.66197772
00537755	sdf/mol2	1	00537755	0.99984619	0.81023940
00537755	sdf/mol2	2	89286333	0.71411555	0.65628724
00537755	sdf/mol2	3	89286335	0.71272183	0.65581047
00537755	sdf/mol2	4	26008225	0.85084668	0.65553656
00537755	sdf/mol2	5	67878453	0.64157793	0.65528239

USRCAT, which requires classification of atoms into five predefined pharmacophoric subsets prior to moment calculation.

Such misclassification of a certain pharmacophoric subset other than heavy atoms should not affect USR score, which merely replies on heavy atoms to calculate the features. The USR score for the query in sdf/mol2 format was somewhat less than one because the coordinates in sdf/mol2 format have 4 decimal digits, whereas those in pdbqt format have 3 decimal digits.

9.5.3 Execution time

For the 19 selected queries, we inserted additional code to the daemon to record their execution time on the server equipped with Intel Xeon W3520 and 16GB ECC DDR3. Their execution times were averaged to 167 seconds and were quite consistent with a small standard deviation across the 19 query ligands (Table 9.6) regardless of their molecular size in terms of number of heavy atoms, molecular weight, or number of rotatable bonds.

We further measured to what extent preloading the precalculated USRCAT features of the 23 million compounds would help to shorten the query time. The size of entire features is $8B * 60 * 23M = 10GB$. These features can be loaded from hard disk *ad hoc* or in advance. In the latter case, there are two additional one-off steps, memory allocation and file reading. Memory allocation took 5 seconds, which could be longer in case of insufficient free memory and therefore requirement of spare capacity from hard disk swapping. File reading from hard

Table 9.6: Execution time in seconds of the 19 queries when the USRCAT features were loaded *ad hoc* or in advance.

ZINC ID	*ad hoc* (s)	in advance (s)
06827693	171	34
03641271	165	20
00000882	172	29
03594299	173	30
01760831	172	30
00000163	173	29
00000931	164	31
00000706	163	29
00577115	163	30
03784182	165	30
00537755	167	32
33359785	166	28
53073961	167	31
34801951	165	31
29416466	166	32
08101127	161	27
08101051	164	31
85536932	164	30
96006018	164	30
Average	167	30
Std dev	4	3

disk took another 149 seconds. Subsequently during USR and USRCAT score calculation, it required just 30 seconds to complete a query on average ($t_{score} \times n_{conf}$ in equations (9.7) and (9.8)), compared to 167 seconds when the features were loaded *ad hoc*. In other words, the pairwise similarity score computation accounted for only 18% (=30/167=$t_{score}/(t_{read} + t_{score})$) of the matching time, while the majority 82% time was spent in file reading. This suggests that job execution is IO bound, hence deploying the daemon on a server with sufficient memory or a fast solid state drive helps to reduce query time significantly. The query time of 30 seconds can be possibly furtuer shortened by proper structural clustering and indexing in the future to prune dispensable moment vector matching.

9.6 Conclusions

Searching for compounds that resemble the shape of a given query molecule is a widely seen yet daunting problem in ligand-based virtual screening [140, 254–258] and macromolecular target prediction [260–262]. The USR-like methods [19, 20, 268] represent an entirely new class of non-superposition algorithms that effectively capture the molecular shape information independent of spatial position and orientation. These methods circumvent the requirement of structural alignment and show outstanding computational efficiency with respect to superposition-based methods [264–266].

In this study we have reviewed the traditional USR method

[19] and its various extensions [20, 249, 260, 261, 268, 270–274], as well as their applications, both retrospective [20, 254] and prospective [140, 255–258, 275]. We have highlighted the pros and cons of three web servers [249, 261, 271] which use USR variants as their underlying methods. Consequently, to address the existing constraints, we have designed our pragmatic implementation of USR [19] and USRCAT [20] on istar [9] and explained its methodological advancement in terms of functionality, usability, and efficiency.

First and foremost, our molecular database has been populated with more than 23 million small and diverse molecules that are collected from the ZINC database [27, 28] and thus possibly commercially available to purchase for subsequent biological assays. We have also proposed to generate low-energy conformers that are likely to occur in nature. The reason why a molecular database should be populated with multiple conformers of each flexible compound is to reduce the possibility of missing compounds with similar shape to the query. We would use RDKit and the postprocessing algorithm suggested in [284] for the conformer generation task in the near future. The USR and USRCAT features of these 23 million compounds have been precalculated, which was a one-off exercise.

Second, we have estimated the storage size of all descriptors across the entire database, and suggested two approaches, i.e. preloading all descriptors at once and loading them chunk by chunk, and analyzed their theoretical execution time. Based on the time analysis, we have exploited three levels of parallelism,

which map multiple jobs to multiple servers, multiple queries to multiple CPU cores, and multiple descriptors to multiple CPU registers, respectively, to fully utilize all available computational resources so as to accelerate job execution. Notably, we have described a novel AVX implementation of sum of absolute differences to calculate the USR or USRCAT scores between two moment vectors.

Moreover, our implementation of USR and USRCAT, denoted as USR@istar for short, supports a query ligand in SDF, MOL2, XYZ, PDB or PDBQT format. It also features our iview [11] WebGL visualizer to aid result interpretation in an interactive manner.

To benchmark USR@istar comprehensively, we selected 19 query ligands with different numbers of heavy atoms. To our expectation, sorting by USR or USRCAT score yielded different output, especially when the input ligand was large. From another perspective, when NRB was large, both methods did not manage to recover the query ligand in a different pose in the output because of the large conformational diversity implied by a large NRB. To our surprise, input file format impaired the classification of atoms into predefined pharmacophoric subsets, probably due to a bug in OpenBabel [187]. We have also found that preloading database features boosted matching performance by four times.

We believe our USR@istar web service for ligand-based virtual screening purpose can perfectly supplement our idock@istar web service for structure-based virtual screening purpose. We

suggest users try both services to reach a consensus. In the future we would provide a hybrid service that incorporates the advantages of idock@istar and USR@istar.

9.7 Availability

USR@istar is free and open source under Apache License 2.0. It is available as a module of istar [9]. Our deployment of USR@istar is running at http://istar.cse.cuhk.edu.hk/usr. Sample query files are provided therein.

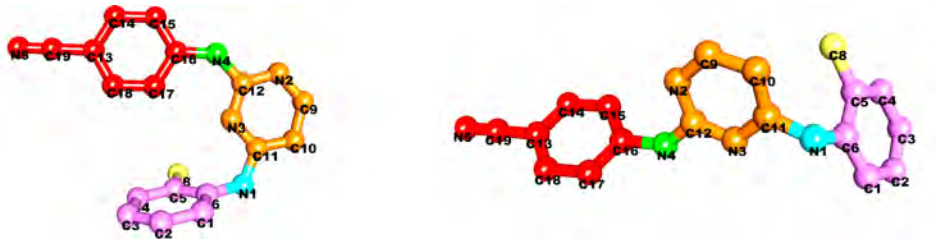
9.8 Future works

Although USR is advantageous in being independent of position and orientation, it is dependent on torsions. Likewise, none of the surveyed USR extensions are invariant of torsions. Therefore multiple conformers must be generated for each ligand in a large database in order to represent conformational diversity to some extent. The more conformers that are to generate, the higher degree in which the conformational space will be covered. So there is a tradeoff between database storage size and conformational diversity exhaustiveness.

We present USRT (Ultrafast Shape Recognition with Torsions), the first USR-like algorithm that can identify different conformations of the same ligand. In other words, different conformers generated from the same ligand will result in identical USRT descriptors. This is a huge advantage over existing meth-

ods because it circumvents the task of conformer generation for a large database, leading to greatly reduced storage requirement. Moreover, it covers the entire conformational space spanned by all conformations of a ligand. Since no conformer generation is required, there are no more considerations of whether to use the bound or unbound conformations of a bioactive molecule even though the two conformations could be in principle significantly different. Apart from the circumvention of conformer generation, other applications of USRT include the detection of duplicate ligands in virtual screening campaigns [269] or in *de novo* fragment-based drug design [12, 13], and ligand clustering [254, 263].

The methodological idea of USRT is quite straightforward. Compared to UFSRAT [249] and USRCAT [20], instead of subdividing atoms into subsets according to pharmacophoric properties, USRT subdivides atoms into subsets according to branches that are connected via rotatable bonds. Figure 9.5 illustrates two conformations of the same ligand with different torsions. Figures 9.6 and 9.7 show the underlying PDBQT contents. The ligand has five rotatable bonds. The atoms and bonds in the same branch are rendered in the same color. For each branch, the only reference atom is chosen to be the atom connecting the parent branch, i.e. the second atom involved in the rotatable bond or the Y atom in the line of "BRANCH X Y". It is always the first atom of the current branch if the PDBQT file is produced by AutoDockTools4 [32]. For the rigid root where there is no parent, the only reference atom is simply chosen to



(a) Structure of the 1st conformation. (b) Structure of the 2nd conformation.

Figure 9.5: Two different conformations of the same ligand, with branches highlighted in separate colors.

be the first atom. Next, the atomic distance distributions are computed between the atoms of the current branch and the corresponding reference atom. In this way, the relative positional information within a rigid branch is captured regardless of the torsions introduced by, say, flexible ligand docking. The remaining steps remain intact, where the first three moments are calculated from the atomic distance distribution of each branch.

Like other USR extensions, USRT also inherits the major advantages from USR such as being ultrafast and extensible, and avoiding the need of aligning molecules before testing for similarity. USRT can be possibly combined with other USR variants [20, 268, 270] for different applications.

Although USRT seems very attractive, there are some major problems to solve before USRT becomes practically useful. The length of the resulting moment vector is proportional to one plus the number of branches, which is a variable. This raises the obvious question of how to compare molecules with different numbers of rotatable bonds. One possible solution is to output the descriptors in a tree structure rather than in a linear vector,

```
ROOT
ATOM      1  C13 T27 A 557      49.799 -31.025  35.312  1.00 25.09     0.044 A
ATOM      2  C14 T27 A 557      50.269 -31.013  33.977  1.00 23.67    -0.061 A
ATOM      3  H14 T27 A 557      50.154 -31.913  33.349  1.00  0.00     0.085 H
ATOM      4  C15 T27 A 557      50.883 -29.861  33.441  1.00 27.38    -0.055 A
ATOM      5  H15 T27 A 557      51.243 -29.877  32.399  1.00  0.00     0.087 H
ATOM      6  C16 T27 A 557      51.048 -28.673  34.221  1.00 27.55     0.049 A
ATOM      7  C17 T27 A 557      50.561 -28.702  35.570  1.00 21.87    -0.055 A
ATOM      8  H17 T27 A 557      50.671 -27.804  36.201  1.00  0.00     0.087 H
ATOM      9  C18 T27 A 557      49.940 -29.861  36.113  1.00 27.25    -0.061 A
ATOM     10  H18 T27 A 557      49.570 -29.855  37.152  1.00  0.00     0.085 H
ATOM     11  N5  T27 A 557      48.698 -33.181  36.272  1.00 26.72    -0.191 NA
ATOM     12  C19 T27 A 557      49.186 -32.213  35.845  1.00 29.50     0.099 C
ENDROOT
BRANCH   6  13
ATOM     13  N4  T27 A 557      51.661 -27.501  33.717  1.00 25.92    -0.192 N
ATOM     14  H4  T27 A 557      51.728 -27.498  32.699  1.00  0.00     0.184 HD
BRANCH  13  15
ATOM     15  C12 T27 A 557      52.195 -26.349  34.296  1.00 26.30     0.684 A
ATOM     16  N3  T27 A 557      51.982 -26.078  35.581  1.00 22.76    -0.176 N
ATOM     17  C11 T27 A 557      52.499 -24.952  36.144  1.00 25.85     0.122 A
ATOM     18  H3  T27 A 557      51.427 -26.719  36.148  1.00  0.00     0.186 HD
ATOM     19  C10 T27 A 557      53.261 -24.038  35.410  1.00 25.43    -0.025 A
ATOM     20  C9  T27 A 557      53.448 -24.380  34.061  1.00 23.08    -0.002 A
ATOM     21  H10 T27 A 557      53.682 -23.121  35.855  1.00  0.00     0.091 H
ATOM     22  N2  T27 A 557      52.922 -25.523  33.487  1.00 25.29    -0.202 N
ATOM     23  H9  T27 A 557      54.045 -23.703  33.427  1.00  0.00     0.112 H
ATOM     24  H2  T27 A 557      53.071 -25.740  32.501  1.00  0.00     0.183 HD
BRANCH  17  25
ATOM     25  N1  T27 A 557      52.219 -24.743  37.509  1.00 19.93    -0.341 N
ATOM     26  H1  T27 A 557      52.727 -24.015  38.011  1.00  0.00     0.167 HD
BRANCH  25  27
ATOM     27  C6  T27 A 557      51.256 -25.511  38.206  1.00 22.06     0.045 A
ATOM     28  C5  T27 A 557      51.633 -26.804  38.759  1.00 27.85    -0.034 A
ATOM     29  C4  T27 A 557      50.673 -27.581  39.439  1.00 26.06    -0.065 A
ATOM     30  C3  T27 A 557      49.357 -27.139  39.629  1.00 25.95    -0.035 A
ATOM     31  H4  T27 A 557      50.967 -28.568  39.834  1.00  0.00     0.085 H
ATOM     32  C2  T27 A 557      48.975 -25.867  39.102  1.00 24.94    -0.065 A
ATOM     33  C1  T27 A 557      49.920 -25.039  38.401  1.00 29.53    -0.034 A
ATOM     34  H2  T27 A 557      47.938 -25.514  39.235  1.00  0.00     0.085 H
BRANCH  28  35
ATOM     35  C8  T27 A 557      53.020 -27.393  38.614  1.00 23.50    -0.049 C
ATOM     36  H83 T27 A 557      53.412 -27.671  39.620  1.00  0.00     0.033 H
ATOM     37  H82 T27 A 557      52.949 -28.397  38.135  1.00  0.00     0.033 H
ATOM     38  H81 T27 A 557      53.779 -26.779  38.076  1.00  0.00     0.033 H
ENDBRANCH  28  35
ENDBRANCH  25  27
ENDBRANCH  17  25
ENDBRANCH  13  15
ENDBRANCH   6  13
TORSDOF 5
```

Figure 9.6: PDBQT content of the 1st conformation.

```
ROOT
ATOM      1  C13 T27 A 557      47.272 -29.026  39.801  1.00 25.09     0.538 A
ATOM      2  C14 T27 A 557      47.742 -29.014  38.466  1.00 23.67     2.969 A
ATOM      3  H14 T27 A 557      47.627 -29.914  37.838  1.00  0.00     0.000 H
ATOM      4  C15 T27 A 557      48.356 -27.862  37.930  1.00 27.38     1.876 A
ATOM      5  H15 T27 A 557      48.716 -27.878  36.888  1.00  0.00     0.000 H
ATOM      6  C16 T27 A 557      48.521 -26.674  38.710  1.00 27.55     0.473 A
ATOM      7  C17 T27 A 557      48.034 -26.703  40.059  1.00 21.87    -0.866 A
ATOM      8  H17 T27 A 557      48.144 -25.805  40.690  1.00  0.00     0.000 H
ATOM      9  C18 T27 A 557      47.413 -27.862  40.602  1.00 27.25    -0.619 A
ATOM     10  H18 T27 A 557      47.043 -27.856  41.641  1.00  0.00     0.000 H
ATOM     11  N5  T27 A 557      46.171 -31.182  40.761  1.00 26.72    17.532 NA
ATOM     12  C19 T27 A 557      46.659 -30.214  40.334  1.00 29.50     6.094 C
ENDROOT
BRANCH   6  13
ATOM     13  N4  T27 A 557      49.134 -25.502  38.206  1.00 25.92    -0.349 N
ATOM     14  H4  T27 A 557      50.068 -25.351  38.587  1.00  0.00     0.000 HD
BRANCH  13  15
ATOM     15  C12 T27 A 557      48.750 -24.506  37.306  1.00 26.30    -0.011 A
ATOM     16  N3  T27 A 557      49.452 -23.380  37.206  1.00 22.76     0.289 N
ATOM     17  C11 T27 A 557      49.076 -22.407  36.332  1.00 25.85     3.493 A
ATOM     18  H3  T27 A 557      50.278 -23.246  37.789  1.00  0.00     0.000 HD
ATOM     19  C10 T27 A 557      47.948 -22.539  35.515  1.00 25.43     4.413 A
ATOM     20  C9  T27 A 557      47.251 -23.748  35.669  1.00 23.08     2.795 A
ATOM     21  H10 T27 A 557      47.632 -21.756  34.806  1.00  0.00     0.000 H
ATOM     22  N2  T27 A 557      47.631 -24.737  36.558  1.00 25.29     1.340 N
ATOM     23  H9  T27 A 557      46.354 -23.918  35.050  1.00  0.00     0.000 H
ATOM     24  H2  T27 A 557      47.099 -25.603  36.651  1.00  0.00     0.000 HD
BRANCH  17  25
ATOM     25  N1  T27 A 557      49.874 -21.247  36.296  1.00 19.93     3.521 N
ATOM     26  H1  T27 A 557      49.540 -20.404  36.762  1.00  0.00     0.000 HD
BRANCH  25  27
ATOM     27  C6  T27 A 557      51.125 -21.215  35.635  1.00 22.06     0.103 A
ATOM     28  C5  T27 A 557      51.225 -21.697  34.265  1.00 27.85     1.217 A
ATOM     29  C4  T27 A 557      52.472 -21.683  33.608  1.00 26.06     0.413 A
ATOM     30  C3  T27 A 557      53.632 -21.191  34.222  1.00 25.95    -0.206 A
ATOM     31  H4  T27 A 557      52.538 -22.071  32.577  1.00  0.00     0.000 H
ATOM     32  C2  T27 A 557      53.547 -20.704  35.563  1.00 24.94    -0.166 A
ATOM     33  C1  T27 A 557      52.296 -20.699  36.273  1.00 29.53    -0.337 A
ATOM     34  H2  T27 A 557      54.457 -20.326  36.061  1.00  0.00     0.000 H
BRANCH  28  35
ATOM     35  C8  T27 A 557      50.047 -22.257  33.497  1.00 23.50     5.748 C
ATOM     36  H83 T27 A 557      49.949 -21.720  32.525  1.00  0.00     0.000 H
ATOM     37  H82 T27 A 557      50.290 -23.285  33.142  1.00  0.00     0.000 H
ATOM     38  H81 T27 A 557      49.061 -22.268  34.017  1.00  0.00     0.000 H
ENDBRANCH  28  35
ENDBRANCH  25  27
ENDBRANCH  17  25
ENDBRANCH  13  15
ENDBRANCH   6  13
TORSDOF 5
```

Figure 9.7: PDBQT content of the 2nd conformation.

and uses dynamic programming with branch insertion and deletion and molecular connectivity to construct a mapping between the branches of the two molecules being compared. In another issue, some branches are hydroxyl groups —OH, amine groups —NH_2 or methyl groups —CH_3 where there is only one heavy atom. In this case the calculated moments are meaningless. One possible solution is to incorporate the connecting atoms of child branches into the calculation of the atomic distance distribution of the current branch.

In addition to the USRT development, another future work is to base on USRCAT [20] but expand the atom set to also incorporate protein atoms within the binding site using protein-ligand complex data from PDB [22, 144] or PDBbind [136–138]. Apparently the application is no longer for ligand-based virtual screening, but for characterizing and clustering intermolecular binding patterns in terms of shape. In this way, to discover inhibitors of a target protein, we can borrow knowledge from another well-studied protein-ligand complex that has similar binding site interaction patterns. A related work is FragVLib [288], a free tool for performing similarity search across ligand-receptor complexes using 3D-geometric and chemical similarity of the atoms forming the binding pocket for identifying binding pockets similar to that of a target receptor of interest. Another related work is RAPMAD [289], which permits large-scale mining for similar protein binding pockets on the fly.

Another interesting study [290] presents a rotation-translation invariant molecular descriptor of partial charges and its use in

ligand-based virtual screening. Porting this novel method to USR@istar deserves further investigations.

□ **End of chapter.**

Chapter 10

Case study of CDK2-related cancers

Human colorectal cancer has been reported to express high level of cyclin-dependent kinase 2 (CDK2), a key factor regulating the cell cycle G1 to S transition and a hallmark for cancers. In this study, we used idock prospectively for the first time to identify potential CDK2 inhibitors from 4,311 FDA-approved small molecule drugs with a repurposing strategy. Among the top compounds sorted by idock score, nine were purchased. Among them, adapalene (CD271, 6-[3-(1-adamantyl)-4-methoxyphenyl]-2-naphtoic acid) exhibited the highest anti-proliferative effect in human colon cancer LOVO and DLD1 cells. We demonstrated for the first time that adapalene treatment significantly increased the percentage of cells in G1 phase, and decreased the expressions of CDK2, cyclin E and Rb, as well as the phosphorylations of CDK2 on Thr160 and Rb on Ser795. We then examined the anti-cancer effect of adapalene *in vivo* in BALB/C nude mice subcutaneously xenografted with human colorectal

cancer DLD1 cells. Our results showed that oral adapalene treatment significantly ($p<0.05$) and dose-dependently inhibited tumor growth. Adapalene (20 mg/kg) exhibited strong anti-tumor activity, comparable to that of the leading cancer drug oxaliplatin (40 mg/kg). The combination with adapalene and oxaliplatin exhibited the highest therapeutic effect. These results suggested for the first time that adapalene is a potential CDK2 inhibitor and a candidate anti-cancer drug for the treatment of human colorectal cancer.

This was a collaborative project with Prof. Marie Chia-Mi Lin and Ms. Xi-Nan Shi from Biotechnology Center, Kumming Medical University, China. It was accepted for publication in *Molecular Medicine Reports* on 7 January 2015.

10.1 Background

Cyclin-dependent kinase 2 (CDK2) is one of the serine/threonine protein kinases. It plays a pivotal role in regulating the cell cycle transition from G1 to S phase, and thus in controlling cell proliferation. Abnormally high expression of CDK2 has been reported in many human neoplasias, such as colorectal, ovarian, breast and prostate cancers. Hence, CDK2 inhibitors are potential effective anti-cancer agents.

10.2 Motivation

Although a number of CDK2 inhibitors have been described in the literature and some have entered clinical trial phases [291], e.g. flavopiridol [292], roscovitine [293] and olomoucine [294], none of them is available for clinical use due to various reasons such as toxicity and multi-target specificity.

10.3 Objective

We utilized our free and open-source protein-ligand docking software idock [7, 9] to screen FDA-approved small molecule drugs against CDK2. We adopted the approach of structure-based virtual screening to repurpose approved toxicity-free drugs for the treatment of cancers that involve CDK2 regulation.

10.4 Methods and materials

10.4.1 Ensemble docking and compound selection

44 X-ray crystallographic structures of CDK2 in complex with a bound ligand (Table 10.1) were collected from the PDB (Protein Data Bank) [22, 144]. Figure 10.1, created by iview [11], depicts human CDK2 in complex with ATP. The molecular surface of CDK2 is colored by secondary structure, with an opacity of 0.9 to show the underlying secondary structure in cylinder & plate representation. ATP is rendered in stick representation colored by atom type. Waters are shown as red dots and metal ions are

shown as green dots.

A previously written script [9] was re-used to automatically define the docking search space by finding the smallest cubic box that covers the entire co-crystallized ligand and subsequently extending the box by 10Å in all the three dimensions. The 44 CDK2 structures were manually extracted from their corresponding complexes with the co-crystallized ligands and waters removed, and then converted from PDB format to PDBQT format using the prepare_receptor4.py script of AutoDockTools [32].

The structures of FDA-approved drugs were obtained from the dbap and fda catalogs of the ZINC database [27, 28], where the dbap catalog comprises approved drugs collected from the DrugBank database [296] and the fda catalog comprises approved drugs collected via the DSSTox (Distributed Structure-Searchable Toxicity) project. The dbap catalog of version 2014-03-19 with 1,738 compounds and the fda catalog of version 2012-07-25 with 3,176 compounds were downloaded. Among these 4,914 compounds, 4,311 were unique in terms of ZINC ID. These 4,914 compounds in Mol2 format were then converted to PDBQT format using the prepare_ligand4.py script of AutoDockTools [32].

Our free and open-source docking software idock v2.1.2 [9] was then executed to predict the binding conformations and the binding affinities of the 4,914 compounds when docked against the 44 CDK2 structures using an ensemble docking strategy [115–117]. For each protein structure, free energy grid maps

Table 10.1: The 44 CDK2 holo structures used for ensemble docking.

PDB ID	Resolution (Å)	UniProt ID
1AQ1	2.00	P24941
1CKP	2.05	P24941
1DI8	2.20	P24941
1DM2	2.10	P24941
1E1V	1.95	P24941
1E1X	1.85	P24941
1FVT	2.20	P24941
1G5S	2.61	P24941
1GIH	2.80	P24941
1GII	2.00	P24941
1GIJ	2.20	P24941
1GZ8	1.30	P24941
1H00	1.60	P24941
1H01	1.79	P24941
1H07	1.85	P24941
1H08	1.80	P24941
1H0V	1.90	P24941
1H0W	2.10	P24941
1JSV	1.96	P24941
1JVP	1.53	P24941
1KE5	2.20	P24941
1KE6	2.00	P24941
1KE7	2.00	P24941
1KE8	2.00	P24941
1KE9	2.00	P24941
1OIQ	2.31	P24941
1OIR	1.91	P24941
1OIT	1.60	P24941
1P2A	2.50	P24941
1PF8	2.51	P24941
1PXI	1.95	P24941
1PXJ	2.30	P24941
1PXK	2.80	P24941
1PXL	2.50	P24941
1PXM	2.53	P24941
1PXN	2.50	P24941
1PXO	1.96	P24941
1PXP	2.30	P24941
1PYE	2.00	P24941
1R78	2.00	P24941
1URW	1.60	P24941
1V1K	2.31	P24941
1VYZ	2.21	P24941
1W0X	2.20	P24941

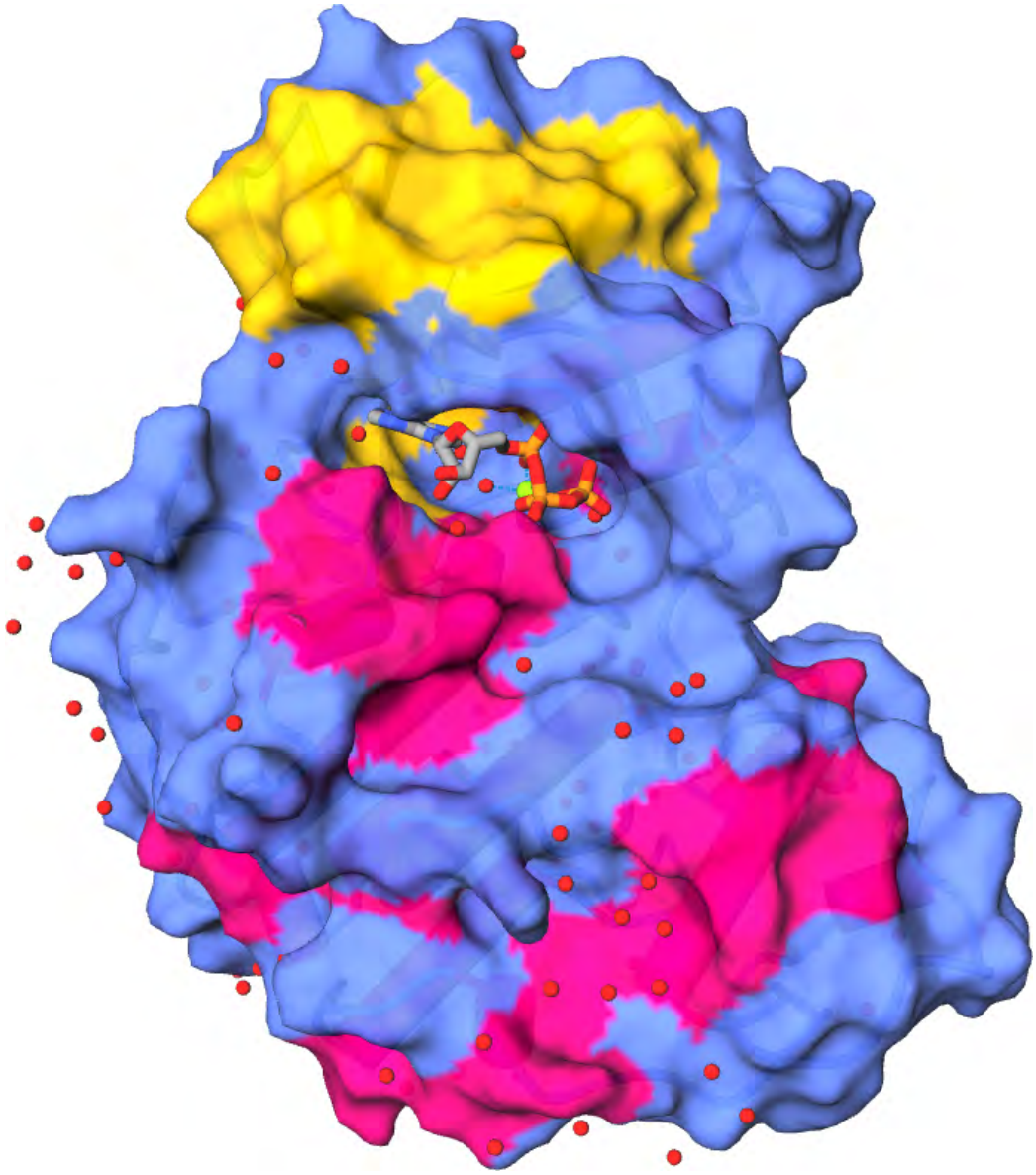


Figure 10.1: Crystal structure of human CDK2 with ATP (PDB ID: 1HCK) [295].

with a fine granularity of 0.08 Å were constructed in parallel, and for each compound, 256 Monte Carlo conformational optimization tasks were run in parallel across multiple CPU cores.

After docking, idock outputted a maximum number of nine predicted conformations for each input compound. The docked conformation with the best idock score was selected because it was previously shown to be the most likely one closest to the crystal conformation with a redocking success rate of more than 50% on multiple benchmarks [9]. The 4,914 compounds were sorted in the ascending order of their predicted binding free energy averaged across the 44 CDK2 structures, and the top-scoring ones were visually examined using iview [11] and PoseView [34] in the context of CDK2 using the X-ray structure of the highest resolution, i.e. PDB ID 1GZ8 in this case (Table 10.1). Finally, commercially available compounds were purchased via the Chemical Book website http://www.chemicalbook.com/ and subsequently validated *in vitro* and *in vivo*.

10.4.2 Chemicals, antibodies, cell lines and cell culture

The selected chemicals and the leading cancer drug oxaliplatin were purchased from Sigma-Aldrich, USA. Anti-cyclin D, B1, E, CDK2, Rb, Pho-CDK2 (Thr160), Pho-Rb (Ser795) and GAPDH were obtained from Cell Signaling Technology, Inc., Danvers, Massachusetts, USA.

Colorectal cancer cell lines LOVO and DLD1 were obtained from the American Type Culture Collection, Manassas, Vir-

ginia, USA. These cell lines were cultured in RPMI 1640 medium containing 10% fetal bovine serum (FBS) (Invitrogen, Rockville, Maryland, USA) at 37°C in 5% CO_2 and 95% humidified air.

Cells were plated in 96-, 24-, or 6-well plates with 0.125% FBS medium for 24 hours and then treated with 10% FBS medium containing the testing compounds at various concentrations of 1, 3, 10, 30μM, and incubated for 24, 48, or 72 hours.

10.4.3 MTT assay

Cells were plated at an initial density of 9x10^3 cells/well in 96 well plates and incubated with 0.5mg/ml 3-(4,5-methylthiazol-2-yl)-2,5-diphenyl-tetrazolium bromide for 4 hours. The medium was then discarded and 200μl of formazan in dimethylsulphoxide (DMSO) was added. The absorbance was measured at 570 nm according the standard protocol. The IC_{50} values were calculated by Graphpad prime5.

10.4.4 Cell cycle analysis

LOVO or DLD1 cells (4x10^4) were seeded in 24-well plates in RPMI 1640 medium containing 0.125% FBS, and cultured for 24 hours. The cells were incubated in medium containing 10% FBS and various doses of adapalene (1, 3, 10, 30 μM) for 12, 24, 36 hours at 37°C, then fixed in ice-cold 70% ethanol and stained with a Coulter DNA-Prep Reagents kit (Beckman Coulter, Fullerton, California, USA). Cellular DNA content of 1x10^4 cells from each sample was determined with the use of an EPICS

ALTRA flow cytometer (Beckman Coulter). Cell cycle phase distribution was analyzed with the ModFit LT 2.0 software (Verity Software House, Topsham, Maine, USA). All results were obtained from two separate experiments, each of which was done in triplicate.

10.4.5 Western blotting

LOVO and DLD1 cells were plated at 6-well plates with 0.125% FBS medium for 24 hours and then with 10% FBS medium containing adapalene at concentration 3, 10, 30μM. Cells were harvested after 6 hours of incubation. Cells were lysed with RIPA buffer containing 1 mM PMSF and protease inhibitor cocktail at 4°C for 30 minutes. After centrifugation at 13,000 rpm for 15 minutes, the supernatants were recovered and the protein concentration was measured by BCA Protein Assay Kit (Thermo). Equal amounts of cell lysates were resolved in 10% SDS-PAGE and transferred onto nitrocellulose membranes (Sigma). After blocking, the membranes were incubated sequentially with the appropriate diluted primary and secondary antibodies. Proteins were detected by the enhanced chemiluminescence detection system (Amersham, Piscataway, New Jersey, USA). To ensure equal loading of the samples, the membranes were re-probed with an anti-GAPDH antibody (Cell Signalling Technologies).

10.4.6 Adapalene treatment *in vivo* in nude mice xenografted with colorectal cancer DLD1 cells

Female BALB/C nude mice, 4 to 5 weeks old from Vital River Laboratory Technology Co. Ltd, Peking, China, were kept under specific pathogen-free conditions and cared for in accordance with the guidelines of the laboratory animal ethics committee of Kunming Medical University. For the xenografted tumor growth assay, $1x10^6$/0.2ml PBS DLD1 cells were injected subcutaneously into the right flank of the mice. Tumor size was measured every day. One week after inoculation when the tumors grew to a volume of 80 to 100 m^3, the mice were randomly divided into groups of 5 mice per group, and fed by gavage daily for 21 days with 0.5% CMC-NaCl containing various doses (15, 20, 65 and 100mg/kg) of adapalene and oxaliplatin (40mg/kg). The mice were then sacrificed by cervical dislocation. Tumor volume was calculated by the formula $V = ab^2/2$, where a is the longest axis and b is shortest axis.

10.4.7 Statistical analysis

The results were obtained from at least three different experiments and expressed as mean ± SD. Statistical analysis was performed by Student's t test and differences were considered to be statistically significant if $p < 0.05$. Statistically significant results are marked with the ※ symbol.

Table 10.2: The nine top-scoring compounds purchased and validated.

ZINC ID	idock score (kcal/mol)	name
06716957	-10.46	nilotinib
03830332	-10.43	LS-194959
03784182	-10.38	adapalene
03830768	-10.23	estradiol benzoate
03881613	-10.08	nandrolone phenylpropionate
01542113	-10.06	vilazodone
00897240	-10.01	azelastine hydrochloride
33974796	-9.98	latuda
01481956	-9.95	paliperidone

10.5 Results

10.5.1 Ensemble docking results and selection of candidate inhibitors

Totally 4,914 FDA-approved drugs were docked and ranked according to their average predicted binding affinity across 44 X-ray crystal structures of CDK2 (Table 10.1). The docking prediction results with iview visualization [11] are freely available at http://istar.cse.cuhk.edu.hk/idock/iview/?1GZ8-dbap. Based on commercial availability, nine top-scoring compounds (Table 10.2) were selected and purchased for further investigations.

10.5.2 Adapalene decreased cell viability of colorectal cancer

We first evaluated the anti-cancer effect of the nine compounds by MTT assay (Figure 10.2). All the nine compounds decreased cell viability in LOVO and DLD1 cells, but had discrepant cy-

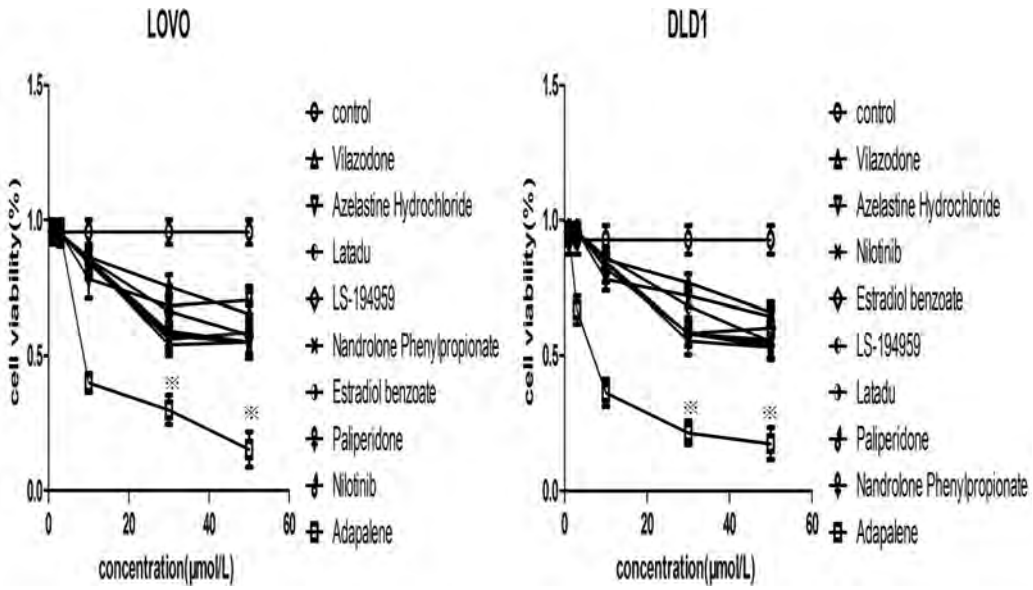


Figure 10.2: Comparison of the effect of the nine compounds on the viability of LOVO and DLD1 colorectal cancer cells.

totoxicity at different concentrations. Among them, adapalene had the lowest IC_{50}, i.e. 7.135μM for LOVO and 4.43μM for DLD1. In other words, adapalene exhibited the highest cytotoxicity compared to the control with statistical significance ($\divideontimes p<0.05$).

Adapalene exhibited dose- and time-dependent inhibition effect on cell viability in LOVO and DLD1 cell lines compared to the control ($\divideontimes p<0.05$) (Figure 10.3). Marked inhibition was observed at 10μM and 30μM, but no significant effect was observed at concentrations below 3μM.

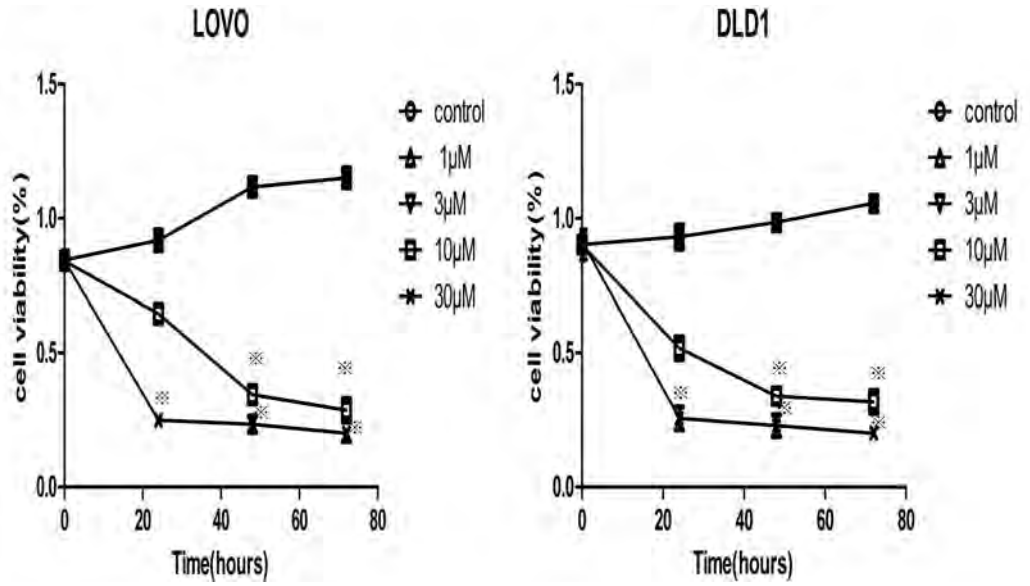


Figure 10.3: The growth inhibition effect of adapalene on LOVO and DLD1 colorectal cancer cells.

10.5.3 Adapalene treatment arrested cell cycle in G1 phase

We analyzed the effect of adapalene treatment with concentrations of 3, 10, 30 μM for 6, 12, 24 hours on cell cycle profile in LOVO and DLD1 cells by flow cytometry (Figure 10.4) in order to understand if adapalene inhibited CDK2 activities in colorectal cancer cells. Adapalene treatment significantly increased the percentage of cells in G1 phase compared to the control ($\divideontimes p<0.05$) in a dose- and time-dependent manner. At 30μM or 10μM concentrations, adapalene treatment continuously increased the percentage of G1 phase for 24 hours.

Figure 10.5 shows the changes of cell cycle profile of G0-G1, S, and G2-M phases after 24 hours of adapalene treatment. The increase of the G1 phase was accompanied by the simultaneous

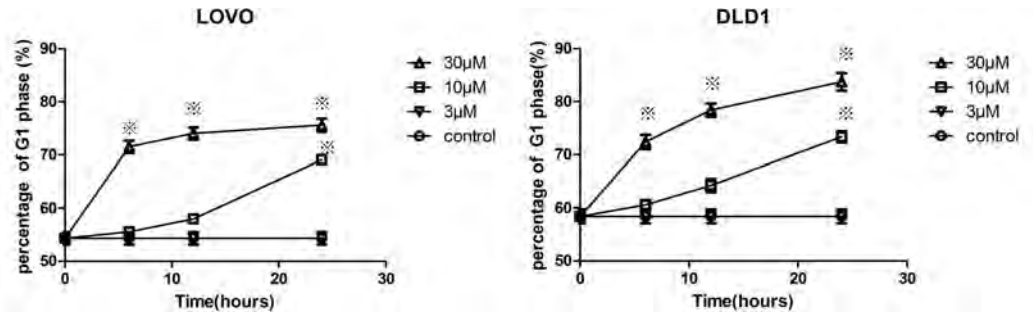


Figure 10.4: Dose- and time-dependent effect of adapalene treatment on the percentage of cells in G1 phase.

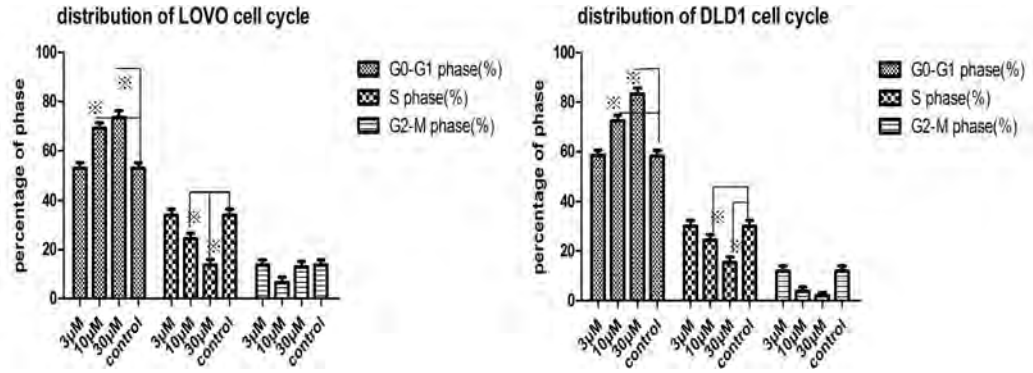


Figure 10.5: Cell cycle distributions at 24 hours after adapalene treatment.

decrease of S and G2-M phases.

10.5.4 Adapalene treatment decreased the expressions of CDK2, Rb, cyclin E, pho-CDK2 and pho-Rb, but not cyclin D and cyclin B1

We investigated the effect of adapalene on the expressions of critical proteins involved in G1-to-S transition by western blotting in LOVO and DLD1 cells (Figure 10.6). Adapalene treatment reduced the expressions of CDK2, Rb, pho-CDK2, pho-Rb and cyclin E. In contrast, the expression levels of cyclin D1 and cyclin B1 remained unchanged. These results are consistent with

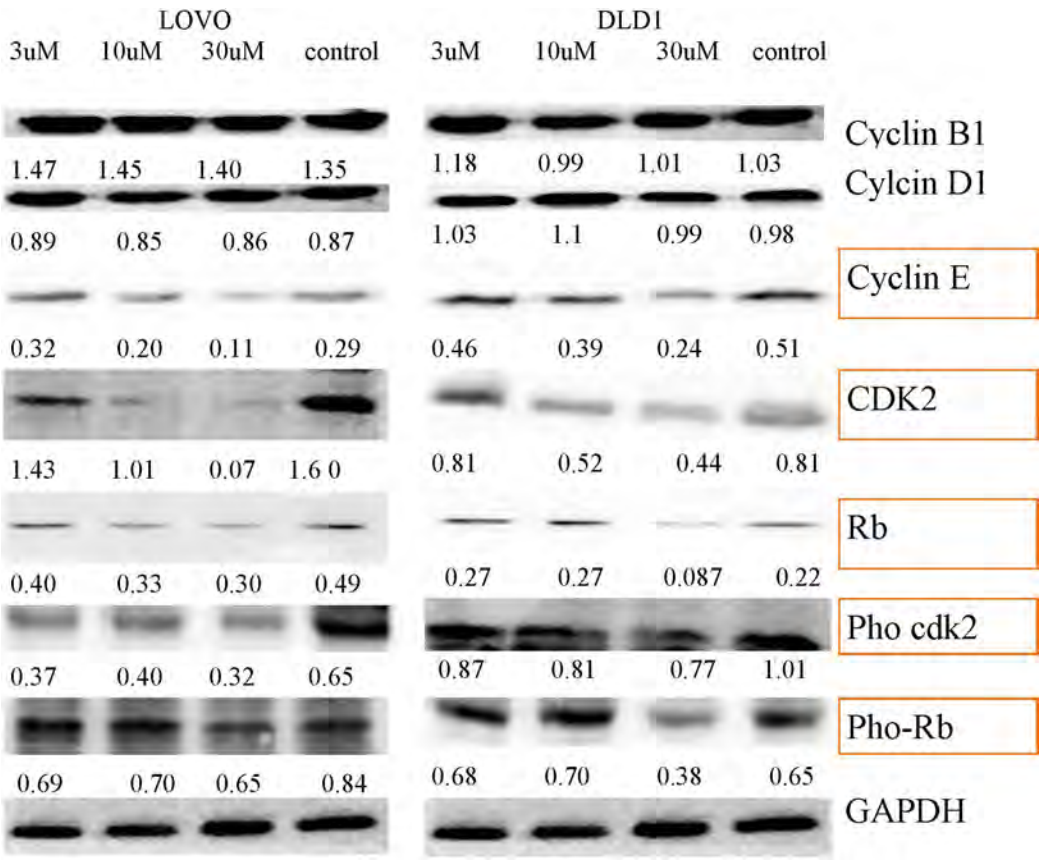


Figure 10.6: Effect of adapalene treatment on the expressions of cyclins, CDK2 and Rb.

what are expected from a CDK2 inhibitor.

10.5.5 Daily oral adapalene treatment reduced tumor growth *in vivo*

To evaluate the effect of adapalene on the growth of colorectal carcinoma *in vivo*, BALB/C nude mice were subcutaneously injected with DLD1 cells. Carcinoma volumes were measured every 3 to 4 days after tumor appearance. At day 7 after tumor inoculation, the tumor volume reached 80 to 100 mm^3, then var-

ious doses (15, 65, 100mg/kg in 0.5% CMC-NaCl) of adapalene were administered daily for 21 days by oral gavage. Figure 10.7 shows that oral adapalene treatment significantly (※p<0.05) inhibited tumor growth. At day 21 after treatment, 15 mg/kg adapalene resulted in significant reduction of tumor weight and volume compared to the control (※p<0.05). Nevertheless, there is no significant difference between 15 and 65 mg/kg adapalene treatment.

In a separate experiment, we compared the efficacy of adapalene (20mg/kg), oxaliplatin (40mg/kg) and the combination of adapalene (20mg/kg) plus oxaliplatin (40mg/kg) (Figure 10.8). The anti-tumor activity of oral adapalene (20 mg/kg) was comparable to that of oxaliplatin (40 mg/kg). Importantly, the combinatorial therapy exhibited the highest therapeutic effect. These results suggested for the first time that adapalene is a potential CDK2 inhibitor and a candidate anti-cancer drug for the treatment of human colorectal cancer.

10.5.6 Structural analysis of the predicted conformation of adapalene docked against CDK2

Figure 10.9 plots the predicted conformation of adapalene in complex with CDK2 (PDB ID: 1GZ8) using iview [11]. Figure 10.10 plots the intermolecular interaction diagram using PoseView [34]. Adapalene was predicted to reside in the ATP-binding site of CDK2 and interact with CDK2 mainly through hydrophobic contacts with Phe82, Ile10, Leu134, Lys33 and

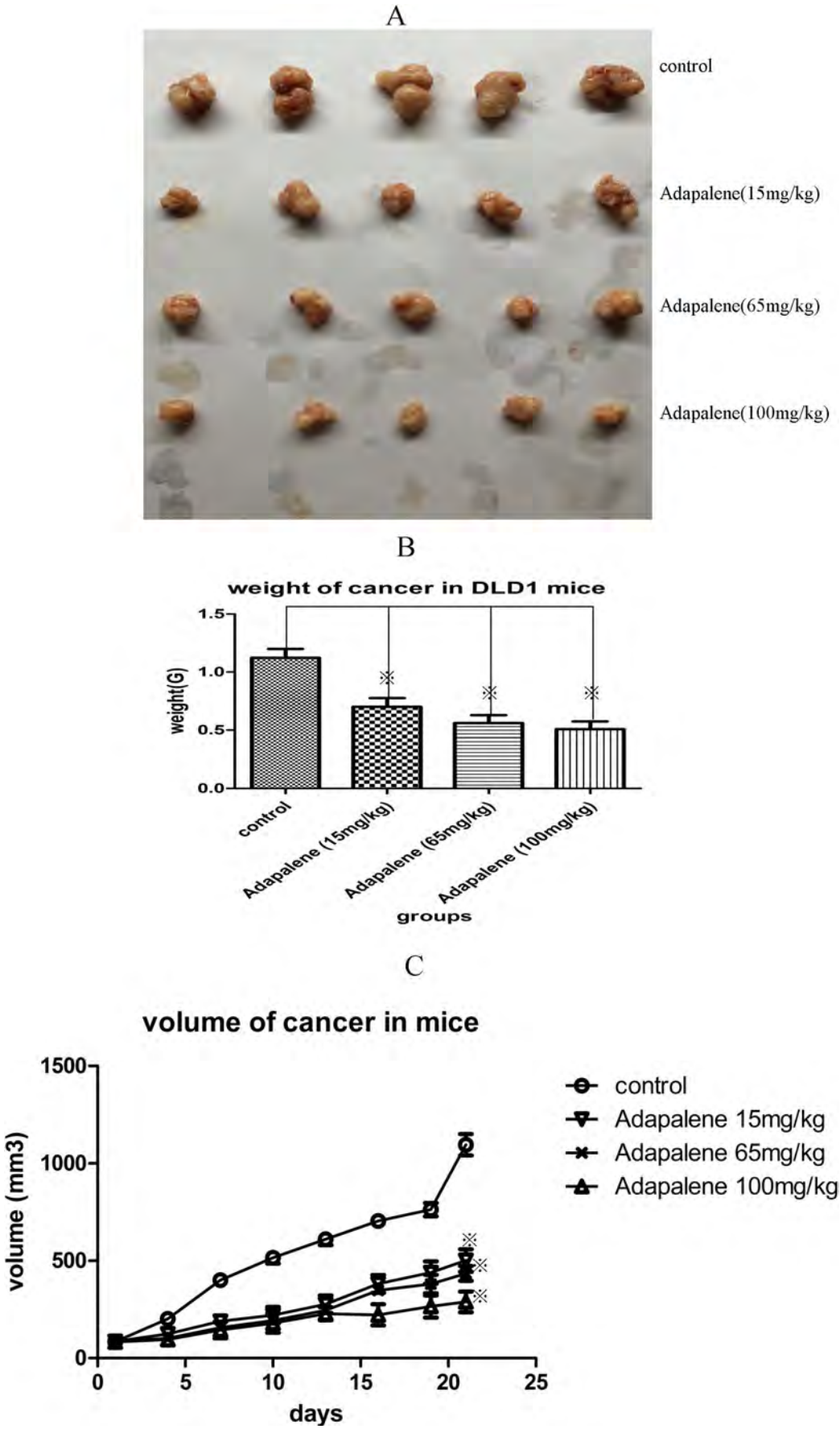


Figure 10.7: Effect of oral treatment of adapalene on tumor growth *in vivo* in nude mice xenografted with DLD1 cells.

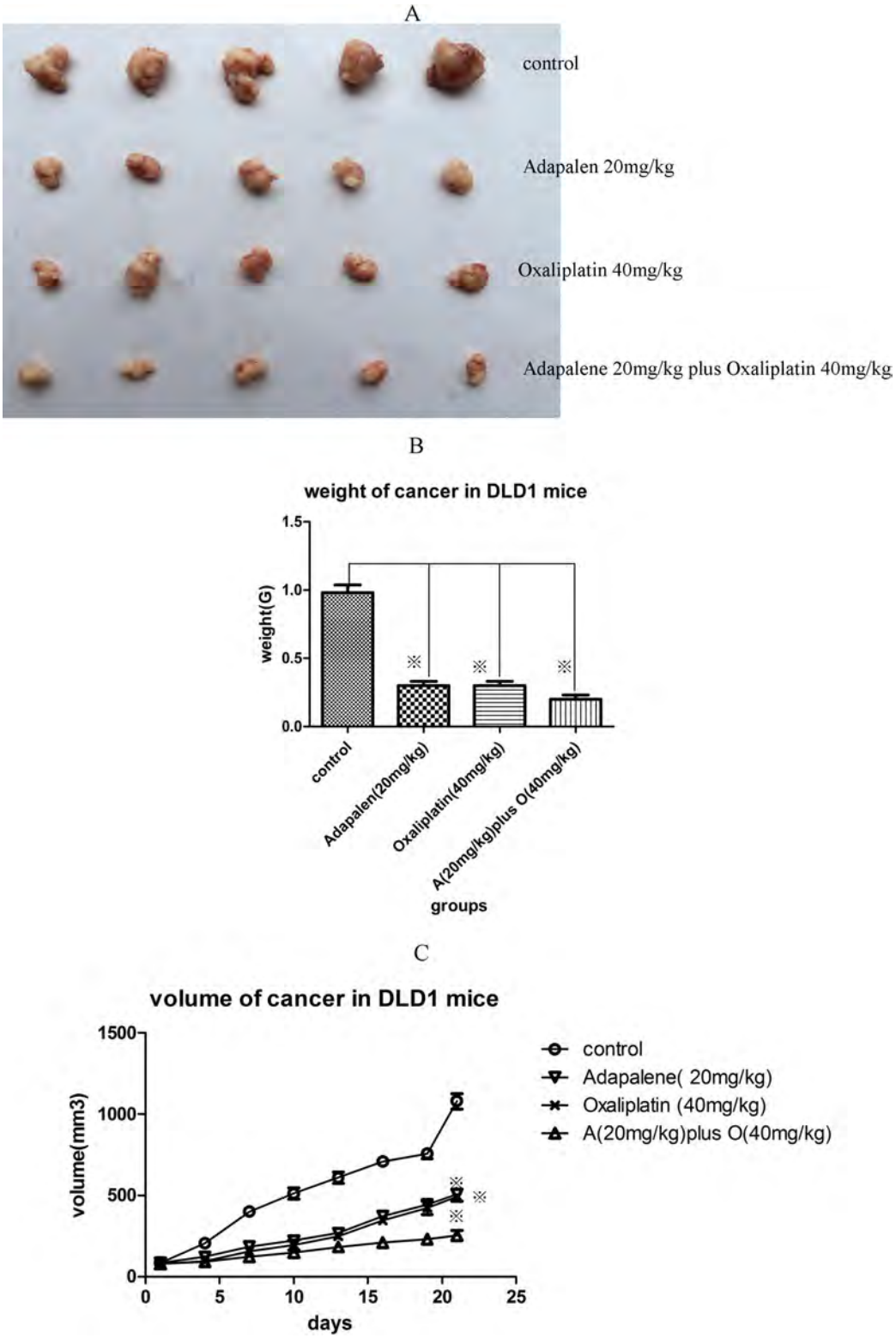


Figure 10.8: Effect of oral treatment of adapalene combined with oxaliplatin on tumor growth *in vivo* in nude mice xenografted with DLD1 cells.

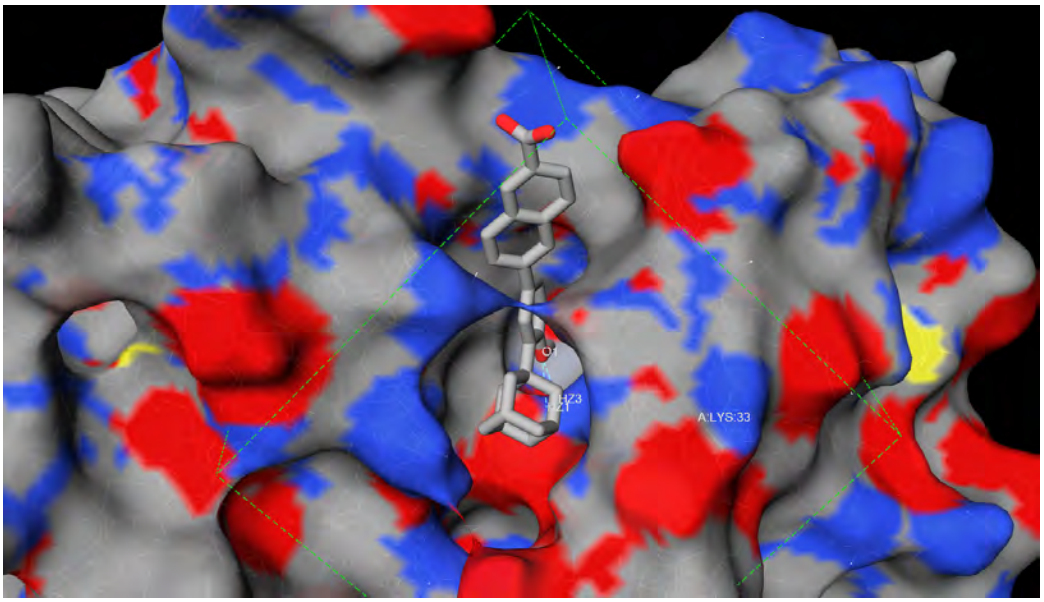


Figure 10.9: The predicted conformation of adapalene in complex with CDK2.

His84, and a hydrogen bond with Lys33.

10.6 Discussion

Cell cycle progress is sequentially and strictly processed through the interactions of CDKs and cyclins. Different cyclin-CDK complexes are activated in different phases of the cell cycle. When the cell cycle goes through G1 to S phase, the cyclin D1-CDK4/6 and cyclin E-CDK2 complexes are ordinally activated and the retinoblastoma protein (pRB) is hyper-phosphorylated on serine and threonine residues. The hyper-phosphorylated pRB promotes the release of E2F transcription factors, which in turn facilitate the transcription of numerous genes required for G1 to S transition and S phase progression. From the medici-

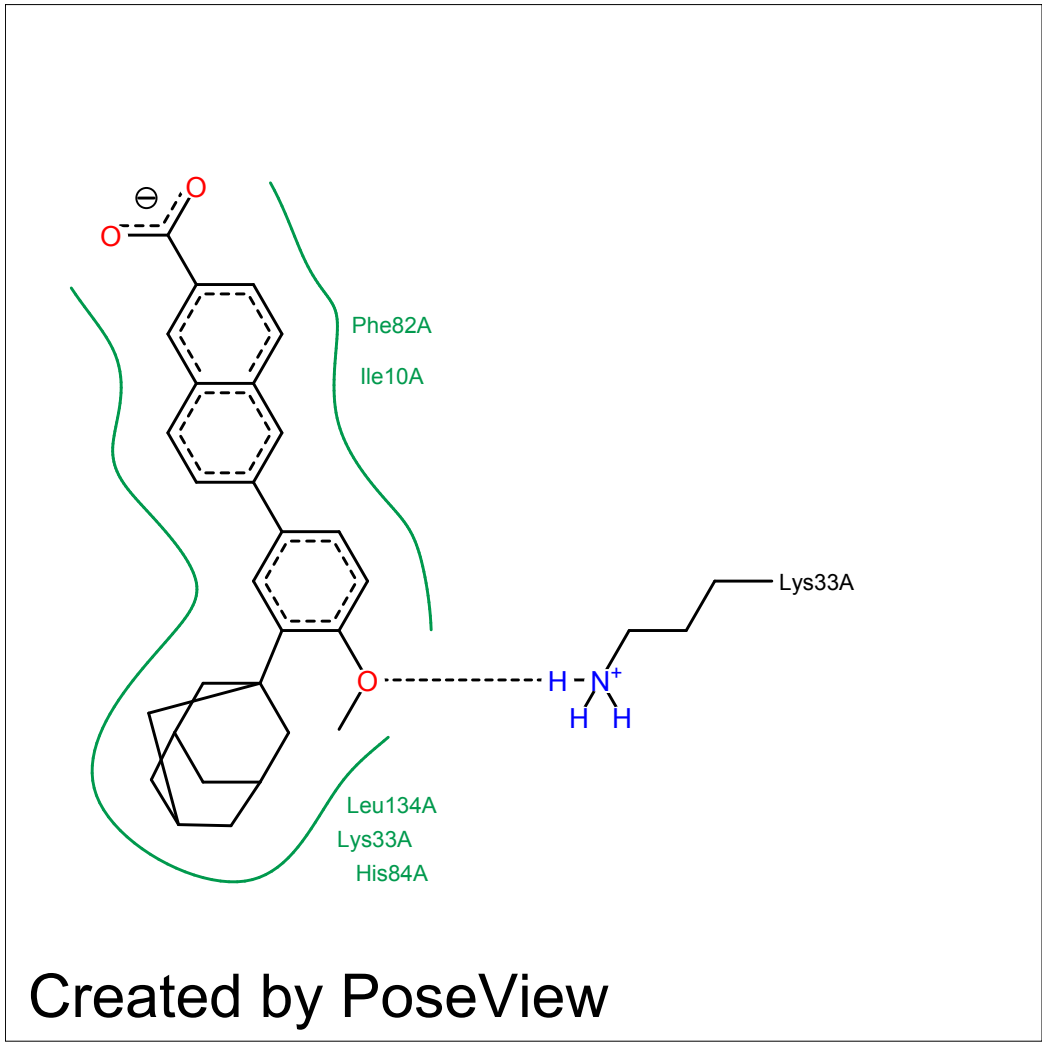


Figure 10.10: The putative interactions of adapalene with CDK2.

nal perspective, CDK2 has long been a classical and important target for cancer therapy.

Though a number of CDK2 inhibitors have entered clinical trial phases, none has been officially approved for clinical use, probably because of their toxicity and multi-target specificity. Given the obstacle that developing a new drug *de novo* is a laborious and costly endeavor, repurposing toxicity-free old drugs for new uses is a favorable strategy.

The powerful synergy of *in silico* methods in drug repurposing by structure-based virtual screening (SBVS) was highlighted in several recent reports [190]. To name a few successful repurposing cases by SBVS, [297] rediscovered 2,4-Dichlorophenoxy acetic acid, a well-known plant auxin, as a new anti-inflammatory agent through *in silico* molecular modeling and docking studies along with drug formulation and *in vivo* anti-inflammatory inspection; [298] attempted to repurpose FDA-approved drugs by an integrated SBVS approach and reported the discovery of piperacillin **1** as an inhibitor of NEDD8-activating enzyme (NAE) in cell-free and cell-based systems with high selectivity.

In addition to SBVS, ligand-based virtual screening (LBVS) also finds its successful applications in repurposing. [256] used Ultrafast Shape Recognition (USR) [19] to search for compounds with similar shape to a previously reported inhibitor of protein arginine deiminase type 4 (PAD4), a new therapeutic target for the treatment of rheumatoid arthritis, and identified a novel compound that has a strikingly different structure from the template inhibitor yet showed significant inhibition on the

enzymatic activity of PAD4.

Encouraged by these successful stories, in this study we adopted the repurposing strategy, and utilized the computational methodology of SBVS by protein-ligand docking to shortlist candidates from FDA-approved small molecule drugs. Specifically, we used our fast docking program idock [7, 9] in combination with our convenient visualizer iview [11] for the task of rediscovering existing drugs as CDK2 inhibitors. idock is an exciting development not only because it has been vigorously shown [9] to outperform the state-of-the-art docking software AutoDock Vina [8] in terms of docking speed by at least 8.69 times and at most 37.51 times while maintaining comparable redocking success rates, but also because it is free and open source under a permissive license. The latter guarantees that users from both industry and academia can freely utilize idock in their own projects that require protein-ligand docking.

To facilitate the use of idock, its input arguments and output results were purposely designed to be similar to those of AutoDock Vina, therefore existing users can easily transit to idock and benefit from considerable speedup in SBVS performance. Moreover, to promote prospective SBVS by idock, a web server called istar [9] was developed and made available at http://istar.cse.cuhk.edu.hk/idock, where there are as many as over 23 million purchasable small molecule compounds ready for docking against any protein supplied by the user. Both idock [7] and istar [9] would hopefully supplement the efforts of medicinal chemists in drug discovery research.

Regarding the structural data in use, there are so far as many as 346 solved X-ray crystal structures of CDK2 with a UniProt ID of P24941. To account for their structural variability and to mine knowledge from multiple structures of CDK2, we selected 44 holo structures of CDK2 in a bound state with a ligand in complex to carry out ensemble docking. The final score used to prioritize compounds was purposely designed to be the average score of that compound when docked to the 44 selected structures of CDK2 with their native ligand removed manually before docking. In this way the top-scoring compounds would guarantee a consistent binding strength on average. In the aspect of data source of approved drugs, although we chose the dbap and fda catalogs of the ZINC database [27, 28], it is also possible to use some other freely accessible drug databases such as NCGC [299], DrugBank [296], KEGG DRUG [300] and e-Drug 3D [181].

After ensemble docking experiments with idock [7, 9] followed by careful visual inspections with iview [11], we purchased nine top-ranking compounds for subsequent wet experiments. Among them, adapalene was selected for further investigations because its IC_{50} was less than 10 umol/L as determined by MTT assay. Adapalene is the third generation of synthetic retinoids, currently used for the topical therapy of acne vulgaris [301]. Its anti-proliferative and proapoptotic effects *in vitro* were first reported in colon carcinoma cell lines (CC-531, HT-29 and LOVO) [302] and hepatoma cells (HepG2 and Hep1B) [303] by increasing the activity of caspase 3 via up-regulating bax and down-regulating bcl-2.

In this study, we reported for the first time that adapalene is a potential CDK2 inhibitor, and demonstrated for the first time that oral administration of adapalene (20 mg/kg) exhibited significant and strong anti-cancer efficacy as compared to the leading cancer drug oxaliplatin (40 mg/kg) *in vivo* in nude mice xenografted with colorectal DLD1 cells. Most importantly, the combination of effective dose of adapalene and oxaliplatin produced even higher therapeutic effect, indicating that adapalene may work through a different mechanism than oxaliplatin, which further indicates that adapalene could be combined with other chemotherapy drugs to achieve synergistic therapeutic effect.

No obvious toxicity was previously reported by either intraperitoneal injection of 100 mg/kg adapalene in carrageenan induced paw oedema rat, or topic use of 10% (10 mg/ml) in UV induced erythema guinea pig [304]. In this study, we did not observe significant change in body weight by oral administration of adapalene (15 to 100 mg/kg) for 21 days, suggesting that oral administration or intraperitoneal injection of adapalene is relatively safe.

Intraperitoneal injection of 5 and 10 mg/kg oxaliplatin in B6D2F mice subcutaneously xenografted with colon 38 was previously reported to significantly reduced tumor weight to 38% and 16% of the control level, respectively, at 21 day post-treatment [305]. In this study, we tested oral oxaliplatin treatment by gavage at doses of 10, 20, and 40mg/kg, and found that 40mg/kg effectively reduced the tumor weight to 28% of the control on day

21 post-treatment without showing any body weight change, indicating that oral administration of oxaliplatin is relatively safe and effective.

10.7 Conclusions

This study presents the first successful prospective application of idock [7, 9] in identifying CDK2 inhibitors from FDA-approved small molecule drugs using a repurposing strategy. We have showed that adapalene (CD271, 6-[3-(1-adamantyl)-4-methoxyphenyl]-2-naphtoic acid), currently used for the topical therapy of acne vulgaris, exhibited anti-cancer effect in human colorectal LOVO and DLD1 cells. We have demonstrated for the first time that oral adapalene treatment significantly and dose-dependently inhibited tumor growth. Most importantly, the combinatorial therapy of adapalene and the leading cancer drug oxaliplatin exhibited higher therapeutic effect. These results have suggested for the first time that adapalene is a potential CDK2 inhibitor and a candidate anti-cancer drug for the treatment of human colorectal cancer. The potential application of adapalene combined with other chemotherapy drugs for the treatment of colorectal neoplasms and other cancers warrants further investigations.

□ **End of chapter.**

Chapter 11

Case study of influenza A

Influenza is a serious respiratory disease that causes severe illness and death in high risk populations. The rapid emergence of drug-resistant viral mutations void existing drugs. It is thus in urgent need of new anti-influenza drugs that inhibit novel viral proteins such as nucleoprotein (NP) and the RNA-dependent RNA polymerase (RdRP) subunits PA, PB1 and PB2.

In this study, we targeted at three novel targets: the tail-loop binding domain of NP, the PB1-binding domain of PA, and the cap-binding domain of PB2. We utilized idock to perform structure-based virtual screening of 273,880 cheaply available compounds, and identified several hits that were predicted to establish strong interactions with their respective viral protein target and believed to exhibit strong inhibitory effects. These identified compounds may serve as promising candidates for subsequent investigations *in vitro* and *in vivo*.

This is an ongoing collaborative project with Prof. Pang-Chui Shaw and Mr. Edwin Lo from School of Biomedical Sciences,

Chinese University of Hong Kong, Hong Kong.

11.1 Background

According to the fact sheets of World Health Organization, available at http://www.who.int/mediacentre/factsheets/fs211/en/, influenza viruses have been causing sporadic pandemics and annual epidemics worldwide, every year claiming 250,000 to 500,000 lives and resulting in about 3 to 5 million cases of severe illness. The H5N1 bird flu outbreak in Hong Kong in 1997, the H1N1 swine flu outbreak in Mexico in 2009, and the ever-present threat of H5N1 acquiring human-to-human transmission capability remind us of the imminent danger posed by the influenza viruses.

Influenza viruses are negative-sense single-stranded RNA viruses. They are classified into types A, B and C based on the antigenic difference in their nucleoproteins and matrix proteins. Influenza A is the major pathogen for most cases of epidemic influenza. The influenza A genome comprises 8 segments of RNA coding for 11 proteins, which are hemagglutinin (HA), neuraminidase (NA), matrix protein 1 (M1), M2 proton channel, nucleoprotein (NP), non-structural protein 1 (NS1), nuclear export protein (NEP), polymerase acid protein (PA), polymerase basic proteins (PB1 and PB2) and PB1-F2. Influenza A viruses are further organized into 16 hemagglutinin subtypes (H1-H16) and 9 neuraminidase subtypes (N1-N9) according to their distinct antigenic properties.

The life cycle of influenza viruses has been well studied [306–308] and nearly all the viral proteins have become potential therapeutic targets [306, 308–311]. To date, four anti-influenza drugs have been approved by the US FDA (Food and Drug Administration). In order of their release, they are two M2 channel blockers, amantadine (Symmetrel®) and rimantadine (Flumadine®), and two NA inhibitors, zanamivir (Relenza®) and oseltamivir (Tamiflu®). Unfortunately, oseltamivir, amantadine and rimantadine are now found to be ineffective against circulating strains due to the rapid emergence of drug-resistant viral mutations in pandemic and seasonal influenza viruses. Even worse, amantadine and rimantadine exhibit side effects on the central nervous system. No drugs have been firmly established for the other viral proteins, although leads have been proposed in some cases [307, 308, 310, 311]. These alarming facts highlight the urgent need for designing new anti-influenza drugs.

In this study we concentrate on discovering inhibitors of three influenza A proteins: NP, PA and PB2. These viral proteins are structurally related in that NP forms homo-oligomers and multiple copies of NP wrap around genomic RNA, along with a trimeric RNA-dependent RNA polymerase (RdRP) of subunits PA, PB1 and PB2 making up a ribonucleoprotein (RNP) complex.

11.1.1 Nucleoprotein (NP)

NP is the most abundantly expressed viral protein during the course of infection. NP is a polypeptide of 498 amino acids, which fold into a crescent shape with a head and a body domain. Functionally speaking, NP not only encapsidates the viral RNA, but also forms homo-oligomers. NP homo-oligomerizes by inserting the flexible tail loop (amino acids 402 to 428) into the groove of the body domain of its neighboring NP molecule.

Several crystal structures of NP have been solved. The first is a 3.2Å resolution structure from human origin H1N1 (PDB ID: 2IQH) [312]. The second is a 3.3Å resolution structure from avian origin H5N1 (PDB ID: 2Q06) [313]. Figure 11.1 shows the H1N1 NP crystal structure [312], rendered by iview [11]. H5N1 NP and H1N1 NP share 94% sequence identity [313]. Their root mean square deviation (RMSD) is 1.0Å after aligning 398 residues [313]. The interaction of the tail loop of one NP molecule with the neighboring protomer in H5N1 and H1N1 is virtually identical [313]. Other more recently solved structures of H1N1 NP include 3RO5 [314], 3TG6, 4IRY [315], 3ZDP [316], 4DYA, 4DYB, 4DYN, 4DYP, 4DYT and 4DYS.

The NP tail loop makes extensive interactions with the binding groove through intermolecular β-sheets, hydrophobic interactions and salt bridges [312]. Specifically at the residual level, the salt bridge between E339 lining the binding pocket and R416 on the tail loop is essential. The E339A mutant totally abolishes the RNP activities [317]. E339A and R416A are unable to

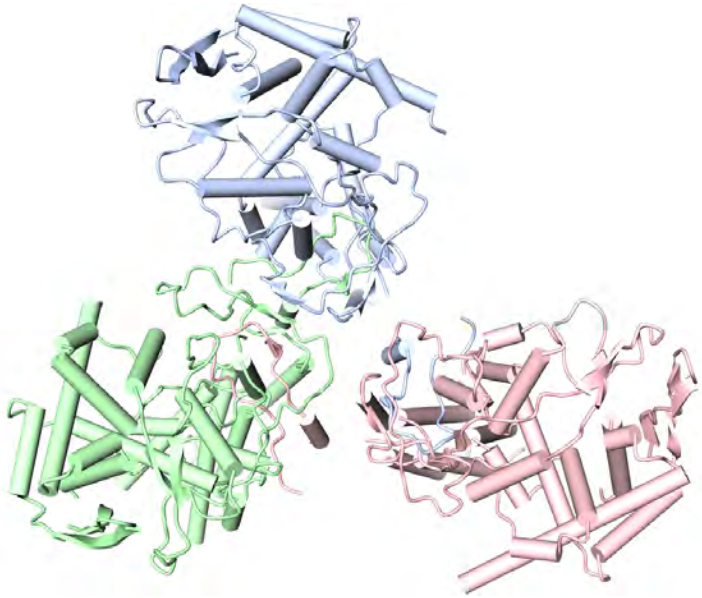


Figure 11.1: Crystal structure of H1N1 nucleoprotein trimer with three subunits shown in different colors.

support viral replication in the absence of wild type NP [318]. The R267A and E449A mutants decrease the RNP activities by more than 50% [317]. These results indicate that within the tail-loop binding groove E339 is critical while R267 and E449 are important for NP homo-oligomerization.

The displacement of the tail loop from its binding pocket causes significant structural rearrangements in NP and completely abolishes the replication and transcription functions [313]. Tail-loop peptides are shown to disrupt NP-NP interaction and inhibit viral replication [318]. The amino acids in the tail-loop binding groove for NP oligomerization are highly conserved across 4430 sequences of NP among all influenza A virus subtypes from all hosts [319]. Therefore the tail-loop binding groove is an attractive target for inhibitor design. Chemical compounds

which competitively displace the tail loop from its binding pocket would interfere with viral genome replication, and thus serve as promising candidates for anti-influenza drug development [312, 313, 317, 318]. Targeting at the tail-loop binding site, a few novel inhibitors [318] have been identified to be effective against both wild-type and mutant strains using structure-based virtual screening, but none have been approved as drugs.

11.1.2 Polymerase acidic protein (PA)

The RNA-dependent RNA polymerase is a heterotrimer composed of three subunits, namely PA, PB1 and PB2. PA contains the endonuclease domain, PB1 carries the polymerase active site, and PB2 includes the capped-RNA recognition domain. All the three subunits are required for both viral transcription and replication.

The amino-terminal residues of PB1 interact with the carboxy-terminal domain of PA. Two crystal structures of PA_C-$PB1_N$ complex have been solved. The first is a 2.9Å resolution structure of avian H5N1 influenza A virus PA (PA_C, residues 257-716) in complex with the PA-binding region of PB1 ($PB1_N$, residues 1-25) (PDB ID: 3CM8) [320]. The second is a 2.3Å resolution structure of influenza A H1N1 (PDB ID: 2ZNL) [321]. Figure 11.2 shows the H1N1 PA_C-$PB1_N$ crystal structure, rendered by iview [11]. In addition to PA_C-$PB1_N$ structures, two apo crystal structures of PA_C in the absence of PB1 have been reported recently [322]. The first is a 1.9Å resolution structure of H1N1

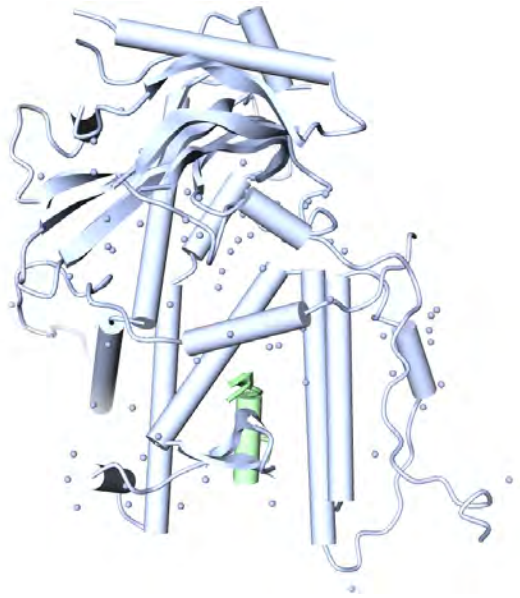


Figure 11.2: Crystal structure of the C-terminal domain of H1N1 PA bound to the N-terminal peptide of PB1.

PA_C (PDB ID: 4IUJ). The second is a 2.2Å resolution structure of H7N9 PA_C (PDB ID: 4P9A).

The C-terminal domain of PA forms a deep and highly hydrophobic groove into which the N-terminal residues of PB1 can fit by forming a 3_{10} helix $P_5TLLFLK_{11}$ and interacting through an array of hydrogen bonds and hydrophobic contacts [321]. Four double mutations W706A/Q670A, L666G/F710E, L666G/F710G and W706A/F710Q disrupt the binding of $PB1_N$ to PA_C [320]. Four point mutations V636S, L640D, L666D and W706A greatly weaken or abolish PB1 binding, and similarly reduce viral RNA synthesis in human cells [321].

The loss of PA abolishes RNA polymerase activity and viral replication. Peptides corresponding to the PA-binding domain of PB1 block the polymerase activity and inhibits viral spread

[323–326]. The residues from PA_C and $PB1_N$ at the interface are highly conserved in H1N1, H5N1 and other influenza A viruses [320, 324]. Key interface residue mutations and $PB1_N$-derived peptides inhibit viral replication and transcription, suggesting a crucial role of PA_C-$PB1_N$ interactions in polymerase activity and heterotrimer formation. Therefore, novel chemotherapeutic agents mimicking the $PB1_N$ 3_{10} helix are potential inhibitors of PA_C-$PB1_N$ dimerization. Targeting at the PB1 binding site of PA, two FDA approved medications [327] and several novel small molecules [328–332] have been identified, but none have been approved as anti-influenza drugs.

11.1.3 Polymerase basic protein 2 (PB2)

Transcription of influenza virus can be divided into the following stages [333]: 1) binding of the 5' and 3' vRNA sequences to PB1, probably causing a conformational change in the polymerase complex; 2) binding of the 5' cap of a host pre-mRNA to PB2; 3) cleavage of a phosphodiester bond 10 to 13 nucleotides downstream of the cap by PA; and 4) activation of the viral mRNA transcription at the cleaved 3' end of the capped fragment.

PB2 residues 318 to 483 form the cap-binding domain, which is essential for cap-dependent transcription by viral RNPs *in vitro* and *in vivo*. Figure 11.3 shows the 2.3Å resolution crystal structure of the influenza A H3N2 PB2 cap-binding domain in complex with a 5'cap analog m^7GTP (PDB ID: 2VQZ) [334],

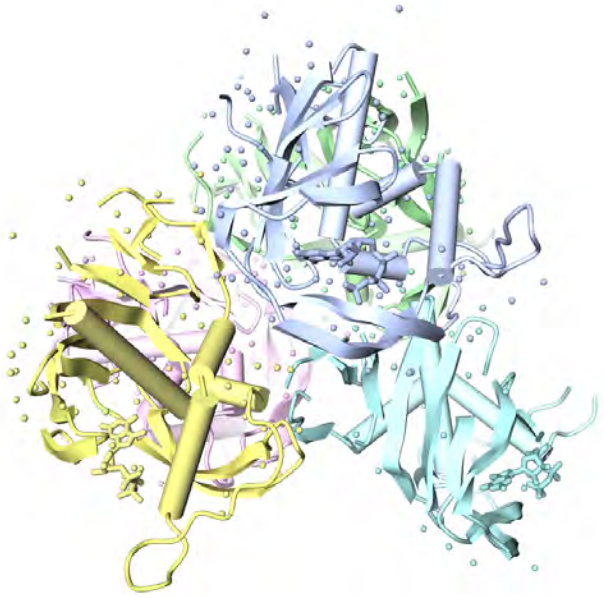


Figure 11.3: Crystal structure of H3N2 PB2 cap binding domain in complex with m^7GTP.

rendered by iview [11]. Other recently solved crystal structures of $PB2_{cap}$ include 4EQK [335], 4NCE [336], 4NCM [336] and 4P1U [336] for the H3N2 strain, 4ENF [335], 3WI0 [337], 3WI1 [337] and 4J2R [338] for the H1N1 strain, and 4ES5 [335], 4CB4 [339], 4CB5 [339], 4CB6 [339] and 4CB7 [339] for the H5N1 strain.

The binding of m^7GTP to $PB2_{cap}$ is assisted by a hydrophobic sandwich between the aromatic residues Phe325 and Phe404. This results in strong electrostatic interactions between the positively charged methylated base m^7G and the aromatic π-electrons, giving up to 100-fold discrimination against a non-methylated base, i.e. m^7GTP versus GTP [334]. On the solvent side of the ligand, the sandwich is completed by His357, which stacks parallel to the base. The key acidic residue Glu361 makes hydrogen

bonds with the N1 and N2 atoms of the guanine. Lys376 is involved in base recognition by interaction with O6. Phe323 stacks on the ribose. His432 and Asn429 interact with the α-phosphate. His357, Lys339 and Arg355 interact with the γ-phosphate. Mutants E361A and K376A had no remaining affinity. Mutants F325A, H357A and F404A had greatly reduced affinity. Mutant F323A had weak binding activity [334].

Phe404 is conserved in influenza B and C viruses, whereas His357, on the other side of the sandwich, is unique to influenza A and is replaced by the more conventional cap-stacking residue tryptophan in influenza B and C [334]. A statistical analysis of 13 key residues using 9246 sequences with unambiguous host annotation showed that most of the 13 residues were highly conserved, except that Lys339 and Arg355 showed polymorphisms in certain subtypes [335].

$PB2_{cap}$ is structurally distinct from other cap binding proteins [334]. Hence $PB2_{cap}$ appears to be a favorable drug target for the development of new antiviral drugs. Targeting at the $PB2_{cap}$ binding site, several novel small molecules [333, 336, 339] have been identified, but none have been approved as anti-influenza drugs.

11.2 Motivation

The influenza viruses have been constantly mutating into drug-resistent strains. The four existing anti-influenza drugs gradually lose their effectiveness. New drugs targeting novel viral

proteins are thus highly desired. The key residues located in the tail-loop binding groove of NP, the $PB1_N$ binding pocket of PA_C and the cap-binding domain of PB2 are highly conserved, and therefore serve as attractive drug targets. Although some inhibitors have been proposed, designed, synthesized and evaluated in some cases, none have been firmly established as new anti-influenza drugs. In this study, we attempted to discover novel anti-influenza small-molecule inhibitors targeting the above three conserved sites on three viral proteins, and hopefully optimize the inhibitors into approved drugs.

11.3 Objective

We aimed at the discovery of anti-influenza small molecules. Particularly, we utilized our docking tool idock [7, 9] to perform structure-based virtual screening, as well as our visualization tool iview [11] to analyze intermolecular interactions.

11.4 Methods

We downloaded the X-ray crystallographic structures of NP trimer (PDB ID: 2IQH) [312], PA_C in complex with $PB1_N$ (PDB ID: 2ZNL) [321], and $PB2_{cap}$ in complex with m^7GTP (PDB ID: 2VQZ) [334]. For the 2IQH NP trimer structure, only chain A was retained and chains B and C were removed. For the 2ZNL PA_C-$PB1_N$ structure, only PA_C was retained and $PB1_N$ was removed. For the 2VQZ $PB2_{cap}$-m^7GTP structure, only PB2 chain

Table 11.1: Search space defined for docking the 2IQH, 2ZNL and 2VQZ structures.

PDB ID	center_x	center_y	center_z	size_x	size_y	size_z
2IQH	82.440	100.261	26.307	30	24	22
2ZNL	-7.965	-56.681	20.928	30	24	24
2VQZ	46.857	24.018	-30.614	16	14	18

A was retained and PB2 chains B, D, E, F, and m^7GTP were removed. Table 11.1 lists the centers and sizes of the docking space of the three structures.

We collected 273,880 compounds from version 2013-02-19 of the Specs catalog of the ZINC database [27, 28]. The Specs compounds were chosen because they are readily available and commercially cheap.

We then executed idock v2.1.3 [9] with a grid map granularity of 0.08Å. Each compound was docked against the specified binding site of each of the three viral proteins, and was subsequently ranked according to the predicted idock score in kcal/mol units or the predicted RF-Score-v3 in pKd units. Finally the top hits were structurally examined with the help of iview [11].

11.5 Results

11.5.1 Nucleoprotein (NP)

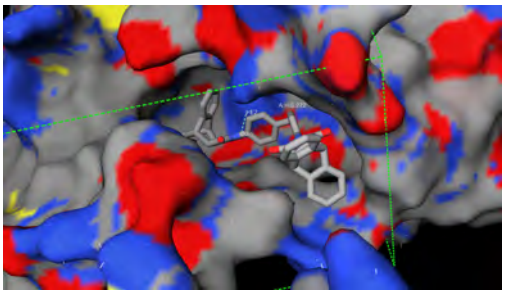
Table 11.2 lists the top hits targeting at the NP tail-loop binding groove. The best idock score obtained was -15.78 kcal/mol, which translates to K_d = 2.66 pM (picomolar). The best RF-Score-v3 obtained was 9.79 pKd, which translates to K_d = 0.16

Table 11.2: Predicted top ten NP-tail-loop-binding-groove-targeted compounds ranked by idock score (top half) and RF-Score-v3 (bottom half).

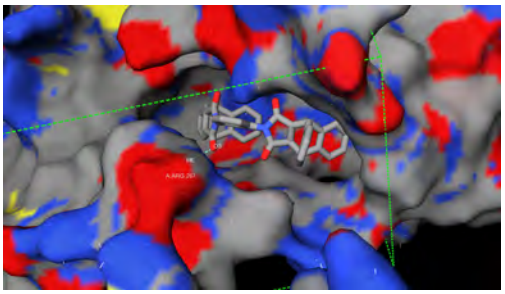
ZINC ID	idock score (kcal/mol)	RF-Score-v3 (pKd)
08398177	-15.78	8.50
04527915	-14.79	7.54
08427160	-14.51	8.29
08448951	-14.45	8.62
08443691	-14.40	8.71
08455791	-14.22	8.49
08425107	-14.16	8.22
08453194	-14.07	8.10
08443534	-13.99	8.21
08442491	-13.96	8.37
08384690	-9.76	9.79
08384620	-9.25	9.68
06143179	-8.97	9.56
08430094	-9.08	9.35
08432234	-8.18	9.32
08384589	-10.31	9.30
08399495	-10.01	9.27
08399544	-10.35	9.24
08399490	-10.09	9.21
08384414	-10.66	9.20

nM (nanomolar).

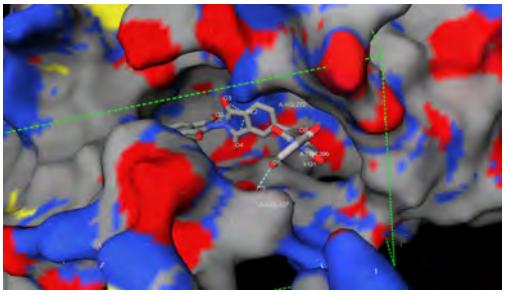
Figure 11.4 visualizes the predicted structures of NP in complex of the top four compounds ranked by idock score and the top four compounds ranked by RF-Score-v3. Putative hydrogen bonds are shown as cyan dashed lines. For instance, the O3 atom of ZINC08398177 forms a putative hydrogen bond with the HE2 atom of HIS272 at a distance of 2.05Å.



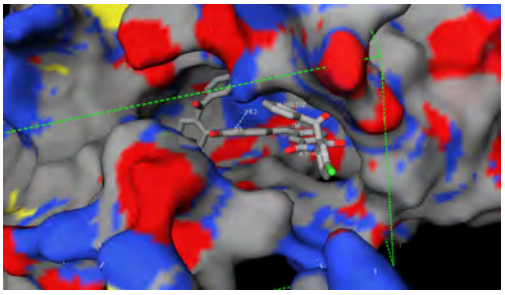
(a) ZINC08398177 forms a hydrogen bond with HIS272.



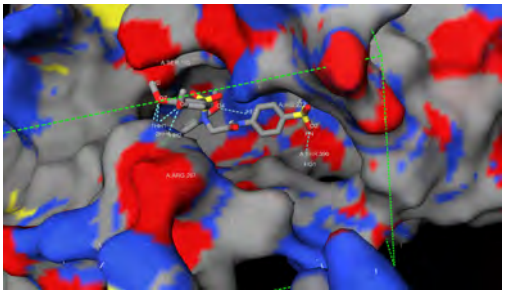
(b) ZINC04527915 forms a hydrogen bond with ARG267.



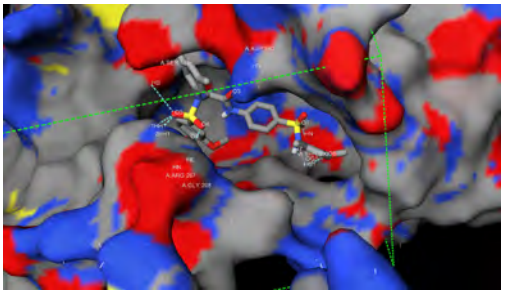
(c) ZINC08427160 forms 5 hydrogen bonds with HIS272, THR390 and SER457.



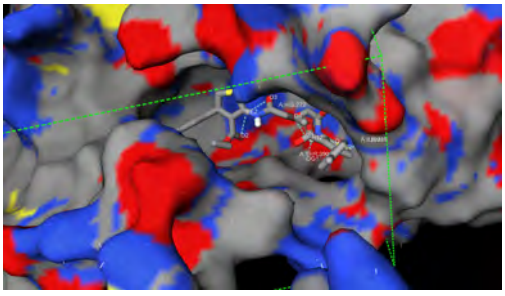
(d) ZINC08448951 forms 2 hydrogen bonds with HIS272 and THR390.



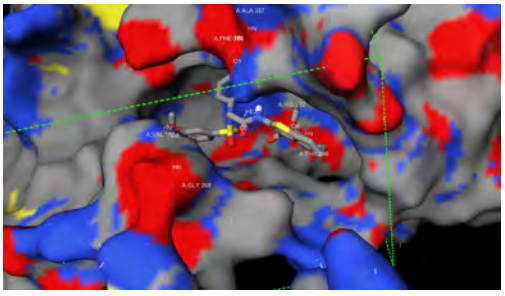
(e) ZINC08384690 forms 8 hydrogen bonds with SER165, ARG267, HIS272 and THR390.



(f) ZINC08384620 forms 9 hydrogen bonds SER165, ARG267, GLY268, ASP340 and THR390.



(g) ZINC06143179 forms 6 hydrogen bonds with HIS272, ILE388 and THR390.



(h) ZINC08430094 forms 6 hydrogen bonds with VAL186, GLY268, HIS272, ALA337, PHE338 and THR390.

Figure 11.4: Predicted structures of NP in complex of the top four compounds ranked by idock score (a to d) and the top four compounds ranked by RF-Score-v3 (e to h).

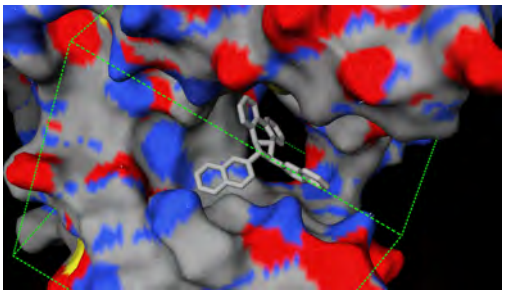
Table 11.3: Predicted top ten PA_C-targeted compounds ranked by idock score (top half) and RF-Score-v3 (bottom half).

ZINC ID	idock score (kcal/mol)	RF-Score-v3 (pKd)
08383903	-15.45	8.24
08398361	-14.53	8.01
08417364	-14.09	8.53
04176726	-13.85	7.96
16526187	-13.84	7.57
00652029	-13.79	7.84
08453194	-13.76	8.01
08427363	-13.67	7.16
08444058	-13.61	8.24
04527916	-13.49	8.05
08439610	-11.20	8.90
08396899	-10.81	8.88
08384461	-9.75	8.84
08443595	-11.59	8.83
08399680	-9.78	8.77
08454594	-11.78	8.74
08399439	-10.26	8.72
08442229	-10.21	8.71
08398784	-10.88	8.70
08397557	-11.66	8.69

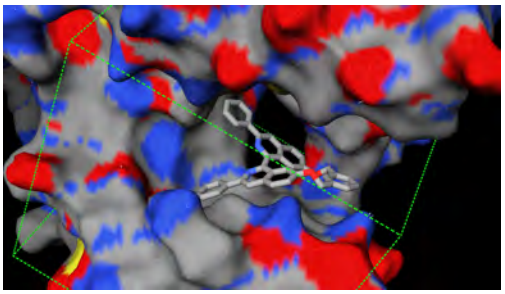
11.5.2 Polymerase acidic protein (PA)

Table 11.3 lists the top hits targeting at the $PB1_N$ binding site of PA_C. The best idock score obtained was -15.45 kcal/mol, which translates to $K_d = 4.65$ pM. The best RF-Score-v3 obtained was 8.90 pKd, which translates to $K_d = 1.26$ nM.

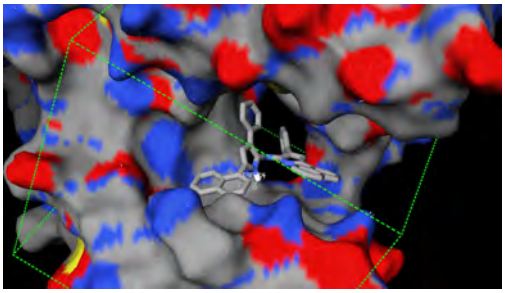
Figure 11.5 visualizes the predicted structures of PA_C in complex of the top four compounds ranked by idock score and the top four compounds ranked by RF-Score-v3.



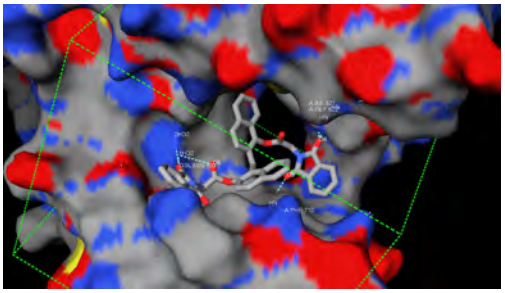
(a) ZINC08383903 forms no hydrogen bond.



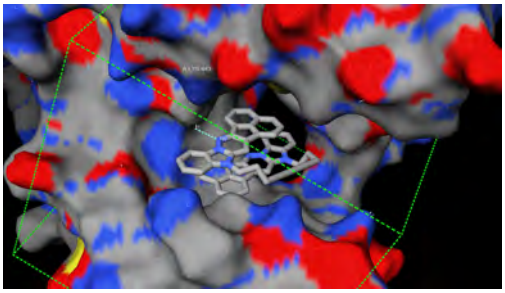
(b) ZINC08398361 forms no hydrogen bond.



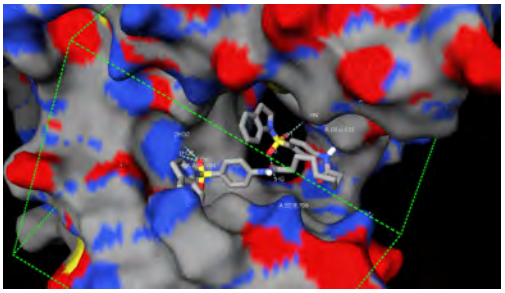
(c) ZINC08417364 forms a hydrogen bond with TRP706.



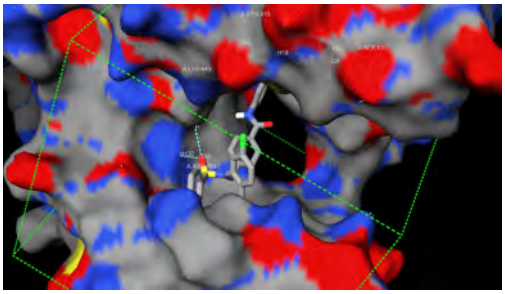
(d) ZINC04176726 forms 6 hydrogen bonds with ILE621, GLY622, ASN703 and PHE710.



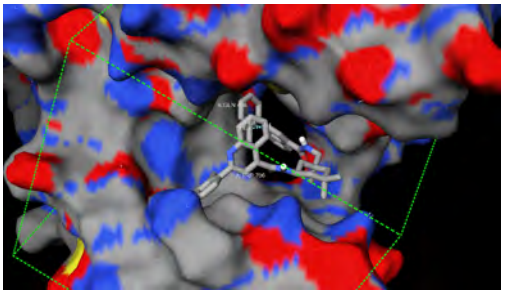
(e) ZINC08439610 forms a hydrogen bond with LYS643.



(f) ZINC08396899 forms 4 hydrogen bonds with GLU623, ASN703 and SER709.



(g) ZINC08384461 forms 4 hydrogen bonds with SER631, LYS643, ASN703 and CYS415.



(h) ZINC08443595 forms 2 hydrogen bonds with GLN408 and TRP706.

Figure 11.5: Predicted structures of PA_C in complex of the top four compounds ranked by idock score (a to d) and the top four compounds ranked by RF-Score-v3 (e to h).

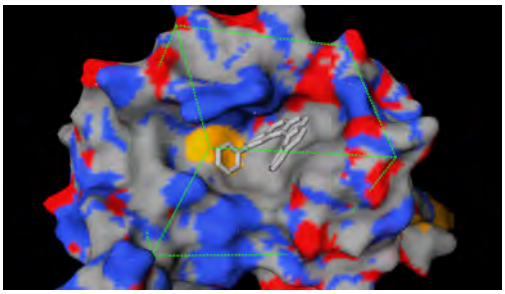
Table 11.4: Predicted top ten $PB2_{cap}$-targeted compounds ranked by idock score (top half) and RF-Score-v3 (bottom half).

ZINC ID	idock score (kcal/mol)	RF-Score-v3 (pKd)
08383936	-12.52	7.67
08386295	-12.15	7.41
03015113	-11.85	7.67
15188425	-11.74	7.53
02125231	-11.69	7.09
02154274	-11.56	7.31
08453194	-11.56	7.40
02077599	-11.46	7.21
02077477	-11.45	7.36
00657519	-11.45	7.65
08439610	-8.77	8.47
08439605	-8.59	8.38
08399683	-7.38	8.37
08437929	-9.27	8.33
08455074	-9.81	8.25
08444191	-9.56	8.23
08446351	-9.96	8.20
08452812	-9.49	8.20
02752464	-10.58	8.19
08446353	-8.82	8.19

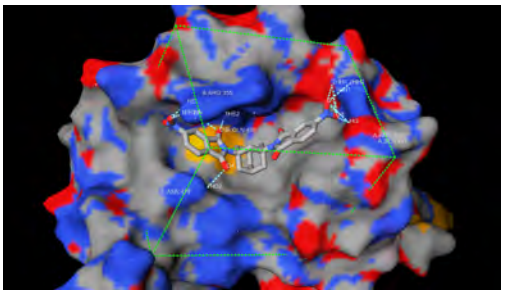
11.5.3 Polymerase basic protein 2 (PB2)

Table 11.4 lists the top hits targeting at the cap binding site of PB2. The best idock score obtained was -12.52 kcal/mol, which translates to $K_d = 0.66$ nM. The best RF-Score-v3 obtained was 8.47 pKd, which translates to $K_d = 3.39$ nM.

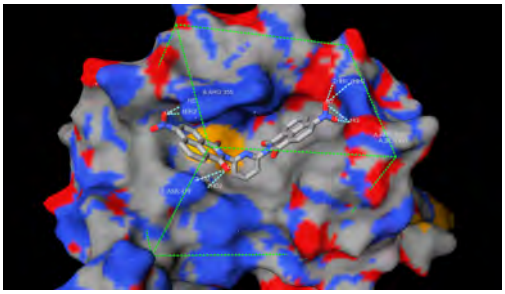
Figure 11.6 visualizes the predicted structures of $PB2_{cap}$ in complex of the top four compounds ranked by idock score and the top four compounds ranked by RF-Score-v3.



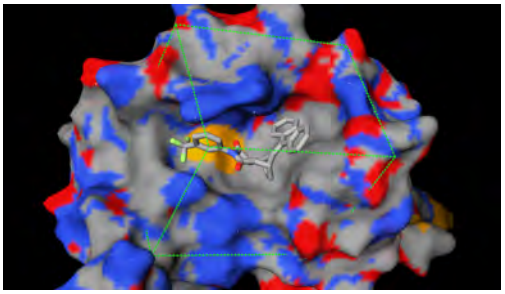
(a) ZINC08383936 forms no hydrogen bond.



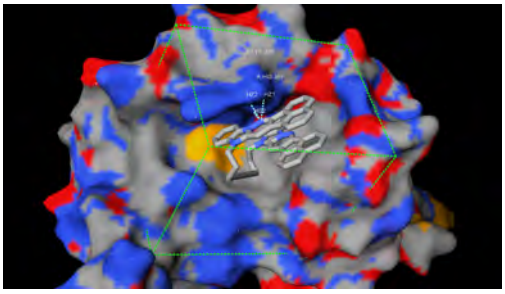
(b) ZINC08386295 forms 11 hydrogen bonds with SER321, ARG332, ARG355, GLN406 and ASN429.



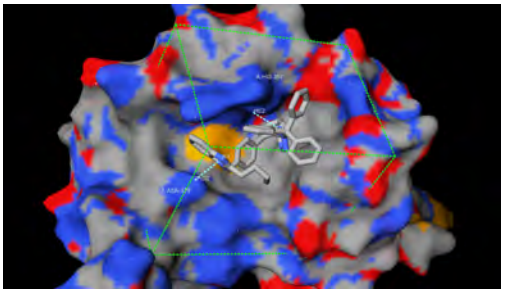
(c) ZINC03015113 forms 8 hydrogen bonds with SER321, ARG332, ARG355 and ASN429.



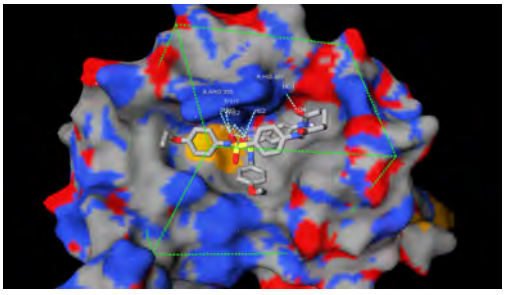
(d) ZINC15188425 forms no hydrogen bond.



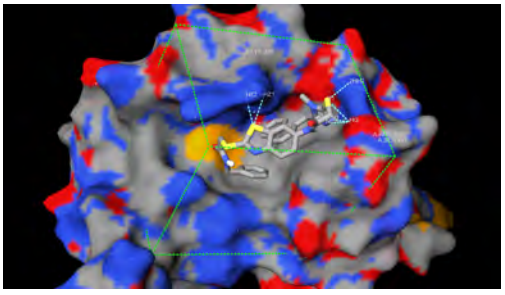
(e) ZINC08439610 forms 3 hydrogen bonds with LYS339 and HIS357.



(f) ZINC08439605 forms 2 hydrogen bonds with HIS357 and ASN429.



(g) ZINC08399683 forms 6 hydrogen bonds with ARG355, HIS357 and GLN406.



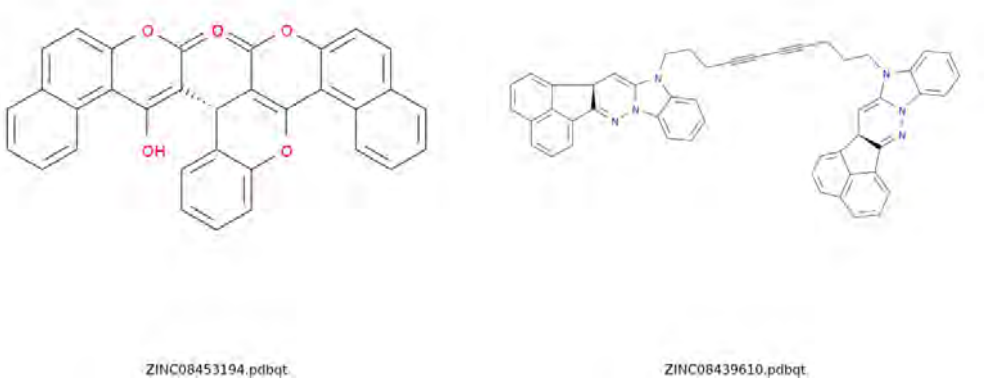
(h) ZINC08437929 forms 6 hydrogen bonds with SER321, ARG332 and LYS339.

Figure 11.6: Predicted structures of $PB2_{cap}$ in complex of the top four compounds ranked by idock score (a to d) and the top four compounds ranked by RF-Score-v3 (e to h).

11.6 Discussion

For the top hits in all the three docking cases, the binding affinities predicted by idock are generally at pM level, whereas those predicted by RF-Score-v3 are generally at nM level. This means idock tends to estimate a significantly higher binding affinity for a top hit than RF-Score-v3 does. The top hits of NP and PA_C have significantly higher binding affinities than those of $PB2_{cap}$ on average, probably because the chosen binding sites of NP and PA_C are remarkably larger in volume, as can be seen in Table 11.1. In contrast, the $PB2_{cap}$ binding site locates at the protein surface and constitutes a shallow pocket. Apparently a larger binding site permits a more comprehensive exploration of conformational flexibility of the ligand.

In each of the three docking cases, there is no duplicate compound in both the top ten hits ranked by idock score and the top ten hits ranked by RF-Score-v3. This suggests the two scores tend to prioritize compounds differently. However, interestingly, ZINC08453194 appears in the idock top ten lists across all the three cases, and ZINC08439610 appears in the RF-Score-v3 top ten lists across the cases of PA_C and $PB2_{cap}$. Their chemical structures are shown in Figure 11.7. A certain type of scoring function seems to favor some certain atom types. ZINC08453194 contains mostly carbon atoms and five oxygen atoms, while ZINC08439610 contains mostly carbon atoms and six nitrogen atoms. Besides, looking at the quasi-linear formula to calculate idock score [7–9], the number of rotatable bonds,



(a) ZINC08453194 appears in the idock top ten lists across the docking cases of NP, PA_Cv3 and $PB2_{cap}$.

(b) ZINC08439610 appears in the RF-Score top ten lists across the docking cases of PA_C and $PB2_{cap}$.

Figure 11.7: Duplicate top hits across docking cases.

denoted as N_{rot}, is in the denominator so as to penalize ligand flexibility, therefore idock score also favors ligands that have few rotatable bonds, such as ZINC08453194 which has just one rotatable bond. Taken together, these indicate that some particular ligand-only properties already guarantee that the ligand will be ranked high when docking against certain types of proteins.

11.7 Conclusions

Treatment of seasonal and pandemic influenza is currently limited by the availability of only few drugs that are challenged by emergence of drug-resistant mutants. In this study, we have targeted at novel binding sites of three viral proteins, namely

the tail-loop binding domain of NP, the PB1-binding domain of PA, and the cap-binding domain of PB2, and reported the top hits obtained from structure-based virtual screening of 273,880 cheaply available compounds from Specs using idock [7, 9] as the molecular docking tool and iview [11] as the interaction visualization tool. These top hits may serve as potential promising starting points for subsequent wet experiments. Moreover, we have observed that idock tends to output a substantially higher binding affinity for a top hit than RF-Score-v3 does, and certain types of scoring functions tend to favor certain ligand-only properties, which permit certain ligands to be ranked high regardless of the protein in study.

11.8 Future works

Recently, two apo crystal structures of H1N1 and H7N9 PA_C (PDB ID: 4IUJ and 4P9A) in the absence of PB1 have been reported to exhibit the same global topology as other strains, but differ extensively in the PB1 binding pocket [322]. These structural changes demonstrate plasticity in the PA_C-$PB1_N$ binding interface, which can be exploited in the development of novel therapeutic drugs. Furthermore, the structure of polymerase has recently been solved [340, 341], which provides valuable structural insights for drug design.

The top scoring compounds reported in this study will be subjected to post-screening evaluations, including Lipinski's rule filter [83], visual inspection and consensus docking [342] using

DOCK [39], AutoDock Vina [8], or PLANTS [42–44]. The commercially available compounds will be purchased for subsequent biological evaluations.

The cytotoxicity of the compounds will first be tested by MTT assay. Influenza RNP reconstitution assay will then be performed to investigate their ability to inhibit RNP transcriptional activity. Hit compounds causing significant reduction of RNP activity will be subjected to whole virus assay including plaque reduction assay and yield reduction assay using seasonal flu viruses. Surface plasmon resonance will also be performed to test the *in vitro* binding affinity of the compounds to the target protein.

For compounds that exhibit substantial anti-influenza properties, chemical analogues will be purchased for further evaluation. Structure activity relationship study will be performed to further characterize the interaction between the compound and the target protein.

□ **End of chapter.**

Chapter 12

Conclusions

Drug discovery has been an expensive and long-term practice over the decades. The cost of drug development has now reached US$2.6B [2]. On the other hand, computer-aided drug discovery (CADD) methods are becoming cheaper and faster, and their predictive accuracy are continuously improving. This thesis presents our pragmatic CADD toolset as well as its prospective applications.

Chapter 2 describes idock [7] for multithreaded flexible ligand docking. idock adopts a substantially simplified numerical model and implements dimension reduction for stochastic optimization. Compared to the competitive docking tool AutoDock Vina [8], idock obtains a speedup of 3.3 in CPU time and a speedup of 7.5 in elapsed time on average. A faster implementation permits testing more compounds or finding lower energy conformations in a large virtual screen.

Chapter 3 describes istar [9] as a heterogeneous web platform for hosting diverse web services from multiple disciplines, in-

cluding idock for large-scale prospective structure-based virtual screening. istar features a huge molecular database of over 23 million compounds, and provides comfortable and unique user experience via the proper use of modern web technologies. istar is now getting more attentions worldwide according to Google Analytics.

Chapter 4 describes iview [11] for quick elucidation of molecular interactions on web pages interactively. iview eliminates the prerequisite of Java in browsers and utilizes WebGL instead, enabling GPU hardware acceleration. iview supports the helpful features of macromolecular surface construction and virtual reality effects. iview is also highly customizable that a specific version for visualizing idock results is derived and deployed on istar.

Chapter 5 describes iSyn [12, 13] for generating potent compounds *de novo* from molecular fragments with desired molecular properties. iSyn circumvents the compound database diversity limitation imposed by virtual screening methods. iSyn guarantees synthetic feasibility with click chemistry, and interfaces with idock and iview to provide consistent experience. iSyn is capable of producing extraordinarily novel compounds within a reasonable runtime.

Chapters 6, 7 and 8 describe our separate studies [15–18] on the use of random forest (RF) to improve binding affinity prediction with related but different motivations. We have shown that the simple functional form typically implemented in classical scoring functions is detrimental for their predictive performance

due to their incapability of exploiting abundant training samples, and substituting machine learning techniques like RF for the commonly-used multiple linear regression (MLR) model can substantially improve predictive accuracy [15–17]. This finding is significant because RF-based scoring functions will continue to gain their competitive edge over MLR-based scoring functions given the future availability of more experimental data. We have also investigated the impact of pose generation error on the predictive performance and found that re-training the scoring functions on docked poses can be a simple and quick solution to reduce the negative impact of pose generation error [18].

Chapter 9 describes USR@istar for convenient identification for compounds structurally similar to a query using the ultrafast shape recognition algorithm USR [19] and its extension USRCAT [20]. As a novel feature, our USR@istar exploits the AVX SIMD instructions of modern processors to accelerate similarity score computation. As many as 19 sample query ligands with different molecular sizes have been selected to benchmark USR@istar. To our expectation, USR and USRCAT prioritize completely different compounds when the query has a large number of heavy atoms. To our surprise, however, different file formats of the same query ligand yield different output. With the calculated features preloaded on the server side, searching 23 million compounds requires merely 30 seconds on average, compared to 167 seconds when the precalculated features are loaded *ad hoc*.

It is noteworthy that all of our CADD tools are free and open source so as to promote their use. In addition to tool development, described in the chapters above, we also emphasize prospective applications, presented in the chapters below.

Chapter 10 presents our case study of cancers related to CDK2 (cyclin-dependent kinase 2). We have utilized idock [7, 9] and [11] iview prospectively for the first time in identifying potential CDK2 inhibitors from approved small molecule drugs using a repurposing strategy and an ensemble docking methodology. The anti-acne drug adapalene exhibits the anti-proliferative effect in human colon cancer *in vitro* and significantly inhibited tumor growth *in vivo* in nude mice subcutaneously xenografted with human colorectal cancer cells, rendering adapalene a candidate anti-cancer drug.

Chapter 11 presents our case study of influenza A. We have selected three novel protein targets and utilized idock [7, 9] to screen 273,880 commercially cheap compounds, and identify hits predicted to establish strong interactions with their respective viral protein target and hence believed to yield strong inhibitory effects.

In conclusion, we believe our toolset constitutes a step toward generalizing the use of CADD tools beyond the traditional purely experimental community, and our successful drug discovery endeavors in real life would hopefully inspire researchers in the CADD field.

□ **End of chapter.**

Appendix A

Publications

A.1 Journal publications

1) **Hongjian Li**, Kwong-Sak Leung, Pedro J. Ballester and Man-Hon Wong. istar: A Web Platform for Large-Scale Protein-Ligand Docking. *PLoS ONE*, 9(1):e85678, 2014.

2) **Hongjian Li**, Kwong-Sak Leung, Takanori Nakane and Man-Hon Wong. iview: an interactive WebGL visualizer for protein-ligand complex. *BMC Bioinformatics*, 15(1):56, 2014.

3) **Hongjian Li**, Kwong-Sak Leung, Man-Hon Wong and Pedro J. Ballester. Substituting random forest for multiple linear regression improves binding affinity prediction of scoring functions: Cyscore as a case study. *BMC Bioinformatics*, 15(1):291, 2014.

4) **Hongjian Li**, Kwong-Sak Leung, Man-Hon Wong and Pedro J. Ballester. Improving AutoDock Vina using Random Forest: the growing accuracy of binding affinity prediction by the effective exploitation of larger data sets. *Molecular Informatics*, 34(2-3):115-126, 2015.

5) Xi-Nan Shi, **Hongjian Li**, Hong Yao, Xu Liu, Ling Li, Kwong-Sak Leung, Hsiang-fu Kung, Man-Hon Wong and Marie Chia-mi Lin. Adapalene Inhibited the Activity of Cyclin-Dependent Kinase 2 in Colorectal Carcinoma. *Molecular Medicine Reports*, in press.

6) **Hongjian Li**, Kwong-Sak Leung, Man-Hon Wong and Pedro J. Ballester. The importance of the regression model in the structure-based prediction of protein-ligand binding. *Lecture Notes in Bioinformatics*, in press.

7) **Hongjian Li**, Kwong-Sak Leung, Man-Hon Wong and Pedro J. Ballester. The impact of docking pose generation error on the prediction of binding affinity. *Lecture Notes in Bioinformatics*, in press.

A.2 Conference publications

1) **Hongjian Li**, Kwong-Sak Leung and Man-Hon Wong. idock: A Multithreaded Virtual Screening Tool for Flexible Ligand Docking. In *Proceedings of the 2012 IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB)*, San Diego, US, 9-12 May 2012.

2) **Hongjian Li**, Kwong-Sak Leung, Man-Hon Wong and Pedro J. Ballester. The importance of the regression model in the prediction of intermolecular binding using AutoDock Vina. In *Proceedings of the 11th International Meeting on Computational Intelligence Methods for Bioinformatics and Biostatistics (CIBB)*, Cambridge, UK, 26-28 June 2014.

3) **Hongjian Li**, Kwong-Sak Leung, Man-Hon Wong and Pedro J. Ballester. The impact of docked pose generation error on the accuracy of machine-learning scoring functions. In *Proceedings of the 11th International Meeting on Computational Intelligence Methods for Bioinformatics and Biostatistics (CIBB)*, Cambridge, UK, 26-28 June 2014.

4) **Hongjian Li**, Kwong-Sak Leung, Chun Ho Chan, Hei Lun Cheung and Man-Hon Wong. iSyn: De Novo Drug Design with Click Chemistry Support. In *Proceedings of the 2014 Conference Companion on Genetic and Evolutionary Computation Conference (GECCO)*, Vancouver, Canada, 12-16 July 2014.

5) **Hongjian Li**, Kwong-Sak Leung, Chun Ho Chan, Hei Lun Cheung and Man-Hon Wong. iSyn: WebGL-based interactive de novo drug design. In *Proceedings of the 18th International Conference on Information Visualisation (IV)*, Paris, France, 15-18 July 2014.

□ **End of chapter.**

Bibliography

[1] Steve Morgan, Paul Grootendorst, Joel Lexchin, Colleen Cunningham, and Devon Greyson. The cost of drug development: A systematic review. *Health Policy*, 100(1):4–17, 2011.

[2] Asher Mullard. New drugs cost US$2.6 billion to develop. *Nature Reviews Drug Discovery*, 13(12):877–877, 2014.

[3] Steven M. Paul, Daniel S. Mytelka, Christopher T. Dunwiddie, Charles C. Persinger, Bernard H. Munos, Stacy R. Lindborg, and Aaron L. Schacht. How to improve R&D productivity: the pharmaceutical industry's grand challenge. *Nature Reviews Drug Discovery*, 9(3):203–214, 2010.

[4] Michael S. Kinch, Austin Haynesworth, Sarah L. Kinch, and Denton Hoyer. An overview of FDA-approved new molecular entities: 1827–2013. *Drug Discovery Today*, 19 (8):1033–1039, 2014.

[5] Thomas Lengauer, André Altmann, Alexander Thielen, and Rolf Kaiser. Chasing the AIDS virus. *Communications of the ACM*, 53(3):66–74, 2010.

[6] Roger D. Peng. Reproducible Research in Computational Science. *Science*, 334(6060):1226–1227, 2011.

[7] Hongjian Li, Kwong-Sak Leung, and Man-Hon Wong. idock: A multithreaded virtual screening tool for flexible

ligand docking. In *Proceedings of the 2012 IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB)*, pages 77–84, 2012.

[8] Oleg Trott and Arthur J. Olson. AutoDock Vina: Improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading. *Journal of Computational Chemistry*, 31(2):455–461, 2010.

[9] Hongjian Li, Kwong-Sak Leung, Pedro J. Ballester, and Man-Hon Wong. istar: A Web Platform for Large-Scale Protein-Ligand Docking. *PLoS ONE*, 9(1):e85678, 2014.

[10] Pedro J. Ballester and John B. O. Mitchell. A machine learning approach to predicting protein-ligand binding affinity with applications to molecular docking. *Bioinformatics*, 26(9):1169–1175, 2010.

[11] Hongjian Li, Kwong-Sak Leung, Takanori Nakane, and Man-Hon Wong. iview: an interactive WebGL visualizer for protein-ligand complex. *BMC Bioinformatics*, 15(1): 56, 2014.

[12] Hongjian Li, Kwong-Sak Leung, Chun Ho Chan, Hei Lun Cheung, and Man-Hon Wong. iSyn: De Novo Drug Design with Click Chemistry Support. In *Proceedings of the 2014 Conference Companion on Genetic and Evolutionary Computation Companion (GECCO)*, pages 43–44, 2014.

[13] Hongjian Li, Kwong-Sak Leung, Chun Ho Chan, Hei Lun Cheung, and Man-Hon Wong. iSyn: WebGL-based interactive de novo drug design. In *Proceedings of the 18th International Conference on Information Visualisation (IV)*, pages 302–307, 2014.

[14] Yang Cao and Lei Li. Improved protein–ligand binding affinity prediction by using a curvature-dependent surface-area model. *Bioinformatics*, 30(12):1674–1680, 2014.

[15] Hongjian Li, Kwong-Sak Leung, Man-Hon Wong, and Pedro Ballester. Substituting random forest for multiple linear regression improves binding affinity prediction of scoring functions: Cyscore as a case study. *BMC Bioinformatics*, 15(1):291, 2014.

[16] Hongjian Li, Kwong-Sak Leung, Man-Hon Wong, and Pedro J. Ballester. Improving AutoDock Vina Using Random Forest: The Growing Accuracy of Binding Affinity Prediction by the Effective Exploitation of Larger Data Sets. *Molecular Informatics*, 34(2-3):115–126, 2015.

[17] Hongjian Li, Kwong-Sak Leung, Man-Hon Wong, and Pedro J. Ballester. The importance of the regression model in the prediction of intermolecular binding using AutoDock Vina. In *Proceedings of the 11th International Meeting on Computational Intelligence Methods for Bioinformatics and Biostatistics (CIBB)*, pages 144–149, 2014.

[18] Hongjian Li, Kwong-Sak Leung, Man-Hon Wong, and Pedro J. Ballester. The impact of docked pose generation error on the accuracy of machine-learning scoring functions. In *Proceedings of the 11th International Meeting on Computational Intelligence Methods for Bioinformatics and Biostatistics (CIBB)*, pages 138–143, 2014.

[19] Pedro J. Ballester and W. Graham Richards. Ultrafast shape recognition to search compound databases for similar molecular shapes. *Journal of Computational Chemistry*, 28(10):1711–1723, 2007.

[20] Adrian Schreyer and Tom Blundell. USRCAT: real-time ultrafast shape recognition with pharmacophoric constraints. *Journal of Cheminformatics*, 4(1):27, 2012.

[21] Helen M. Berman, T. N. Bhat, Philip E. Bourne, Zukang Feng, Gary Gilliland, Helge Weissig, and John Westbrook. The Protein Data Bank and the challenge of structural genomics. *Nature Structural & Molecular Biology*, 7:957–959, 2000.

[22] Helen Berman, Kim Henrick, and Haruki Nakamura. Announcing the worldwide Protein Data Bank. *Nature Structural & Molecular Biology*, 10(12):980–980, 2003.

[23] Anna Gaulton, Louisa J. Bellis, A. Patricia Bento, Jon Chambers, Mark Davies, Anne Hersey, Yvonne Light, Shaun McGlinchey, David Michalovich, Bissan Al-Lazikani, and John P. Overington. ChEMBL: a large-scale bioactivity database for drug discovery. *Nucleic Acids Research*, 40(D1):D1100–D1107, 2012.

[24] Yanli Wang, Jewen Xiao, Tugba O. Suzek, Jian Zhang, Jiyao Wang, Zhigang Zhou, Lianyi Han, Karen Karapetyan, Svetlana Dracheva, Benjamin A. Shoemaker, Evan Bolton, Asta Gindulyte, and Stephen H. Bryant. PubChem's BioAssay Database. *Nucleic Acids Research*, 40 (D1):D400–D412, 2012.

[25] Lars Ruddigkeit, Ruud van Deursen, Lorenz C. Blum, and Jean-Louis Reymond. Enumeration of 166 Billion Organic Small Molecules in the Chemical Universe Database GDB-17. *Journal of Chemical Information and Modeling*, 52(11): 2864–2875, 2012.

[26] Yanli Wang, Jewen Xiao, Tugba O. Suzek, Jian Zhang, Jiyao Wang, and Stephen H. Bryant. PubChem: a pub-

lic information system for analyzing bioactivities of small molecules. *Nucleic Acids Research*, 37(suppl_2):W623–W633, 2009.

[27] John J. Irwin and Brian K. Shoichet. ZINC - A Free Database of Commercially Available Compounds for Virtual Screening. *Journal of Chemical Information and Modeling*, 45(1):177–182, 2005.

[28] John J. Irwin, Teague Sterling, Michael M. Mysinger, Erin S. Bolstad, and Ryan G. Coleman. ZINC: A Free Tool to Discover Chemistry for Biology. *Journal of Chemical Information and Modeling*, 52(7):1757–1768, 2012.

[29] Calvin Yu-Chian Chen. TCM Database@Taiwan: The World's Largest Traditional Chinese Medicine Database for Drug Screening In Silico. *PLoS ONE*, 6(1):e15939, 2011.

[30] Eric F. Pettersen, Thomas D. Goddard, Conrad C. Huang, Gregory S. Couch, Daniel M. Greenblatt, Elaine C. Meng, and Thomas E. Ferrin. UCSF Chimera - A visualization system for exploratory research and analysis. *Journal of Computational Chemistry*, 25(13):1605–1612, 2004.

[31] William Humphrey, Andrew Dalke, and Klaus Schulten. VMD: Visual molecular dynamics. *Journal of Molecular Graphics*, 14(1):33–38, 1996.

[32] Garrett M. Morris, Ruth Huey, William Lindstrom, Michel F. Sanner, Richard K. Belew, David S. Goodsell, and Arthur J. Olson. AutoDock4 and AutoDockTools4: Automated docking with selective receptor flexibility. *Journal of Computational Chemistry*, 30(16):2785–2791, 2009.

[33] Christopher D. Lau, Marshall J. Levesque, Shu Chien, Susumu Date, and Jason H. Haga. ViewDock TDW: high-throughput visualization of virtual screening results. *Bioinformatics*, 26(15):1915–1917, 2010.

[34] Katrin Stierand and Matthias Rarey. PoseView - molecular interaction patterns at a glance. *Journal of Cheminformatics*, 2(Suppl 1):P50, 2010.

[35] Roman A. Laskowski and Mark B. Swindells. LigPlot+: Multiple Ligand-Protein Interaction Diagrams for Drug Discovery. *Journal of Chemical Information and Modeling*, 51(10):2778–2786, 2011.

[36] Guillaume Bouvier, Nathalie Evrard-Todeschi, Jean-Pierre Girault, and Gildas Bertho. Automatic clustering of docking poses in virtual screening process using self-organizing map. *Bioinformatics*, 26(1):53–60, 2010.

[37] Jean-François Truchon and Christopher I. Bayly. Evaluating Virtual Screening Methods: Good and Bad Metrics for the "Early Recognition" Problem. *Journal of Chemical Information and Modeling*, 47(2):488–508, 2007.

[38] Wei Zhao, Kirk Hevener, Stephen White, Richard Lee, and James Boyett. A statistical framework to evaluate virtual screening. *BMC Bioinformatics*, 10(1):225, 2009.

[39] P. Therese Lang, Scott R. Brozell, Sudipto Mukherjee, Eric F. Pettersen, Elaine C. Meng, Veena Thomas, Robert C. Rizzo, David A. Case, Thomas L. James, and Irwin D. Kuntz. DOCK 6: Combining techniques to model RNA–small molecule complexes. *RNA*, 15(6):1219–1230, 2009.

[40] Ryan G. Coleman, Michael Carchia, Teague Sterling, John J. Irwin, and Brian K. Shoichet. Ligand Pose and

Orientational Sampling in Molecular Docking. *PLoS ONE*, 8(10):e75992, 2013.

[41] Stephanus Daniel Handoko, Xuchang Ouyang, Chinh Tran To Su, Chee Keong Kwoh, and Yew Soon Ong. QuickVina: Accelerating AutoDock Vina Using Gradient-Based Heuristics for Global Optimization. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 9:1266–1272, 2012.

[42] Oliver Korb, Thomas Stützle, and Thomas Exner. PLANTS: Application of Ant Colony Optimization to Structure-Based Drug Design. In *Ant Colony Optimization and Swarm Intelligence*, volume 4150 of *Lecture Notes in Computer Science*, pages 247–258. Springer Berlin Heidelberg, 2006.

[43] Oliver Korb, Thomas Stützle, and Thomas E. Exner. Empirical Scoring Functions for Advanced Protein-Ligand Docking with PLANTS. *Journal of Chemical Information and Modeling*, 49(1):84–96, 2009.

[44] Oliver Korb, Thomas Stützle, and Thomas E. Exner. Accelerating Molecular Docking Calculations Using Graphics Processing Units. *Journal of Chemical Information and Modeling*, 51(4):865–876, 2011.

[45] Christopher R. Corbeil, Pablo Englebienne, and Nicolas Moitessier. Docking Ligands into Flexible and Solvated Macromolecules. 1. Development and Validation of FITTED 1.0. *Journal of Chemical Information and Modeling*, 47(2):435–449, 2007.

[46] Christopher R. Corbeil, Pablo Englebienne, Constantin G. Yannopoulos, Laval Chan, Sanjoy K. Das, Darius Bilimoria, Lucille L'Heureux, and Nicolas Moitessier. Docking

Ligands into Flexible and Solvated Macromolecules. 2. Development and Application of Fitted 1.5 to the Virtual Screening of Potential HCV Polymerase Inhibitors. *Journal of Chemical Information and Modeling*, 48(4):902–909, 2008.

[47] Álvaro Cortés Cabrera, Javier Klett, Helena Dos Santos G., Almudena Perona, Rub Gil-Redondo, Sandrea M. Francis, Eva M. Priego, Federico Gago, and Antonio Morreale. CRDOCK: An Ultrafast Multipurpose Protein–Ligand Docking Tool. *Journal of Chemical Information and Modeling*, 52(8):2300–2309, 2012.

[48] Claudia Beato, Andrea R. Beccari, Carlo Cavazzoni, Simone Lorenzi, and Gabriele Costantino. Use of Experimental Design To Optimize Docking Performance: The Case of LiGenDock, the Docking Module of Ligen, a New De Novo Design Program. *Journal of Chemical Information and Modeling*, 53(6):1503–1517, 2013.

[49] Bingjie Hu and Markus Lill. PharmDock: a pharmacophore-based docking program. *Journal of Cheminformatics*, 6(1):14, 2014.

[50] Pedro J. Ballester, Adrian Schreyer, and Tom L. Blundell. Does a More Precise Chemical Description of Protein–Ligand Complexes Lead to More Accurate Prediction of Binding Affinity? *Journal of Chemical Information and Modeling*, 54(3):944–955, 2014.

[51] Christoph A. Sotriffer, Paul Sanschagrin, Hans Matter, and Gerhard Klebe. SFCscore: Scoring functions for affinity prediction of protein-ligand complexes. *Proteins: Structure, Function, and Bioinformatics*, 73(2):395–419, 2008.

[52] David Zilian and Christoph A. Sotriffer. SFCscoreRF: A Random Forest-Based Scoring Function for Improved Affinity Prediction of Protein–Ligand Complexes. *Journal of Chemical Information and Modeling*, 53(8):1923–1933, 2013.

[53] Zheng Zheng and Kenneth M. Merz. Ligand Identification Scoring Algorithm (LISA). *Journal of Chemical Information and Modeling*, 51(6):1296–1306, 2011.

[54] Jacob D. Durrant and J. A. McCammon. NNScore 2.0: A Neural-Network Receptor–Ligand Scoring Function. *Journal of Chemical Information and Modeling*, 51(11):2897–2903, 2011.

[55] Guo-Bo Li, Ling-Ling Yang, Wen-Jing Wang, Lin-Li Li, and Sheng-Yong Yang. ID-Score: A New Empirical Scoring Function Based on a Comprehensive Set of Descriptors Related to Protein–Ligand Interactions. *Journal of Chemical Information and Modeling*, 53(3):592–600, 2013.

[56] N. Moitessier, P. Englebienne, D. Lee, J. Lawandi, and C. R. Corbeil. Towards the development of universal, fast and highly accurate docking/scoring methods: a long way to go. *British Journal of Pharmacology*, 153(S1):S7–S26, 2008.

[57] Subha Kalyaanamoorthy and Yi-Ping Phoebe Chen. Structure-based drug design to augment hit discovery. *Drug Discovery Today*, 16(17-18):831–839, 2011.

[58] Max W. Chang, Christian Ayeni, Sebastian Breuer, and Bruce E. Torbett. Virtual Screening for HIV Protease Inhibitors: A Comparison of AutoDock 4 and Vina. *PLoS ONE*, 5(8):e11955, 2010.

[59] Daniel Seeliger and Bert de Groot. Ligand docking and binding site analysis with PyMOL and Autodock/Vina. *Journal of Computer-Aided Molecular Design*, 24(5):417–422, 2010.

[60] Rui Abreu, Hugo Froufe, Maria Queiroz, and Isabel Ferreira. MOLA: a bootable, self-configuring system for virtual screening using AutoDock4/Vina on computer clusters. *Journal of Cheminformatics*, 2(1):10, 2010.

[61] Natsumi Baba and Eiichi Akaho. VSDK: Virtual screening of small molecules using AutoDock Vina on Windows platform. *Bioinformation*, 6(10):387, 2011.

[62] Gaddam Sandeep, Kurre Purna Nagasree, Muppaneni Hanisha, and Muthyala Murali Krishna Kumar. AUDocker LE: A GUI for virtual screening with AUTODOCK Vina. *BMC Research Notes*, 4(1):445, 2011.

[63] Sally R. Ellingson, Jeremy C. Smith, and Jerome Baudry. VinaMPI: Facilitating multiple receptor high-throughput virtual docking on high-performance computers. *Journal of Computational Chemistry*, 34(25):2212–2221, 2013.

[64] Xiaohua Zhang, Sergio E. Wong, and Felice C. Lightstone. Message passing interface and multithreading hybrid for parallel molecular docking of large databases on petascale high performance computing machines. *Journal of Computational Chemistry*, 34(11):915–927, 2013.

[65] Ramaswamy Sree Latha, Ramadoss Vijayaraj, Ettayapuram Ramaprasad Azhagiya Singam, Krishnaswamy Chitra, and Venkatesan Subramanian. 3D-QSAR and Docking Studies on the HEPT Derivatives of HIV-1 Reverse Transcriptase. *Chemical Biology & Drug Design*, 78(3): 418–426, 2011.

[66] Rui M. V. Abreu, Hugo J. C. Froufe, Maria-João R. P. Queiroz, and Isabel C. F. R. Ferreira. Selective Flexibility of Side-Chain Residues Improves VEGFR-2 Docking Score using AutoDock Vina. *Chemical Biology & Drug Design*, 79(4):530–534, 2012.

[67] Sugunadevi Sakkiah, Sundarapandian Thangapandian, Chanin Park, Minky Son, and Keun W. Lee. Molecular Docking and Dynamics Simulation, Receptor-based Hypothesis: Application to Identify Novel Sirtuin 2 Inhibitors. *Chemical Biology & Drug Design*, 80(2):315–327, 2012.

[68] Wayne A. Warner, Ricardo Sanchez, Alex Dawoodian, Esther Li, and Jamil Momand. Identification of FDA-approved Drugs that Computationally Bind to MDM2. *Chemical Biology & Drug Design*, 80(4):631–637, 2012.

[69] Garrett M. Morris, David S. Goodsell, Robert S. Halliday, Ruth Huey, William E. Hart, Richard K. Belew, and Arthur J. Olson. Automated docking using a Lamarckian genetic algorithm and an empirical binding free energy function. *Journal of Computational Chemistry*, 19 (14):1639–1662, 1998.

[70] Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. Springer, 2006. ISBN 9780387303031.

[71] Csaba Hetényi, Uko Maran, Alfonso T. García-Sosa, and Mati Karelson. Structure-based calculation of drug efficiency indices. *Bioinformatics*, 23(20):2678–2685, 2007.

[72] Alfonso T. García-Sosa, Csaba Hetényi, and Uko Maran. Drug efficiency indices for improvement of molecular docking scoring functions. *Journal of Computational Chemistry*, 31(1):174–184, 2010.

[73] Emanuele Perola. An Analysis of the Binding Efficiencies of Drugs and Their Leads in Successful Drug Discovery Programs. *Journal of Medicinal Chemistry*, 53(7):2986–2997, 2010.

[74] Luis Menéndez-Arias. Molecular basis of human immunodeficiency virus drug resistance: An update. *Antiviral Research*, 85(1):210–231, 2010.

[75] Patrick Marcellin, E. Jenny Heathcote, Maria Buti, Ed Gane, Robert A. de Man, Zahary Krastev, George Germanidis, Sam S. Lee, Robert Flisiak, Kelly Kaita, Michael Manns, Iskren Kotzev, Konstantin Tchernev, Peter Buggisch, Frank Weilert, Oya Ovung Kurdas, Mitchell L. Shiffman, Huy Trinh, Mary Kay Washington, Jeff Sorbel, Jane Anderson, Andrea Snow-Lampart, Elsa Mondou, Joe Quinn, and Franck Rousseau. Tenofovir Disoproxil Fumarate versus Adefovir Dipivoxil for Chronic Hepatitis B. *New England Journal of Medicine*, 359(23):2442–2455, 2008.

[76] Caroline M. Perry and Dene Simpson. Tenofovir Disoproxil Fumarate: In Chronic Hepatitis B. *Drugs*, 69(16):2245–2256, 2009.

[77] Francesc Vidal, Joan Carles Domingo, Jordi Guallar, Maria Saumoy, Begona Cordobilla, Rainel Sanchez de la Rosa, Marta Giralt, Maria Luisa Alvarez, Miguel Lopez-Dupla, Ferran Torres, Francesc Villarroya, Tomas Cihlar, and Pere Domingo. In Vitro Cytotoxicity and Mitochondrial Toxicity of Tenofovir Alone and in Combination with Other Antiretrovirals in Human Renal Proximal Tubule Cells. *Antimicrobial Agents and Chemotherapy*, 50(11):3824–3832, 2006.

[78] Pedro S. C. F. Rocha, Mazhar Sheikh, Rosalba Melchiorre, Mathilde Fagard, Stephanie Boutet, Rebecca Loach, Barbara Moffatt, Conrad Wagner, Herve Vaucheret, and Ian Furner. The Arabidopsis HOMOLOGY-DEPENDENT GENE SILENCING1 Gene Codes for an S-Adenosyl-L-Homocysteine Hydrolase Required for DNA Methylation-Dependent Gene Silencing. *The Plant Cell*, 17(2):404–417, 2005.

[79] Brian R. Lawson, Yulia Manenkova, Jasimuddin Ahamed, Xiaoru Chen, Jian-Ping Zou, Roberto Baccala, Argyrios N. Theofilopoulos, and Chong Yuan. Inhibition of Transmethylation Down-Regulates CD4 T Cell Activation and Curtails Development of Autoimmunity in a Model System. *The Journal of Immunology*, 178(8):5366–5374, 2007.

[80] İlker Durak, Recep Çetin, Erdinç Devrim, and İmge B. Ergüder. Effects of black grape extract on activities of dna turn-over enzymes in cancerous and non cancerous human colon tissues. *Life Sciences*, 76(25):2995–3000, 2005.

[81] Linda F. Thompson, Van De Wiele, Aletha B. Laurent, Scott W. Hooker, James G. Vaughn, Hong Jiang, Kamayani Khare, Rodney E. Kellems, Michael R. Blackburn, Michael S. Hershfield, and Regina Resta. Metabolites from apoptotic thymocytes inhibit thymopoiesis in adenosine deaminase-deficient fetal thymic organ cultures. *The Journal of Clinical Investigation*, 106(9):1149–1157, 2000.

[82] Michael J. Keiser, Bryan L. Roth, Blaine N. Armbruster, Paul Ernsberger, John J. Irwin, and Brian K. Shoichet. Relating protein pharmacology by ligand chemistry. *Nature Biotechnology*, 25(2):197–206, 2007.

[83] Christopher A. Lipinski, Franco Lombardo, Beryl W. Dominy, and Paul J. Feeney. Experimental and computational approaches to estimate solubility and permeability in drug discovery and development settings. *Advanced Drug Delivery Reviews*, 23(1-3):3–25, 1997.

[84] Ashutosh Kumar and Kam Y. J. Zhang. Investigation on the Effect of Key Water Molecules on Docking Performance in CSARdock Exercise. *Journal of Chemical Information and Modeling*, 53(8):1880–1892, 2013.

[85] Mette A. Lie, René Thomsen, Christian N. S. Pedersen, Birgit Schiøtt, and Mikael H. Christensen. Molecular Docking with Ligand Attached Water Molecules. *Journal of Chemical Information and Modeling*, 51(4):909–917, 2011.

[86] Hitesh Patel, Björn A. Grüning, Stefan Günther, and Irmgard Merfort. PyWATER: a PyMOL plug-in to find conserved water molecules in proteins by clustering. *Bioinformatics*, 30(20):2978–2980, 2014.

[87] Daniel Alvarez-Garcia and Xavier Barril. Relationship between Protein Flexibility and Binding: Lessons for Structure-Based Drug Design. *Journal of Chemical Theory and Computation*, 10(6):2608–2614, 2014.

[88] Montiago X. LaBute, Xiaohua Zhang, Jason Lenderman, Brian J. Bennion, Sergio E. Wong, and Felice C. Lightstone. Adverse Drug Reaction Prediction Using Scores Produced by Large-Scale Drug-Protein Target Docking on High-Performance Computing Machines. *PLoS ONE*, 9 (9):e106298, 2014.

[89] Honglin Li, Zhenting Gao, Ling Kang, Hailei Zhang, Kun Yang, Kunqian Yu, Xiaomin Luo, Weiliang Zhu, Kaixian

Chen, Jianhua Shen, Xicheng Wang, and Hualiang Jiang. TarFisDock: a web server for identifying drug targets with docking approach. *Nucleic Acids Research*, 34(suppl 2): W219–W224, 2006.

[90] Zhenting Gao, Honglin Li, Hailei Zhang, Xiaofeng Liu, Ling Kang, Xiaomin Luo, Weiliang Zhu, Kaixian Chen, Xicheng Wang, and Hualiang Jiang. PDTD: a web-accessible protein database for drug target identification. *BMC Bioinformatics*, 9(1):104, 2008.

[91] Fengshun Lu, Junqiang Song, Fukang Yin, and Xiaoqian Zhu. Performance evaluation of hybrid programming patterns for large CPU/GPU heterogeneous clusters. *Computer Physics Communications*, 183(6):1172–1181, 2012.

[92] B. Sukhwani and M. C. Herbordt. Fast binding site mapping using GPUs and CUDA. In *Proceedings of the IEEE International Symposium on Parallel & Distributed Processing, Workshops and PhD Forum (IPDPSW)*, pages 1–8, 2010.

[93] Yongchao Liu, Bertil Schmidt, and Douglas Maskell. CUDASW++2.0: enhanced Smith-Waterman protein database search on CUDA-enabled GPUs based on SIMT and virtualized SIMD abstractions. *BMC Research Notes*, 3(1): 93, 2010.

[94] Quan Liao, Jibo Wang, and Ian A. Watson. Accelerating Two Algorithms for Large-Scale Compound Selection on GPUs. *Journal of Chemical Information and Modeling*, 51 (5):1017–1024, 2011.

[95] Bharat Sukhwani and Martin C. Herbordt. GPU acceleration of a production molecular docking code. In *Proceed-*

ings of the 2nd Workshop on General Purpose Processing on Graphics Processing Units, pages 19–27, 2009.

[96] Lennart Heinzerling, Robert Klein, and Matthias Rarey. Fast force field-based optimization of protein-ligand complexes with graphics processor. *Journal of Computational Chemistry*, 33(32):2554–2565, 2012.

[97] Bo Hong, Jiadong Wu, and Jun tao Guo. Improving Prediction Accuracy of Protein-DNA Docking with GPU Computing. In *Proceedings of the 2011 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 482–487, 2011.

[98] Jiadong Wu, Bo Hong, Takako Takeda, and Jun tao Guo. High performance transcription factor-DNA docking with GPU computing. *Proteome Science*, 10:S17, 2012.

[99] David W. Ritchie and Vishwesh Venkatraman. Ultra-fast FFT protein docking on graphics processors. *Bioinformatics*, 26(19):2398–2405, 2010.

[100] Zhi wei Feng, Xu hong Tian, and Shan Chang. A Parallel Molecular Docking Approach Based on Graphic Processing Unit. In *Proceedings of the 4th International Conference on Bioinformatics and Biomedical Engineering (iCBBE)*, pages 1–4, 2010.

[101] Masahito Ohue, Takehiro Shimoda, Shuji Suzuki, Yuri Matsuzaki, Takashi Ishida, and Yutaka Akiyama. MEGADOCK 4.0: an ultra–high-performance protein–protein docking software for heterogeneous supercomputers. *Bioinformatics*, 30(22):3281–3283, 2014.

[102] Imran S. Haque, Vijay S. Pande, and W. P. Walters. SIML: A Fast SIMD Algorithm for Calculating LINGO

Chemical Similarities on GPUs and CPUs. *Journal of Chemical Information and Modeling*, 50(4):560–564, 2010.

[103] Chao Ma, Lirong Wang, and Xiang-Qun Xie. GPU Accelerated Chemical Similarity Calculation for Compound Library Comparison. *Journal of Chemical Information and Modeling*, 51(7):1521–1527, 2011.

[104] Marco Maggioni, Marco Domenico Santambrogio, and Jie Liang. GPU-accelerated Chemical Similarity Assessment for Large Scale Databases. *Procedia Computer Science*, 4 (0):2007–2016, 2011.

[105] Mark S. Friedrichs, Peter Eastman, Vishal Vaidyanathan, Mike Houston, Scott Legrand, Adam L. Beberg, Daniel L. Ensign, Christopher M. Bruns, and Vijay S. Pande. Accelerating molecular dynamic simulation on graphics processing units. *Journal of Computational Chemistry*, 30(6): 864–872, 2009.

[106] Weiguo Liu, Bertil Schmidt, Gerrit Voss, and Wolfgang Müller-Wittig. Accelerating molecular dynamics simulations using Graphics Processing Units with CUDA. *Computer Physics Communications*, 179(9):634–641, 2008.

[107] Scott Le Grand, Andreas W. Götz, and Ross C. Walker. SPFP: Speed without compromise—A mixed precision model for GPU accelerated molecular dynamics simulations. *Computer Physics Communications*, 184(2):374–380, 2013.

[108] Christoph Niethammer, Stefan Becker, Martin Bernreuther, Martin Buchholz, Wolfgang Eckhardt, Alexander Heinecke, Stephan Werth, Hans-Joachim Bungartz, Colin W. Glass, Hans Hasse, Jadran Vrabec, and Martin Horsch. ls1 mardyn: The Massively Parallel Molecular

Dynamics Code for Large Systems. *Journal of Chemical Theory and Computation*, 2014.

[109] Imran S. Haque and Vijay S. Pande. PAPER-Accelerating parallel evaluations of ROCS. *Journal of Computational Chemistry*, 31(1):117–132, 2010.

[110] Xin Yan, Jiabo Li, Qiong Gu, and Jun Xu. gWEGA: GPU-accelerated WEGA for molecular superposition and shape comparison. *Journal of Computational Chemistry*, 35(15):1122–1130, 2014.

[111] Yutong Zhao, Fu Kit Sheong, Jian Sun, Pedro Sander, and Xuhui Huang. A fast parallel clustering algorithm for molecular simulation trajectories. *Journal of Computational Chemistry*, 34(2):95–104, 2013.

[112] Kai J. Kohlhoff, Marc H. Sosnick, William T. Hsu, Vijay S. Pande, and Russ B. Altman. CAMPAIGN: an open-source library of GPU-accelerated data clustering algorithms. *Bioinformatics*, 27(16):2322–2323, 2011.

[113] Matthieu Chavent, Bruno Lévy, Michael Krone, Katrin Bidmon, Jean-Philippe Nominé, Thomas Ertl, and Marc Baaden. GPU-powered tools boost molecular visualization. *Briefings in Bioinformatics*, 12(6):689–701, 2011.

[114] Shuang Gao and Gregory D. Peterson. GASPRNG: GPU accelerated scalable parallel random number generator library. *Computer Physics Communications*, 184(4):1241–1249, 2013.

[115] Sheng-You Huang and Xiaoqin Zou. Ensemble docking of multiple protein structures: Considering protein structural variations in molecular docking. *Proteins: Structure, Function, and Bioinformatics*, 66(2):399–421, 2007.

[116] Ian R. Craig, Jonathan W. Essex, and Katrin Spiegel. Ensemble Docking into Multiple Crystallographically Derived Protein Structures: An Evaluation Based on the Statistical Analysis of Enrichments. *Journal of Chemical Information and Modeling*, 50(4):511–524, 2010.

[117] Oliver Korb, Tjelvar S. G. Olsson, Simon J. Bowden, Richard J. Hall, Marcel L. Verdonk, John W. Liebeschuetz, and Jason C. Cole. Potential and Limitations of Ensemble Docking. *Journal of Chemical Information and Modeling*, 52(5):1262–1274, 2012.

[118] Huameng Li, Aiguo Liu, Zhenjiang Zhao, Yufang Xu, Jiayuh Lin, David Jou, and Chenglong Li. Fragment-Based Drug Design and Drug Repositioning Using Multiple Ligand Simultaneous Docking (MLSD): Identifying Celecoxib and Template Compounds as Novel Inhibitors of Signal Transducer and Activator of Transcription 3 (STAT3). *Journal of Medicinal Chemistry*, 54(15):5592–5596, 2011.

[119] Laurent Hoffer and Dragos Horvath. S4MPLE – Sampler For Multiple Protein–Ligand Entities: Simultaneous Docking of Several Entities. *Journal of Chemical Information and Modeling*, 53(1):88–102, 2013.

[120] Christin Schärfer, Tanja Schulz-Gasch, Hans-Christian Ehrlich, Wolfgang Guba, Matthias Rarey, and Martin Stahl. Torsion Angle Preferences in Druglike Chemical Space: A Comprehensive Guide. *Journal of Medicinal Chemistry*, 56(5):2016–2028, 2013.

[121] Robin Taylor, Jason Cole, Oliver Korb, and Patrick McCabe. Knowledge-Based Libraries for Predicting the Geometric Preferences of Druglike Molecules. *Journal of Chemical Information and Modeling*, 54(9):2500–2514, 2014.

[122] Javier Klett, Álvaro Cortés-Cabrera, Rub Gil-Redondo, Federico Gago, and Antonio Morreale. ALFA: Automatic Ligand Flexibility Assignment. *Journal of Chemical Information and Modeling*, 54(1):314–323, 2014.

[123] Julie R. Schames, Richard H. Henchman, Jay S. Siegel, Christoph A. Sotriffer, Haihong Ni, and J. A. McCammon. Discovery of a Novel Binding Trench in HIV Integrase. *Journal of Medicinal Chemistry*, 47(8):1879–1881, 2004.

[124] Prashant Khodade, R. Prabhu, Nagasuma Chandra, Soumyendu Raha, and R. Govindarajan. Parallel implementation of AutoDock. *Journal of Applied Crystallography*, 40(3):598–599, 2007.

[125] Nikita D. Prakhov, Alexander L. Chernorudskiy, and Murat R. Gainullin. VSDocker: a tool for parallel high-throughput virtual screening using AutoDock on Windows-based computer clusters. *Bioinformatics*, 26 (10):1374–1375, 2010.

[126] Andrew Norgan, Paul Coffman, Jean-Pierre Kocher, David Katzmann, and Carlos Sosa. Multilevel Parallelization of AutoDock 4.2. *Journal of Cheminformatics*, 3(1):12, 2011.

[127] John J. Irwin, Brian K. Shoichet, Michael M. Mysinger, Niu Huang, Francesco Colizzi, Pascal Wassam, and Yiqun Cao. Automated Docking Screens: A Feasibility Study. *Journal of Medicinal Chemistry*, 52(18):5712–5720, 2009.

[128] Ryan G. Coleman and Kim A. Sharp. Protein Pockets: Inventory, Shape, and Comparison. *Journal of Chemical Information and Modeling*, 50(4):589–603, 2010.

[129] Tsung-Ying Tsai, Kai-Wei Chang, and Calvin Chen. iScreen: world's first cloud-computing web server for virtual screening and de novo drug design based on TCM

database@Taiwan. *Journal of Computer-Aided Molecular Design*, 25(6):525–531, 2011.

[130] Dominique Douguet, H. Munier-Lehmann, Gilles Labesse, and Sylvie Pochet. LEA3D: A Computer-Aided Ligand Design for Structure-Based Drug Design. *Journal of Medicinal Chemistry*, 48(7):2457–2468, 2005.

[131] Aurélien Grosdidier, Vincent Zoete, and Olivier Michielin. SwissDock, a protein-small molecule docking web service based on EADock DSS. *Nucleic Acids Research*, 39(suppl 2):W270–W277, 2011.

[132] Aurélien Grosdidier, Vincent Zoete, and Olivier Michielin. EADock: Docking of small molecules into protein active sites with a multiobjective evolutionary optimization. *Proteins: Structure, Function, and Bioinformatics*, 67(4): 1010–1025, 2007.

[133] Aurélien Grosdidier, Vincent Zoete, and Olivier Michielin. Fast docking using the CHARMM force field with EADock DSS. *Journal of Computational Chemistry*, 32(10):2149–2159, 2011.

[134] Eric Therrien, Pablo Englebienne, Andrew G. Arrowsmith, Rodrigo Mendoza-Sanchez, Christopher R. Corbeil, Nathanael Weill, Val Campagna-Slater, and Nicolas Moitessier. Integrating Medicinal Chemistry, Organic/Combinatorial Chemistry, and Computational Chemistry for the Discovery of Selective Estrogen Receptor Modulators with Forecaster, a Novel Platform for Drug Discovery. *Journal of Chemical Information and Modeling*, 52(1):210–224, 2012.

[135] Hongjian Li, Bing Ni, Man-Hon Wong, and Kwong-Sak Leung. A fast CUDA implementation of agrep algorithm

for approximate nucleotide sequence matching. In *Proceedings of the 9th IEEE Symposium on Application Specific Processors (SASP)*, pages 74–77, 2011.

[136] Renxiao Wang, Xueliang Fang, Yipin Lu, and Shaomeng Wang. The PDBbind Database: Collection of Binding Affinities for Protein-Ligand Complexes with Known Three-Dimensional Structures. *Journal of Medicinal Chemistry*, 47(12):2977–2980, 2004.

[137] Renxiao Wang, Xueliang Fang, Yipin Lu, Chao-Yie Yang, and Shaomeng Wang. The PDBbind Database: Methodologies and Updates. *Journal of Medicinal Chemistry*, 48 (12):4111–4119, 2005.

[138] Zhihai Liu, Yan Li, Li Han, Jie Li, Jie Liu, Zhixiong Zhao, Wei Nie, Yuchen Liu, and Renxiao Wang. PDB-wide collection of binding data: current status of the PDBbind database. *Bioinformatics*, 31(3):405–412, 2015.

[139] Leo Breiman. Random Forests. *Machine Learning*, 45(1): 5–32, 2001.

[140] Pedro J. Ballester, Martina Mangold, Nigel I. Howard, Richard L. Marchese Robinson, Chris Abell, Jochen Blumberger, and John B. O. Mitchell. Hierarchical virtual screening for the discovery of new molecular scaffolds in antibacterial hit identification. *Journal of The Royal Society Interface*, 9(77):3196–3207, 2012.

[141] Leo Breiman. *Classification and regression trees*. Chapman & Hall, 1984. ISBN 9780412048418.

[142] James B. Dunbar, Richard D. Smith, Chao-Yie Yang, Peter Man-Un Ung, Katrina W. Lexa, Nickolay A. Khazanov, Jeanne A. Stuckey, Shaomeng Wang, and

Heather A. Carlson. CSAR Benchmark Exercise of 2010: Selection of the Protein-Ligand Complexes. *Journal of Chemical Information and Modeling*, 51(9):2036–2046, 2011.

[143] James B. Dunbar, Richard D. Smith, Chao-Yie Yang, Peter Man-Un Ung, Katrina W. Lexa, Nickolay A. Khazanov, Jeanne A. Stuckey, Shaomeng Wang, and Heather A. Carlson. Correction to CSAR Benchmark Exercise of 2010: Selection of the Protein-Ligand Complexes. *Journal of Chemical Information and Modeling*, 51 (9):2146–2146, 2011.

[144] Helen M. Berman, John Westbrook, Zukang Feng, Gary Gilliland, T. N. Bhat, Helge Weissig, Ilya N. Shindyalov, and Philip E. Bourne. The Protein Data Bank. *Nucleic Acids Research*, 28(1):235–242, 2000.

[145] Liegi Hu, Mark L. Benson, Richard D. Smith, Michael G. Lerner, and Heather A. Carlson. Binding MOAD (Mother Of All Databases). *Proteins: Structure, Function, and Bioinformatics*, 60(3):333–340, 2005.

[146] Mark L. Benson, Richard D. Smith, Nickolay A. Khazanov, Brandon Dimcheff, John Beaver, Peter Dresslar, Jason Nerothin, and Heather A. Carlson. Binding MOAD, a high-quality protein-ligand database. *Nucleic Acids Research*, 36(suppl_1):D674–D678, 2008.

[147] Aqeel Ahmed, Richard D. Smith, Jordan J. Clark, James B. Dunbar, and Heather A. Carlson. Recent improvements to Binding MOAD: a resource for protein–ligand binding affinities and structures. *Nucleic Acids Research*, 43(D1):D465–D469, 2015.

[148] Tiejun Cheng, Xun Li, Yan Li, Zhihai Liu, and Renxiao Wang. Comparative Assessment of Scoring Functions on a Diverse Test Set. *Journal of Chemical Information and Modeling*, 49(4):1079–1093, 2009.

[149] Pedro J. Ballester. Machine Learning Scoring Functions Based on Random Forest and Support Vector Regression. In *Pattern Recognition in Bioinformatics*, volume 7632 of *Lecture Notes in Computer Science*, pages 14–25. Springer Berlin Heidelberg, 2012.

[150] Renxiao Wang, Luhua Lai, and Shaomeng Wang. Further development and validation of empirical scoring functions for structure-based binding affinity prediction. *Journal of Computer-Aided Molecular Design*, 16(1):11–26, 2002.

[151] Sebastian J. Schultheiss. Ten Simple Rules for Providing a Scientific Web Resource. *PLoS Computational Biology*, 7(5):e1001126, 2011.

[152] Philip E. Bourne. Ten Simple Rules for Getting Ahead as a Computational Biologist in Academia. *PLoS Computational Biology*, 7(1):e1002001, 2011.

[153] Sean Wilkinson and Jonas Almeida. QMachine: commodity supercomputing in web browsers. *BMC Bioinformatics*, 15(1):176, 2014.

[154] Jacob D. Durrant and J. Andrew McCammon. BINANA: A novel algorithm for ligand-binding characterization. *Journal of Molecular Graphics and Modelling*, 29(6):888–893, 2011.

[155] Ching-Man Tse, Hongjian Li, Kwong-Sak Leung, Kin-Hong Lee, and Man-Hon Wong. Interactive Drug Design in Virtual Reality. In *Proceedings of the 15th International*

Conference on Information Visualisation (IV), pages 226–231, 2011.

[156] Kota Kasahara and Kengzo Kinoshita. GIANT: pattern analysis of molecular interactions in 3D structures of protein-small ligand complexes. *BMC Bioinformatics*, 15(1):12, 2014.

[157] Robert M. Hanson, Jaime Prilusky, Zhou Renjian, Takanori Nakane, and Joel L. Sussman. JSmol and the Next-Generation Web-Based Representation of 3D Molecular Structure as Applied to Proteopedia. *Israel Journal of Chemistry*, 53(3-4):207–216, 2013.

[158] Dong Xu and Yang Zhang. Generating Triangulated Macromolecular Surfaces by Euclidean Distance Transform. *PLoS ONE*, 4(12):e8140, 2009.

[159] Dong Xu, Hua Li, and Yang Zhang. Fast and Accurate Calculation of Protein Depth by Euclidean Distance Transform. In *Proceedings of the 17th International Conference on Research in Computational Molecular Biology*, pages 304–316, 2013.

[160] Marco Callieri, Raluca Mihaela Andrei, Marco Di Benedetto, Monica Zoppè, and Roberto Scopigno. Visualization methods for molecular studies on the web platform. In *Proceedings of the 15th International Conference on Web 3D Technology*, pages 117–126, 2010.

[161] Marco Di Benedetto, Federico Ponchio, Fabio Ganovelli, and Roberto Scopigno. SpiderGL: a JavaScript 3D graphics library for next-generation WWW. In *Proceedings of the 15th International Conference on Web 3D Technology*, pages 165–174, 2010.

[162] Peter W. Rose, Bojan Beran, Chunxiao Bi, Wolfgang F. Bluhm, Dimitris Dimitropoulos, David S. Goodsell, Andreas Prlic, Martha Quesada, Gregory B. Quinn, John D. Westbrook, Jasmine Young, Benjamin Yukich, Christine Zardecki, Helen M. Berman, and Philip E. Bourne. The RCSB Protein Data Bank: redesigned web site and web services. *Nucleic Acids Research*, 39(suppl 1):D392–D401, 2011.

[163] Qiuxiang Tan, Ya Zhu, Jian Li, Zhuxi Chen, Gye Won Han, Irina Kufareva, Tingting Li, Limin Ma, Gustavo Fenalti, Jing Li, Wenru Zhang, Xin Xie, Huaiyu Yang, Hualiang Jiang, Vadim Cherezov, Hong Liu, Raymond C. Stevens, Qiang Zhao, and Beili Wu. Structure of the CCR5 Chemokine Receptor–HIV Entry Inhibitor Maraviroc Complex. *Science*, 341(6152):1387–1390, 2013.

[164] P. Kirkpatrick and C. Ellis. Chemical space. *Nature*, 432 (7019):823, 2004.

[165] Brett R. Beno and David R. Langley. MORPH: A New Tool for Ligand Design. *Journal of Chemical Information and Modeling*, 50(6):1159–1164, 2010.

[166] Patrick Pfeffer, Thomas Fober, Eyke Hüllermeier, and Gerhard Klebe. GARLig: A Fully Automated Tool for Subset Selection of Large Fragment Spaces via a Self-Adaptive Genetic Algorithm. *Journal of Chemical Information and Modeling*, 50(9):1644–1659, 2010.

[167] Yaxia Yuan, Jianfeng Pei, and Luhua Lai. LigBuilder 2: A Practical de Novo Drug Design Approach. *Journal of Chemical Information and Modeling*, 51(5):1083–1091, 2011.

[168] Yan Li, Yuan Zhao, Zhihai Liu, and Renxiao Wang. Automatic Tailoring and Transplanting: A Practical Method that Makes Virtual Screening More Useful. *Journal of Chemical Information and Modeling*, 51(6):1474–1491, 2011.

[169] Andrea R. Beccari, Carlo Cavazzoni, Claudia Beato, and Gabriele Costantino. LiGen: A High Performance Workflow for Chemistry Driven de Novo Design. *Journal of Chemical Information and Modeling*, 53(6):1518–1527, 2013.

[170] Jacob D. Durrant, Rommie E. Amaro, and J. Andrew McCammon. AutoGrow: A Novel Algorithm for Protein Inhibitor Design. *Chemical Biology & Drug Design*, 73(2): 168–178, 2009.

[171] Jacob D. Durrant, Steffen Lindert, and J. Andrew McCammon. AutoGrow 3.0: An improved algorithm for chemically tractable, semi-automated protein inhibitor design. *Journal of Molecular Graphics and Modelling*, 44(0): 104–112, 2013.

[172] Jacob D. Durrant and J. A. McCammon. AutoClickChem: Click Chemistry in Silico. *PLoS Computational Biology*, 8 (3):e1002397, 2012.

[173] Jacob D. Durrant, Aaron J. Friedman, and J. A. McCammon. CrystalDock: A Novel Approach to Fragment-Based Drug Design. *Journal of Chemical Information and Modeling*, 51(10):2573–2580, 2011.

[174] Steffen Lindert, Jacob D. Durrant, and J. Andrew McCammon. LigMerge: A Fast Algorithm to Generate Models of Novel Potential Ligands from Sets of Known Binders. *Chemical Biology & Drug Design*, 80(3):358–365, 2012.

[175] Marco Foscato, Giovanni Occhipinti, Vishwesh Venkatraman, Bjørn K. Alsberg, and Vidar R. Jensen. Automated Design of Realistic Organometallic Molecules from Fragments. *Journal of Chemical Information and Modeling*, 54 (3):767–780, 2014.

[176] Marco Foscato, Vishwesh Venkatraman, Giovanni Occhipinti, Bjørn K. Alsberg, and Vidar R. Jensen. Automated Building of Organometallic Complexes from 3D Fragments. *Journal of Chemical Information and Modeling*, 54(7):1919–1931, 2014.

[177] Erchang Shang, Yaxia Yuan, Xinyi Chen, Ying Liu, Jianfeng Pei, and Luhua Lai. De Novo Design of Multitarget Ligands with an Iterative Fragment-Growing Strategy. *Journal of Chemical Information and Modeling*, 54 (4):1235–1241, 2014.

[178] Kentaro Kawai, Naoya Nagata, and Yoshimasa Takahashi. De Novo Design of Drug-Like Molecules by a Fragment-Based Molecular Evolutionary Approach. *Journal of Chemical Information and Modeling*, 54(1):49–56, 2014.

[179] Chunquan Sheng and Wannian Zhang. Fragment Informatics and Computational Fragment-Based Drug Design: An Overview and Update. *Medicinal Research Reviews*, 33 (3):554–598, 2013.

[180] Gisbert Schneider. Future De Novo Drug Design. *Molecular Informatics*, 33(6-7):397–402, 2014.

[181] Emilie Pihan, Lionel Colliandre, Jean-François Guichou, and Dominique Douguet. e-Drug3D: 3D structure collections dedicated to drug repurposing and fragment-based drug design. *Bioinformatics*, 28(11):1540–1541, 2012.

[182] Gisbert Schneider and Uli Fechner. Computer-based de novo design of drug-like molecules. *Nature Reviews Drug Discovery*, 4(8):649–663, 2005.

[183] Bernard Pirard. The quest for novel chemical matter and the contribution of computer-aided de novo design. *Expert Opinion on Drug Discovery*, 6(3):225–231, 2011.

[184] Zenon D. Konteatis. In silico fragment-based drug design. *Expert Opinion on Drug Discovery*, 5(11):1047–1065, 2010.

[185] Christopher W. Murray and Tom L. Blundell. Structural biology in fragment-based drug design. *Current Opinion in Structural Biology*, 20(4):497–507, 2010.

[186] Gisbert Schneider. Designing the molecular future. *Journal of Computer-Aided Molecular Design*, 26(1):115–120, 2012.

[187] Noel O'Boyle, Michael Banck, Craig James, Chris Morley, Tim Vandermeersch, and Geoffrey Hutchison. Open Babel: An open chemical toolbox. *Journal of Cheminformatics*, 3(1):33, 2011.

[188] Carlo Melchiorre, Maria Laura Bolognesi, Anna Minarini, Michela Rosini, and Vincenzo Tumiatti. Polyamines in Drug Discovery: From the Universal Template Approach to the Multitarget-Directed Ligand Design Strategy. *Journal of Medicinal Chemistry*, 53(16):5906–5914, 2010.

[189] Tiejun Cheng, Qingliang Li, Zhigang Zhou, Yanli Wang, and StephenH Bryant. Structure-Based Virtual Screening for Drug Discovery: a Problem-Centric Review. *The AAPS Journal*, 14(1):133–141, 2012.

[190] Dik-Lung Ma, Daniel Shiu-Hin Chan, and Chung-Hang Leung. Drug repositioning by structure-based virtual

screening. *Chemical Society Reviews*, 42(5):2130–2141, 2013.

[191] William L. Jorgensen. Efficient Drug Lead Discovery and Optimization. *Accounts of Chemical Research*, 42(6):724–733, 2009.

[192] Andrea Volkamer, Daniel Kuhn, Friedrich Rippmann, and Matthias Rarey. DoGSiteScorer: a web server for automatic binding site prediction, analysis and druggability assessment. *Bioinformatics*, 28(15):2074–2075, 2012.

[193] Johannes C. Hermann, Ricardo Marti-Arbona, Alexander A. Fedorov, Elena Fedorov, Steven C. Almo, Brian K. Shoichet, and Frank M. Raushel. Structure-based activity prediction for an enzyme of unknown function. *Nature*, 448(7155):775–779, 2007.

[194] Bernhard Baum, Laveena Muley, Michael Smolinski, Andreas Heine, David Hangauer, and Gerhard Klebe. Non-additivity of Functional Group Contributions in Protein–Ligand Binding: A Comprehensive Study by Crystallography and Isothermal Titration Calorimetry. *Journal of Molecular Biology*, 397(4):1042–1054, 2010.

[195] Liwei Li, Bo Wang, and Samy O. Meroueh. Support Vector Regression Scoring of Receptor-Ligand Complexes for Rank-Ordering and Virtual Screening of Chemical Libraries. *Journal of Chemical Information and Modeling*, 51(9):2132–2138, 2011.

[196] Xuchang Ouyang, Stephanus Daniel Handoko, and Chee Keong Kwoh. CScore: a simple yet effective scoring function for protein-ligand binding affinity prediction using modified CMAC learning architecture. *Journal of Bioinformatics and Computational Biology*, 09:1–14, 2011.

[197] Qian Liu, Chee Keong Kwoh, and Jinyan Li. Binding Affinity Prediction for Protein–Ligand Complexes Based on β Contacts and B Factor. *Journal of Chemical Information and Modeling*, 53(11):3076–3085, 2013.

[198] Christian Kramer and Peter Gedeck. Leave-Cluster-Out Cross-Validation Is Appropriate for Scoring Functions Derived from Diverse Protein Data Sets. *Journal of Chemical Information and Modeling*, 50(11):1961–1969, 2010.

[199] Christian Kramer and Peter Gedeck. Global Free Energy Scoring Functions Based on Distance-Dependent Atom-Type Pair Descriptors. *Journal of Chemical Information and Modeling*, 51(3):707–720, 2011.

[200] Gregory A. Ross, Garrett M. Morris, and Philip C. Biggin. One Size Does Not Fit All: The Limits of Structure-Based Models in Drug Discovery. *Journal of Chemical Theory and Computation*, 9(9):4266–4274, 2013.

[201] Pedro J. Ballester and John B. O. Mitchell. Comments on "Leave-Cluster-Out Cross-Validation Is Appropriate for Scoring Functions Derived from Diverse Protein Data Sets": Significance for the Validation of Scoring Functions. *Journal of Chemical Information and Modeling*, 51 (8):1739–1741, 2011.

[202] Yat T. Tang and Garland R. Marshall. PHOENIX: A Scoring Function for Affinity Prediction Derived Using High-Resolution Crystal Structures and Calorimetry Measurements. *Journal of Chemical Information and Modeling*, 51 (2):214–228, 2011.

[203] Gisbert Schneider. Virtual screening: an endless staircase? *Nature Reviews Drug Discovery*, 9(4):273–276, 2010.

[204] Yusuf Tanrikulu, Björn Krüger, and Ewgenij Proschak. The holistic integration of virtual screening in drug discovery. *Drug Discovery Today*, 18(7–8):358–364, 2013.

[205] Didier Rognan. Structure-Based Approaches to Target Fishing and Ligand Profiling. *Molecular Informatics*, 29 (3):176–187, 2010.

[206] Lei Xie, Li Xie, Sarah L. Kinnings, and Philip E. Bourne. Novel Computational Approaches to Polypharmacology as a Means to Define Responses to Individual Drugs. *Annual Review of Pharmacology and Toxicology*, 52(1):361–379, 2012.

[207] Stéphanie Pérot, Olivier Sperandio, Maria A. Miteva, Anne-Claude Camproux, and Bruno O. Villoutreix. Druggable pockets and binding site centric chemical space: a paradigm shift in drug discovery. *Drug Discovery Today*, 15(15-16):656–667, 2010.

[208] Jennifer L. Lahti, Grace W. Tang, Emidio Capriotti, Tianyun Liu, and Russ B. Altman. Bioinformatics and variability in drug response: a protein structural perspective. *Journal of The Royal Society Interface*, 9(72):1409–1437, 2012.

[209] Jean-Yves Trosset and Nicolas Vodovar. Structure-Based Target Druggability Assessment. In *Target Identification and Validation in Drug Discovery*, volume 986 of *Methods in Molecular Biology*, pages 141–164. Humana Press, 2013.

[210] Elaine C. Meng, Brian K. Shoichet, and Irwin D. Kuntz. Automated docking with grid-based energy evaluation. *Journal of Computational Chemistry*, 13(4):505–524, 1992.

[211] Niu Huang, Chakrapani Kalyanaraman, Katarzyna Bernacki, and Matthew P. Jacobson. Molecular mechan-

ics methods for predicting protein-ligand binding. *Physical Chemistry Chemical Physics*, 8(44):5166–5177, 2006.

[212] Fedor N. Novikov, Alexey A. Zeifman, Oleg V. Stroganov, Viktor S. Stroylov, Val Kulkov, and Ghermes G. Chilov. CSAR Scoring Challenge Reveals the Need for New Concepts in Estimating Protein-Ligand Binding Affinity. *Journal of Chemical Information and Modeling*, 51(9):2090–2096, 2011.

[213] Hans-Joachim Böhm. The development of a simple empirical scoring function to estimate the binding constant for a protein-ligand complex of known three-dimensional structure. *Journal of Computer-Aided Molecular Design*, 8(3):243–256, 1994.

[214] Matthew D. Eldridge, Christopher W. Murray, Timothy R. Auton, Gaia V. Paolini, and Roger P. Mee. Empirical scoring functions: I. The development of a fast empirical scoring function to estimate the binding affinity of ligands in receptor complexes. *Journal of Computer-Aided Molecular Design*, 11(5):425–445, 1997.

[215] Richard A. Friesner, Jay L. Banks, Robert B. Murphy, Thomas A. Halgren, Jasna J. Klicic, Daniel T. Mainz, Matthew P. Repasky, Eric H. Knoll, Mee Shelley, Jason K. Perry, David E. Shaw, Perry Francis, and Peter S. Shenkin. Glide: A New Approach for Rapid, Accurate Docking and Scoring. 1. Method and Assessment of Docking Accuracy. *Journal of Medicinal Chemistry*, 47 (7):1739–1749, 2004.

[216] André Krammer, Paul D. Kirchhoff, X. Jiang, C. M. Venkatachalam, and Marvin Waldman. LigScore: a novel scoring function for predicting binding affinities. *Journal of Molecular Graphics and Modelling*, 23(5):395–407, 2005.

[217] Daniel K. Gehlhaar, Gennady M. Verkhivker, Paul A. Rejto, Christopher J. Sherman, David R. Fogel, Lawrence J. Fogel, and Stephan T. Freer. Molecular recognition of the inhibitor AG-1343 by HIV-1 protease: conformationally flexible docking by evolutionary programming. *Chemistry & Biology*, 2(5):317–324, 1995.

[218] Ingo Muegge and Yvonne C. Martin. A General and Fast Scoring Function for Protein-Ligand Interactions: A Simplified Potential Approach. *Journal of Medicinal Chemistry*, 42(5):791–804, 1999.

[219] Wijnand T. M. Mooij and Marcel L. Verdonk. General and targeted statistical potentials for protein-ligand interactions. *Proteins: Structure, Function, and Bioinformatics*, 61(2):272–287, 2005.

[220] Holger Gohlke, Manfred Hendlich, and Gerhard Klebe. Knowledge-based scoring function to predict protein-ligand interactions. *Journal of Molecular Biology*, 295(2): 337–356, 2000.

[221] Sheng-You Huang, Sam Z. Grinter, and Xiaoqin Zou. Scoring functions and their evaluation methods for protein-ligand docking: recent advances and future directions. *Physical Chemistry Chemical Physics*, 12(40):12899–12908, 2010.

[222] Ken A. Dill. Additivity Principles in Biochemistry. *Journal of Biological Chemistry*, 272(2):701–704, 1997.

[223] Elizabeth Yuriev and Paul A. Ramsland. Latest developments in molecular docking: 2010-2011 in review. *Journal of Molecular Recognition*, 26(5):215–239, 2013.

[224] Wenhu Zhan, Daqiang Li, Jinxin Che, Liangren Zhang, Bo Yang, Yongzhou Hu, Tao Liu, and Xiaowu Dong. Inte-

grating docking scores, interaction profiles and molecular descriptors to improve the accuracy of molecular docking: Toward the discovery of novel Akt1 inhibitors. *European Journal of Medicinal Chemistry*, 75(0):11–20, 2014.

[225] Jui-Chih Wang and Jung-Hsin Lin. Scoring Functions for Prediction of Protein-Ligand Interactions. *Current Pharmaceutical Design*, 19(12):2174–2182, 2013.

[226] Jui-Chih Wang, Jung-Hsin Lin, Chung-Ming Chen, Alex L. Perryman, and Arthur J. Olson. Robust Scoring Functions for Protein–Ligand Interactions with Quantum Chemical Charge Models. *Journal of Chemical Information and Modeling*, 51(10):2528–2537, 2011.

[227] Andrew R. Leach, Brian K. Shoichet, and Catherine E. Peishoff. Prediction of Protein−Ligand Interactions. Docking and Scoring: Successes and Gaps. *Journal of Medicinal Chemistry*, 49(20):5851–5855, 2006.

[228] Georgia B. McGaughey, Robert P. Sheridan, Christopher I. Bayly, J. C. Culberson, Constantine Kreatsoulas, Stacey Lindsley, Vladimir Maiorov, Jean-Francois Truchon, and Wendy D. Cornell. Comparison of Topological, Shape, and Docking Methods in Virtual Screening. *Journal of Chemical Information and Modeling*, 47(4):1504–1519, 2007.

[229] Akifumi Oda, Keiichi Tsuchida, Tadakazu Takakura, Noriyuki Yamaotsu, and Shuichi Hirono. Comparison of Consensus Scoring Strategies for Evaluating Computational Models of Protein-Ligand Complexes. *Journal of Chemical Information and Modeling*, 46(1):380–391, 2006.

[230] Dariusz Plewczynski, Michał Łaźniewski, Rafał Augustyniak, and Krzysztof Ginalski. Can we trust docking re-

sults? Evaluation of seven commonly used programs on PDBbind database. *Journal of Computational Chemistry*, 32(4):742–755, 2011.

[231] Ajay N. Jain. Scoring noncovalent protein-ligand interactions: A continuous differentiable function tuned to compute binding affinities. *Journal of Computer-Aided Molecular Design*, 10(5):427–440, 1996.

[232] Gareth Jones, Peter Willett, and Robert C. Glen. Molecular recognition of receptor sites using a genetic algorithm with a description of desolvation. *Journal of Molecular Biology*, 245(1):43–53, 1995.

[233] Hans F. G. Velec, Holger Gohlke, and Gerhard Klebe. DrugScoreCSD-Knowledge-Based Scoring Function Derived from Small Molecule Crystal Data with Superior Recognition Rate of Near-Native Ligand Poses and Better Affinity Prediction. *Journal of Medicinal Chemistry*, 48(20):6296–6303, 2005.

[234] Hao Fan, Dina Schneidman-Duhovny, John J. Irwin, Guangqiang Dong, Brian K. Shoichet, and Andrej Sali. Statistical Potential for Modeling and Ranking of Protein–Ligand Interactions. *Journal of Chemical Information and Modeling*, 51(12):3078–3092, 2011.

[235] Nadine Schneider, Gudrun Lange, Sally Hindle, Robert Klein, and Matthias Rarey. A consistent description of HYdrogen bond and DEhydration energies in protein–ligand complexes: methods behind the HYDE scoring function. *Journal of Computer-Aided Molecular Design*, 27(1):15–29, 2013.

[236] Sheng-Hung Wang, Ying-Ta Wu, Sheng-Chu Kuo, and John Yu. HotLig: A Molecular Surface-Directed Approach

to Scoring Protein–Ligand Interactions. *Journal of Chemical Information and Modeling*, 53(8):2181–2195, 2013.

[237] Gerd Neudert and Gerhard Klebe. DSX: A Knowledge-Based Scoring Function for the Assessment of Protein–Ligand Complexes. *Journal of Chemical Information and Modeling*, 51(10):2731–2745, 2011.

[238] Julien Michel and Jonathan Essex. Prediction of protein-ligand binding affinity by free energy simulations: assumptions, pitfalls and expectations. *Journal of Computer-Aided Molecular Design*, 24(8):639–658, 2010.

[239] Douglas E. V. Pires, Raquel C. de Melo-Minardi, Carlos H. da Silveira, Frederico F. Campos, and Wagner Meira. aCSM: noise-free graph-based signatures to large-scale receptor-based ligand prediction. *Bioinformatics*, 29 (7):855–861, 2013.

[240] Jacob D. Durrant, Aaron J. Friedman, Kathleen E. Rogers, and J. A. McCammon. Comparing Neural-Network Scoring Functions and the State of the Art: Applications to Common Library Screening. *Journal of Chemical Information and Modeling*, 53(7):1726–1735, 2013.

[241] Bo Ding, Jian Wang, Nan Li, and Wei Wang. Characterization of Small Molecule Binding. I. Accurate Identification of Strong Inhibitors in Virtual Screening. *Journal of Chemical Information and Modeling*, 53(1):114–122, 2013.

[242] Kathrin Heikamp and Jü Bajorath. Comparison of Confirmed Inactive and Randomly Selected Compounds as Negative Training Examples in Support Vector Machine-Based Virtual Screening. *Journal of Chemical Information and Modeling*, 53(7):1595–1601, 2013.

[243] Rafal Kurczab, Sabina Smusz, and Andrzej Bojarski. The influence of negative training set size on machine learning-based virtual screening. *Journal of Cheminformatics*, 6 (1):32, 2014.

[244] Yan Li, Zhihai Liu, Jie Li, Li Han, Jie Liu, Zhixiong Zhao, and Renxiao Wang. Comparative Assessment of Scoring Functions on an Updated Benchmark: 1. Compilation of the Test Set. *Journal of Chemical Information and Modeling*, 54(6):1700–1716, 2014.

[245] Yan Li, Li Han, Zhihai Liu, and Renxiao Wang. Comparative Assessment of Scoring Functions on an Updated Benchmark: 2. Evaluation Methods and General Results. *Journal of Chemical Information and Modeling*, 54(6): 1717–1736, 2014.

[246] A. Mauri, V. Consonni, M. Pavan, and R. Todeschini. DRAGON software: an easy approach to molecular descriptor calculations. *MATCH Communications in Mathematical and in Computer Chemistry*, 56:237–248, 2006.

[247] Chun Wei Yap. PaDEL-descriptor: An open source software to calculate molecular descriptors and fingerprints. *Journal of Computational Chemistry*, 32(7):1466–1474, 2011.

[248] César R. García-Jacas, Yovani Marrero-Ponce, Liesner Acevedo-Martínez, Stephen J. Barigye, José R. Valdés-Martiní, and Ernesto Contreras-Torres. QuBiLS-MIDAS: A parallel free-software for molecular descriptors computation based on multilinear algebraic maps. *Journal of Computational Chemistry*, 35(18):1395–1409, 2014.

[249] Steven R. Shave. *The Development of High Performance*

Structure and Ligand Based Virtual Screening Techniques. PhD thesis, University of Edinburgh, 2010.

[250] Lin Song, Peter Langfelder, and Steve Horvath. Random generalized linear model: a highly accurate and interpretable ensemble predictor. *BMC Bioinformatics*, 14 (1):5, 2013.

[251] Lin Song and Steve Horvath. Predicting COPD status with a random generalized linear model. *Systems Biomedicine*, 1(4):61–67, 2013.

[252] Denis Fourches, Regina Politi, and Alexander Tropsha. Target-Specific Native/Decoy Pose Classifier Improves the Accuracy of Ligand Ranking in the CSAR 2013 Benchmark. *Journal of Chemical Information and Modeling*, 55 (1):63–71, 2015.

[253] Reyhaneh Esmaielbeiki and Jean-Christophe Nebel. Scoring docking conformations using predicted protein interfaces. *BMC Bioinformatics*, 15(1):171, 2014.

[254] Pedro J. Ballester, Paul W. Finn, and W. Graham Richards. Ultrafast shape recognition: Evaluating a new ligand-based virtual screening technology. *Journal of Molecular Graphics and Modelling*, 27(7):836–845, 2009.

[255] Pedro J. Ballester, Isaac Westwood, Nicola Laurieri, Edith Sim, and W. Graham Richards. Prospective virtual screening with Ultrafast Shape Recognition: the identification of novel inhibitors of arylamine N-acetyltransferases. *Journal of The Royal Society Interface*, 7(43):335–342, 2010.

[256] Chian Ying Teo, Mohd Basyaruddin Abdul Rahman, Adam Leow Thean Chor, Abu Bakar Salleh, Pedro J.

Ballester, and Bimo A. Tejo. Ligand-Based Virtual Screening for the Discovery of Inhibitors for Protein Arginine Deiminase Type 4 (PAD4). *Metabolomics*, 3(1):118, 2013.

[257] Sachin P. Patil, Pedro J. Ballester, and Cassidy R. Kerezsi. Prospective virtual screening for novel p53–MDM2 inhibitors using ultrafast shape recognition. *Journal of Computer-Aided Molecular Design*, 28(2):89–97, 2014.

[258] Birgit Hoeger, Maren Diether, Pedro J. Ballester, and Maja Köhn. Biochemical evaluation of virtual screening methods reveals a cell-active inhibitor of the cancer-promoting phosphatases of regenerating liver. *European Journal of Medicinal Chemistry*, 88(0):89–100, 2014.

[259] Lirong Wang, Chao Ma, Peter Wipf, Haibin Liu, Weiwei Su, and Xiang-Qun Xie. TargetHunter: An In Silico Target Identification Tool for Predicting Therapeutic Potential of Small Organic Molecules Based on Chemogenomic Database. *The AAPS Journal*, 15(2):395–406, 2013.

[260] David Gfeller, Olivier Michielin, and Vincent Zoete. Shaping the interaction landscape of bioactive molecules. *Bioinformatics*, 29(23):3073–3079, 2013.

[261] David Gfeller, Aurélien Grosdidier, Matthias Wirth, Antoine Daina, Olivier Michielin, and Vincent Zoete. SwissTargetPrediction: a web server for target prediction of bioactive small molecules. *Nucleic Acids Research*, 42 (W1):W32–W38, 2014.

[262] Xian Liu, Yuan Xu, Shanshan Li, Yulan Wang, Jianlong Peng, Cheng Luo, Xiaomin Luo, Mingyue Zheng, Kaixian Chen, and Hualiang Jiang. In Silico target fishing: addressing a "Big Data" problem by ligand-based similarity

rankings with data fusion. *Journal of Cheminformatics*, 6 (1):33, 2014.

[263] Pedro J. Ballester and W. Graham Richards. Ultrafast shape recognition for similarity search in molecular databases. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Science*, 463(2081):1307–1321, 2007.

[264] G. M. Sastry, Steven L. Dixon, and Woody Sherman. Rapid Shape-Based Ligand Alignment and Virtual Screening Method Based on Atom/Feature-Pair Similarities and Volume Overlap Scoring. *Journal of Chemical Information and Modeling*, 51(10):2455–2466, 2011.

[265] Xiaofeng Liu, Hualiang Jiang, and Honglin Li. SHAFTS: A Hybrid Approach for 3D Molecular Similarity Calculation. 1. Method and Assessment of Virtual Screening. *Journal of Chemical Information and Modeling*, 51(9): 2372–2385, 2011.

[266] Jiayu Gong, Chaoqian Cai, Xiaofeng Liu, Xin Ku, Hualiang Jiang, Daqi Gao, and Honglin Li. ChemMapper: a versatile web server for exploring pharmacology and chemical structure association based on molecular 3D similarity method. *Bioinformatics*, 29(14):1827–1829, 2013.

[267] Chaoqian Cai, Jiayu Gong, Xiaofeng Liu, Daqi Gao, and Honglin Li. SimG: An Alignment Based Method for Evaluating the Similarity of Small Molecules and Binding Sites. *Journal of Chemical Information and Modeling*, 53(8): 2103–2115, 2013.

[268] M. Stuart Armstrong, Paul W. Finn, Garrett M. Morris, and W. Graham Richards. Improving the accuracy of

ultrafast ligand-based screening: incorporating lipophilicity into ElectroShape as an extra dimension. *Journal of Computer-Aided Molecular Design*, 25(8):785–790, 2011.

[269] Quoc Chinh Nguyen, Yew Soon Ong, Harold Soh, and Jer-Lai Kuo. Multiscale Approach to Explore the Potential Energy Surface of Water Clusters (H2O)nn 8. *The Journal of Physical Chemistry A*, 112(28):6257–6261, 2008.

[270] Edward Cannon, Florian Nigsch, and John Mitchell. A novel hybrid ultrafast shape descriptor method for use in virtual screening. *Chemistry Central Journal*, 2(1):3, 2008.

[271] Kun-Yi Hsin, Hugh P. Morgan, Steven R. Shave, Andrew C. Hinton, Paul Taylor, and Malcolm D. Walkinshaw. EDULISS: a small-molecule database with data-mining and pharmacophore searching capabilities. *Nucleic Acids Research*, 39(suppl 1):D1042–D1048, 2011.

[272] M. Stuart Armstrong, Garrett M. Morris, Paul W. Finn, Raman Sharma, and W. Graham Richards. Molecular similarity including chirality. *Journal of Molecular Graphics and Modelling*, 28(4):368–370, 2009.

[273] Ting Zhou, Karine Lafleur, and Amedeo Caflisch. Complementing ultrafast shape recognition with an optical isomerism descriptor. *Journal of Molecular Graphics and Modelling*, 29(3):443–449, 2010.

[274] M. Stuart Armstrong, Garrett M. Morris, Paul W. Finn, Raman Sharma, Loris Moretti, Richard I. Cooper, and W. Graham Richards. ElectroShape: fast molecular similarity calculations incorporating shape, chirality and electrostatics. *Journal of Computer-Aided Molecular Design*, 24(9):789–801, 2010.

[275] Jillian E. Adie. *Structure-based drug design of 11β-hydroxysteroid dehydrogenase type 1 inhibitors*. PhD thesis, University of Edinburgh, 2010.

[276] Niu Huang, Brian K. Shoichet, and John J. Irwin. Benchmarking Sets for Molecular Docking. *Journal of Medicinal Chemistry*, 49(23):6789–6801, 2006.

[277] Adrian Schreyer and Tom Blundell. CREDO: A Protein-Ligand Interaction Database for Drug Discovery. *Chemical Biology & Drug Design*, 73(2):157–167, 2009.

[278] Adrian M. Schreyer and Tom L. Blundell. CREDO: a structural interactomics database for drug discovery. *Database*, 2013, 2013.

[279] Michael M. Mysinger, Michael Carchia, John J. Irwin, and Brian K. Shoichet. Directory of Useful Decoys, Enhanced (DUD-E): Better Ligands and Decoys for Better Benchmarking. *Journal of Medicinal Chemistry*, 55(14):6582–6594, 2012.

[280] A. Patrícia Bento, Anna Gaulton, Anne Hersey, Louisa J. Bellis, Jon Chambers, Mark Davies, Felix A. Krüger, Yvonne Light, Lora Mak, Shaun McGlinchey, Michal Nowotka, George Papadatos, Rita Santos, and John P. Overington. The ChEMBL bioactivity database: an update. *Nucleic Acids Research*, 42(D1):D1083–D1090, 2014.

[281] Xiaofeng Liu, Fang Bai, Sisheng Ouyang, Xicheng Wang, Honglin Li, and Hualiang Jiang. Cyndi: a multi-objective evolution algorithm based method for bioactive molecular conformational generation. *BMC Bioinformatics*, 10(1): 101, 2009.

[282] Fang Bai, Xiaofeng Liu, Jiabo Li, Haoyun Zhang, Hualiang Jiang, Xicheng Wang, and Honglin Li. Bioactive confor-

mational generation of small molecules: A comparative analysis between force-field and multiple empirical criteria based methods. *BMC Bioinformatics*, 11(1):545, 2010.

[283] Paul C. D. Hawkins, A. G. Skillman, Gregory L. Warren, Benjamin A. Ellingson, and Matthew T. Stahl. Conformer Generation with OMEGA: Algorithm and Validation Using High Quality Structures from the Protein Databank and Cambridge Structural Database. *Journal of Chemical Information and Modeling*, 50(4):572–584, 2010.

[284] Jean-Paul Ebejer, Garrett M. Morris, and Charlotte M. Deane. Freely Available Conformer Generation Methods: How Good Are They? *Journal of Chemical Information and Modeling*, 52(5):1146–1158, 2012.

[285] Mikko J. Vainio and Mark S. Johnson. Generating Conformer Ensembles Using a Multiobjective Genetic Algorithm. *Journal of Chemical Information and Modeling*, 47 (6):2462–2474, 2007.

[286] Noel O'Boyle, Tim Vandermeersch, Christopher Flynn, Anita Maguire, and Geoffrey Hutchison. Confab - Systematic generation of diverse low-energy conformers. *Journal of Cheminformatics*, 3(1):8, 2011.

[287] Maria A. Miteva, Frederic Guyon, and Pierre Tufféry. Frog2: Efficient 3D conformation ensemble generator for small compounds. *Nucleic Acids Research*, 38(suppl 2): W622–W627, 2010.

[288] Raed Khashan. FragVLib a free database mining software for generating "Fragment-based Virtual Library" using pocket similarity search of ligand-receptor complexes. *Journal of Cheminformatics*, 4(1):18, 2012.

[289] Timo Krotzky, Christian Grunwald, Ute Egerland, and Gerhard Klebe. Large-Scale Mining for Similar Protein Binding Pockets: With RAPMAD Retrieval on the Fly Becomes Real. *Journal of Chemical Information and Modeling*, 55(1):165–179, 2015.

[290] Francois Berenger, Arnout Voet, Xiao Yin Lee, and Kam Zhang. A rotation-translation invariant molecular descriptor of partial charges and its use in ligand-based virtual screening. *Journal of Cheminformatics*, 6(1):23, 2014.

[291] Gaëlle Mariaule and Philippe Belmont. Cyclin-Dependent Kinase Inhibitors as Marketed Anticancer Drugs: Where Are We Now? A Short Survey. *Molecules*, 19(9):14366–14382, 2014.

[292] Adrian M. Senderowicz. Flavopiridol: the First Cyclin-Dependent Kinase Inhibitor in Human Clinical Trials. *Investigational New Drugs*, 17(3):313–320, 1999.

[293] Walter Filgueira De Azevedo, Sophie Leclerc, Laurent Meijer, Libor Havlicek, Miroslav Strnad, and Sung-Hou Kim. Inhibition of Cyclin-Dependent Kinases by Purine Analogues. *European Journal of Biochemistry*, 243(1-2):518–526, 1997.

[294] Nathalie Glab, Brahim Labidi, Li-Xian Qin, Christophe Trehin, Catherine Bergounioux, and Laurent Meijer. Olomoucine, an inhibitor of the cdc2/cdk2 kinases activity, blocks plant cells at the G1 to S and G2 to M cell cycle transitions. *FEBS Letters*, 353(2):207–211, 1994.

[295] Ursula Schulze-Gahmen, Hendrik L. De Bondt, and Sung-Hou Kim. High-Resolution Crystal Structures of Human Cyclin-Dependent Kinase 2 with and without ATP: Bound

Waters and Natural Ligand as Guides for Inhibitor Design. *Journal of Medicinal Chemistry*, 39(23):4540–4546, 1996.

[296] Vivian Law, Craig Knox, Yannick Djoumbou, Tim Jewison, An Chi Guo, Yifeng Liu, Adam Maciejewski, David Arndt, Michael Wilson, Vanessa Neveu, Alexandra Tang, Geraldine Gabriel, Carol Ly, Sakina Adamjee, Zerihun T. Dame, Beomsoo Han, You Zhou, and David S. Wishart. DrugBank 4.0: shedding new light on drug metabolism. *Nucleic Acids Research*, 42(D1):D1091–D1097, 2014.

[297] Mohammed A. Khedr, Tamer M. Shehata, and Maged E. Mohamed. Repositioning of 2,4-Dichlorophenoxy acetic acid as a potential anti-inflammatory agent: In Silico and Pharmaceutical Formulation study. *European Journal of Pharmaceutical Sciences*, 65(0):130–138, 2014.

[298] Hai-Jing Zhong, Li-Juan Liu, Daniel Shiu-Hin Chan, Hui-Min Wang, Philip Wai Hong Chan, Dik-Lung Ma, and Chung-Hang Leung. Structure-based repurposing of FDA-approved drugs as inhibitors of NEDD8-activating enzyme. *Biochimie*, 102(0):211–215, 2014.

[299] Ruili Huang, Noel Southall, Yuhong Wang, Adam Yasgar, Paul Shinn, Ajit Jadhav, Dac-Trung Nguyen, and Christopher P. Austin. The NCGC Pharmaceutical Collection: A Comprehensive Resource of Clinically Approved Drugs Enabling Repurposing and Chemical Genomics. *Science Translational Medicine*, 3(80):80ps16–80ps16, 2011.

[300] Minoru Kanehisa, Susumu Goto, Yoko Sato, Masayuki Kawashima, Miho Furumichi, and Mao Tanabe. Data, information, knowledge and principle: back to metabolism in KEGG. *Nucleic Acids Research*, 42(D1):D199–D205, 2014.

[301] Larry E. Millikan. Adapalene: an update on newer comparative studies between the various retinoids. *International Journal of Dermatology*, 39(10):784–788, 2000.

[302] Matthias Ocker, Christoph Herold, Marion Ganslmayer, Eckhart G. Hahn, and Detlef Schuppan. The synthetic retinoid adapalene inhibits proliferation and induces apoptosis in colorectal cancer cells in vitro. *International Journal of Cancer*, 107(3):453–459, 2003.

[303] Matthias Ocker, Christoph Herold, Marion Ganslmayer, Steffen Zopf, Eckhart G. Hahn, and Detlef Schuppan. Potentiated anticancer effects on hepatoma cells by the retinoid adapalene. *Cancer Letters*, 208(1):51–58, 2004.

[304] C. Hensby, D. Cavey, M. Bouclier, A. Chatelus, D. Algate, J. Eustache, and B. Shroot. The in vivo andin vitro anti-inflammatory activity of CD271: A new retinoid-like modulator of cell differentiation. *Agents and Actions*, 29 (1-2):56–58, 1990.

[305] T. Tashiro, Y. Kawada, Y. Sakurai, and Y. Kidani. Antitumor activity of a new platinum complex, oxalato (trans-l-1,2-diaminocyclohexane)platinum (II): new experimental data. *Biomedicine & Pharmacotherapy*, 43(4):251–260, 1989.

[306] Kalyan Das, James M. Aramini, Li-Chung Ma, Robert M. Krug, and Eddy Arnold. Structures of influenza A proteins and insights into antiviral drug targets. *Nature Structural & Molecular Biology*, 17(5):530–538, 2010.

[307] Yukiko Matsuoka, Hiromi Matsumae, Manami Katoh, Amie Eisfeld, Gabriele Neumann, Takeshi Hase, Samik Ghosh, Jason Shoemaker, Tiago Lopes, Tokiko Watanabe, Shinji Watanabe, Satoshi Fukuyama, Hiroaki Kitano, and

Yoshihiro Kawaoka. A comprehensive map of the influenza A virus replication cycle. *BMC Systems Biology*, 7(1):97, 2013.

[308] Arianna Loregian, Beatrice Mercorelli, Giulio Nannetti, Chiara Compagnin, and Giorgio Palù. Antiviral strategies against influenza virus: towards new therapeutic approaches. *Cellular and Molecular Life Sciences*, 71(19): 3659–3683, 2014.

[309] Kalyan Das. Antivirals Targeting Influenza A Virus. *Journal of Medicinal Chemistry*, 55(14):6263–6277, 2012.

[310] Juan Du, Timothy A. Cross, and Huan-Xiang Zhou. Recent progress in structure-based anti-influenza drug design. *Drug Discovery Today*, 17(19–20):1111–1120, 2012.

[311] Alicia Davis, Bryan Chabolla, and Laura Newcomb. Emerging antiviral resistant strains of influenza A and the potential therapeutic targets within the viral ribonucleoprotein (vRNP) complex. *Virology Journal*, 11(1):167, 2014.

[312] Qiaozhen Ye, Robert M. Krug, and Yizhi Jane Tao. The mechanism by which influenza A virus nucleoprotein forms oligomers and binds RNA. *Nature*, 444(7122):1078–1082, 2006.

[313] Andy Ka-Leung Ng, Hongmin Zhang, Kemin Tan, Zongli Li, Jin huan Liu, Paul Kay-Sheung Chan, Sui-Mui Li, Wood-Yee Chan, Shannon Wing-Ngor Au, Andrzej Joachimiak, Thomas Walz, Jia-Huai Wang, and Pang-Chui Shaw. Structure of the influenza virus A H5N1 nucleoprotein: implications for RNA binding, oligomerization, and vaccine design. *The FASEB Journal*, 22(10):3638–3647, 2008.

[314] Samuel W. Gerritz, Christopher Cianci, Sean Kim, Bradley C. Pearce, Carol Deminie, Linda Discotto, Brian McAuliffe, Beatrice F. Minassian, Shuhao Shi, Shirong Zhu, Weixu Zhai, Annapurna Pendri, Guo Li, Michael A. Poss, Suzanne Edavettal, Patricia A. McDonnell, Hal A. Lewis, Klaus Maskos, Mario Mörtl, Reiner Kiefersauer, Stefan Steinbacher, Eric T. Baldwin, William Metzler, James Bryson, Matthew D. Healy, Thomas Philip, Mary Zoeckler, Richard Schartman, Michael Sinz, Victor H. Leyva-Grado, Hans-Heinrich Hoffmann, David R. Langley, Nicholas A. Meanwell, and Mark Krystal. Inhibition of influenza virus replication via small molecules that induce the formation of higher-order nucleoprotein oligomers. *Proceedings of the National Academy of Sciences*, 108(37):15366–15371, 2011.

[315] Qiaozhen Ye, Tom S. Y. Guu, Douglas A. Mata, Rei-Lin Kuo, Bartram Smith, Robert M. Krug, and Yizhi J. Tao. Biochemical and Structural Evidence in Support of a Coherent Model for the Formation of the Double-Helical Influenza A Virus Ribonucleoprotein. *mBio*, 4(1), 2013.

[316] Sylvie Chenavas, Leandro F. Estrozi, Anny Slama-Schwok, Bernard Delmas, Carmelo Di Primo, Florence Baudin, Xinping Li, Thibaut Crépin, and Rob W. H. Ruigrok. Monomeric Nucleoprotein of Influenza A Virus. *PLoS Pathogens*, 9(3):e1003275, 2013.

[317] Wai-Hon Chan, Andy Ka-Leung Ng, Nicole C. Robb, Mandy Ka-Han Lam, Paul Kay-Sheung Chan, Shannon Wing-Ngor Au, Jia-Huai Wang, Ervin Fodor, and Pang-Chui Shaw. Functional Analysis of the Influenza Virus H5N1 Nucleoprotein Tail Loop Reveals Amino Acids That Are Crucial for Oligomerization and Ribonucleoprotein Activities. *Journal of Virology*, 84(14):7337–7345, 2010.

[318] Yu-Fang Shen, Yu-Hou Chen, Shao-Ying Chu, Meng-I Lin, Hua-Ting Hsu, Pei-Yu Wu, Chao-Jung Wu, Hui-Wen Liu, Fu-Yang Lin, Gialih Lin, Pang-Hung Hsu, An-Suei Yang, Yih-Shyun E. Cheng, Ying-Ta Wu, Chi-Huey Wong, and Ming-Daw Tsai. E339···R416 salt bridge of nucleoprotein as a feasible target for influenza virus inhibitors. *Proceedings of the National Academy of Sciences*, 108(40):16515–16520, 2011.

[319] Andreas Kukol and David John Hughes. Large-scale analysis of influenza A virus nucleoprotein sequence conservation reveals potential drug-target sites. *Virology*, 454–455 (0):40–47, 2014.

[320] Xiaojing He, Jie Zhou, Mark Bartlam, Rongguang Zhang, Jianyuan Ma, Zhiyong Lou, Xuemei Li, Jingjing Li, Andrzej Joachimiak, Zonghao Zeng, Ruowen Ge, Zihe Rao, and Yingfang Liu. Crystal structure of the polymerase PAC-PB1N complex from an avian influenza H5N1 virus. *Nature*, 454(7208):1123–1126, 2008.

[321] Eiji Obayashi, Hisashi Yoshida, Fumihiro Kawai, Naoya Shibayama, Atsushi Kawaguchi, Kyosuke Nagata, Jeremy R. H. Tame, and Sam-Yong Park. The structural basis for an essential subunit interaction in influenza virus RNA polymerase. *Nature*, 454(7208):1127–1131, 2008.

[322] Spencer O. Moen, Jan Abendroth, James W. Fairman, Ruth O. Baydo, Jameson Bullen, Jennifer L. Kirkwood, Steve R. Barnes, Amy C. Raymond, Darren W. Begley, Greg Henkel, Ken McCormack, Vincent C. Tam, Isabelle Phan, Bart L. Staker, Robin Stacy, Peter J. Myler, Don Lorimer, and Thomas E. Edwards. Structural analysis of H1N1 and H7N9 influenza A virus PA in the absence of PB1. *Scientific Reports*, 4:5944, 2014.

[323] Alexander Ghanem, Daniel Mayer, Geoffrey Chase, Werner Tegge, Ronald Frank, Georg Kochs, Adolfo García-Sastre, and Martin Schwemmle. Peptide-Mediated Interference with Influenza A Virus Polymerase. *Journal of Virology*, 81(14):7801–7804, 2007.

[324] Kerstin Wunderlich, Daniel Mayer, Charlene Ranadheera, Anne-Sophie Holler, Benjamin Mänz, Arnold Martin, Geoffrey Chase, Werner Tegge, Ronald Frank, Ulrich Kessler, and Martin Schwemmle. Identification of a PA-Binding Peptide with Inhibitory Activity against Influenza A and B Virus Replication. *PLoS ONE*, 4(10):e7517, 2009.

[325] Kerstin Wunderlich, Mindaugas Juozapaitis, Charlene Ranadheera, Ulrich Kessler, Arnold Martin, Jessica Eisel, Ulrike Beutling, Ronald Frank, and Martin Schwemmle. Identification of High-Affinity PB1-Derived Peptides with Enhanced Affinity to the PA Protein of Influenza A Virus Polymerase. *Antimicrobial Agents and Chemotherapy*, 55 (2):696–702, 2011.

[326] Geoffrey Chase, Kerstin Wunderlich, Peter Reuther, and Martin Schwemmle. Identification of influenza virus inhibitors which disrupt of viral polymerase protein–protein interactions. *Methods*, 55(2):188–191, 2011.

[327] Mayuko Fukuoka, Moeko Minakuchi, Atsushi Kawaguchi, Kyosuke Nagata, Yuji O. Kamatari, and Kazuo Kuwata. Structure-based discovery of anti-influenza virus A compounds among medicines. *Biochimica et Biophysica Acta (BBA) - General Subjects*, 1820(2):90–95, 2012.

[328] Giulia Muratore, Laura Goracci, Beatrice Mercorelli, Ágnes Foeglein, Paul Digard, Gabriele Cruciani, Giorgio Palù, and Arianna Loregian. Small molecule inhibitors of influenza A and B viruses that act by disrupting subunit

interactions of the viral polymerase. *Proceedings of the National Academy of Sciences*, 109(16):6247–6252, 2012.

[329] Serena Massari, Giulio Nannetti, Laura Goracci, Luca Sancineto, Giulia Muratore, Stefano Sabatini, Giuseppe Manfroni, Beatrice Mercorelli, Violetta Cecchetti, Marzia Facchini, Giorgio Palù, Gabriele Cruciani, Arianna Loregian, and Oriana Tabarrini. Structural Investigation of Cycloheptathiophene-3-carboxamide Derivatives Targeting Influenza Virus Polymerase Assembly. *Journal of Medicinal Chemistry*, 56(24):10118–10131, 2013.

[330] Cristina Tintori, Ilaria Laurenzana, Anna Lucia Fallacara, Ulrich Kessler, Beatrice Pilger, Lilli Stergiou, and Maurizio Botta. High-throughput docking for the identification of new influenza A virus polymerase inhibitors targeting the PA–PB1 protein–protein interaction. *Bioorganic & Medicinal Chemistry Letters*, 24(1):280–282, 2014.

[331] Mafalda Pagano, Daniele Castagnolo, Martina Bernardini, Anna Lucia Fallacara, Ilaria Laurenzana, Davide Deodato, Ulrich Kessler, Beatrice Pilger, Lilli Stergiou, Stephan Strunze, Cristina Tintori, and Maurizio Botta. The Fight against the Influenza A Virus H1N1: Synthesis, Molecular Modeling, and Biological Evaluation of Benzofurazan Derivatives as Viral RNA Polymerase Inhibitors. *ChemMedChem*, 9(1):129–150, 2014.

[332] Susan Lepri, Giulio Nannetti, Giulia Muratore, Gabriele Cruciani, Renzo Ruzziconi, Beatrice Mercorelli, Giorgio Palù, Arianna Loregian, and Laura Goracci. Optimization of Small-Molecule Inhibitors of Influenza Virus Polymerase: From Thiophene-3-Carboxamide to Polyamido Scaffolds. *Journal of Medicinal Chemistry*, 57(10):4337–4350, 2014.

[333] John T. A Hsu, Jiann-Yih Yeh, Ta-Jen Lin, Mei ling Li, Ming-Sian Wu, Chung-Fan Hsieh, Yao Chieh Chou, Wen-Fang Tang, Kean Seng Lau, Hui-Chen Hung, Ming-Yu Fang, Shengkai Ko, Hsing-Pang Hsieh, and Jim-Tong Horng. Identification of BPR3P0128 as an Inhibitor of Cap-Snatching Activities of Influenza Virus. *Antimicrobial Agents and Chemotherapy*, 56(2):647–657, 2012.

[334] Delphine Guilligay, Franck Tarendeau, Patricia Resa-Infante, Rocio Coloma, Thibaut Crepin, Peter Sehr, Joe Lewis, Rob W. H. Ruigrok, Juan Ortin, Darren J. Hart, and Stephen Cusack. The structural basis for cap binding by influenza virus polymerase subunit PB2. *Nature Structural & Molecular Biology*, 15(5):500–506, 2008.

[335] Yong Liu, Kun Qin, Geng Meng, Jinfang Zhang, Jianfang Zhou, Guangyu Zhao, Ming Luo, and Xiaofeng Zheng. Structural and Functional Characterization of K339T Substitution Identified in the PB2 Subunit Cap-binding Pocket of Influenza A Virus. *Journal of Biological Chemistry*, 288(16):11013–11023, 2013.

[336] Michael P. Clark, Mark W. Ledeboer, Ioana Davies, Randal A. Byrn, Steven M. Jones, Emanuele Perola, Alice Tsai, Marc Jacobs, Kwame Nti-Addae, Upul K. Bandarage, Michael J. Boyd, Randy S. Bethiel, John J. Court, Hongbo Deng, John P. Duffy, Warren A. Dorsch, Luc J. Farmer, Huai Gao, Wenxin Gu, Katrina Jackson, Dylan H. Jacobs, Joseph M. Kennedy, Brian Ledford, Jianglin Liang, François Maltais, Mark Murcko, Tiansheng Wang, M. W. Wannamaker, Hamilton B. Bennett, Joshua R. Leeman, Colleen McNeil, William P. Taylor, Christine Memmott, Min Jiang, Rene Rijnbrand, Christopher Bral, Ursula Germann, Azin Nezami, Yuegang Zhang, Francesco G. Salituro, Youssef L. Bennani,

and Paul S. Charifson. Discovery of a Novel, First-in-Class, Orally Bioavailable Azaindole Inhibitor (VX-787) of Influenza PB2. *Journal of Medicinal Chemistry*, 57(15): 6668–6678, 2014.

[337] Toshiharu Tsurumura, Hao Qiu, Toru Yoshida, Yayoi Tsumori, Dai Hatakeyama, Takashi Kuzuhara, and Hideaki Tsuge. Conformational Polymorphism of m7GTP in Crystal Structure of the PB2 Middle Domain from Human Influenza A Virus. *PLoS ONE*, 8(11):e82020, 2013.

[338] Toshiharu Tsurumura, Hao Qiu, Toru Yoshida, Yayoi Tsumori, and Hideaki Tsuge. Crystallization and preliminary X-ray diffraction studies of a surface mutant of the middle domain of PB2 from human influenza A (H1N1) virus. *Acta Crystallographica Section F*, 70(1):72–75, 2014.

[339] Stéphane Pautus, Peter Sehr, Joe Lewis, Antoine Fortuné, Andrea Wolkerstorfer, Oliver Szolar, Delphine Guilligay, Thomas Lunardi, Jean-Luc Décout, and Stephen Cusack. New 7-Methylguanine Derivatives Targeting the Influenza Polymerase PB2 Cap-Binding Domain. *Journal of Medicinal Chemistry*, 56(21):8915–8930, 2013.

[340] Alexander Pflug, Delphine Guilligay, Stefan Reich, and Stephen Cusack. Structure of influenza A polymerase bound to the viral RNA promoter. *Nature*, 516(7531): 355–360, 2014.

[341] Stefan Reich, Delphine Guilligay, Alexander Pflug, Helene Malet, Imre Berger, Thibaut Crepin, Darren Hart, Thomas Lunardi, Max Nanao, Rob W. H. Ruigrok, and Stephen Cusack. Structural insight into cap-snatching and RNA synthesis by influenza polymerase. *Nature*, 516 (7531):361–366, 2014.

[342] Tiziano Tuccinardi, Giulio Poli, Veronica Romboli, Antonio Giordano, and Adriano Martinelli. Extensive Consensus Docking Evaluation for Ligand Pose Prediction and Virtual Screening Studies. *Journal of Chemical Information and Modeling*, 54(10):2980–2986, 2014.