

Supercomputer Docking

Alexey V. Sulimov^{1,2}, *Danil C. Kutov*^{1,2}, *Vladimir B. Sulimov*^{1,2}

© The Authors 2019. This paper is published with open access at SuperFri.org

This review is based on the peer-reviewed research literature including the author's own publications devoted to supercomputer docking. The general view on docking and its role at the initial stage of the rational drug design is presented. Molecules of medicine compounds selectively bind to the active site of a protein, which is responsible for the disease progression, and stop it. Docking programs perform positioning of molecules (ligands) in the active site of the protein and estimate the protein–ligand binding energy. The larger this energy is, the less concentration of the respective compound should be used to observe the desired effect. Several classical docking programs are described in short. Examples of the adaptation of existing docking programs to supercomputing and using them for virtual screening of millions of ligands are presented. Two novel generalized docking programs specially designed for multi-core docking of a single ligand on a supercomputer are described shortly. These programs find a sufficiently wide spectrum of low energy minima of a protein–ligand complex in the frame of a given force field. The quasi-docking procedure using the generalized docking program is described. Quasi-docking allows to perform docking with quantum-chemical semiempirical methods. Finally a summary is made based on the materials presented.

Keywords: docking, protein–ligand, global optimization, tensor train, force field, quantum-chemical method.

Introduction

Docking programs are extremely demanded for the computer aided structural based drug design [78, 80]. The latter is used at the initial stage of the long (10–15 years) and expensive (> 1 billion USD) rational drug design pipeline. This initial stage is the cheapest of all the following stages: preclinical testing (animals), three phases of clinical stages (humans), approving and post approving stages, but the initial stage plays the key role in the whole pipeline success defining the diversity of the active compound–candidates, their selectivity and low toxicity. The main idea of the rational drug development is to find a compound, the molecules of which bind specifically to a definite region (active site) of the given bio-molecule, e.g. a protein, responsible for the disease progression, and stop or change the latter. The larger this binding energy is, the lesser concentration of the active compound should be used to achieve the desired effect. Docking programs carry out positioning of the molecules–candidates (ligands) in the definite site of the target protein and calculate the protein–ligand binding free energy [95] which is directly connected with the measured binding (dissociating) constant of the active compound. There are several dozen of docking programs and a dozen of Internet docking sites available either for free or on a commercial basis [14, 66]. Despite the obvious progress of the docking technique and plenty of success stories, there are still various challenges [9, 96, 107]. While in many cases positioning accuracy of docking is satisfactory, the accuracy of binding energy calculations is insufficient to perform the hit-to-lead optimization on the base of docking results. The increase of docking accuracy should result in an increase in the effectiveness of the whole process of new drug development [90].

The work of many docking programs relies on the docking paradigm which assumes that the ligand binds in the active site of the target protein at the global energy minimum of the

¹Research Computer Center of Lomonosov Moscow State University, Moscow, Russian Federation

²Dimonta Ltd., Moscow, Russian Federation

protein–ligand system [60, 61]. Thus, the docking problem turns into the global optimization problem. The main challenge of docking is a high dimensionality and complicated relief of the energy surface where the global energy minimum should be found. For example, a therapeutic molecule often has 10 internal rotational degrees of freedom (torsions). Thus, the protein–ligand system energy surface has $16 = 10 + 6$ dimensions where 6 dimensions correspond to translations and rotations of the ligand as a rigid body. This case corresponds to the rigid target protein. However, if protein flexibility is taken into account, the dimension of the protein–ligand energy surface increases in line with additional number of protein degrees of freedom, e.g. for every mobile protein atom additional 3 dimensions, atom Cartesian coordinates, should be added. On the other hand, such a docking program should be able to perform docking of many ligands for the reasonable period of time. This contradiction results in many different simplifications implemented in docking programs, especially at the dawn of the docking technique development 25 – 30 years ago when available computer resources were very limited. The “Faster, even faster” mantra long time was the main goal of docking algorithms and some of popular docking programs can dock one ligand at one processor for less than 1 minute even less, e.g. the ICM program can run docking of one ligand for 10 seconds [57]. However over time, more attention has been paid to the docking accuracy, and in some popular docking programs, for example in Glide, in addition of Standard Precision (SP) calculations the Extra Precision mode (XP) has been added [27], and the latter needs much more computing resources. If program docks one ligand at one computing core for the time of half an hour or longer, the use of this program for virtual screening of large databases containing many thousands of ligands inevitably needs supercomputer resources. On the other hand, either for the too complex form of the protein active site and respective complicated protein–ligand energy surface or for the too large number of ligand torsions it might be needed to increase considerably the thoroughness of the global energy minimum search at the expense of a large increase of the docking time of one ligand. This increase can be compensated by the multi–processor performance of the docking program at several dozen or hundred computing cores of a supercomputer. Additional processing of docking results can also demand supercomputer resources especially if the protein–ligand binding energy refinement needs an employment of molecular dynamics (MD) or quantum chemistry (QC) [51]. Actually MD methods have not been used for virtual screening of large ligand databases, but these methods of the protein–ligand binding free energy calculation are applied usually at the post–docking stage for a further refinement of the binding energy of best selected ligands and to optimize them. MD methods being applied to the binding free energy calculations have a lot of specific features and tricks [45], e.g. artificial lowering of energy barriers, and a detailed description of all of them needs a lot of space and it is out of the scope of this review. We only note here several points. First, MD methods and docking have some common features defining accuracy of the calculations and the most important one is the energy calculation method using either a force field or a quantum chemistry method. Second, calculations of the absolute protein–ligand binding free energy is a much more challenging problem than calculations of the relative binding energy and in most publications just the latter is calculated with MD [105]. Third, most of these MD methods sample the Boltzmann distribution of molecular configurations coupled to a heat bath at a given temperature, and same sampling can be done also with Monte Carlo simulations [16]. Fourth, there are several methods to perform the protein–ligand binding free energy calculations [46], and the most widely used ones are thermodynamic integration (TI) [28, 42, 53] and free energy perturbation (FEP) [19] methods, as well as the recently proposed enveloping

distribution sampling (EDS method [70, 75]; a recent survey of the state of art of the binding free energy calculation method is presented in [54] (see also a perpetual review of David L. Mobley, et al. at <https://github.com/MobleyLab/benchmarksets/blob/master/paper/benchmarkset.pdf>). Fifth, the protein–ligand binding affinity can be associated with the ratio of kinetically determined rates k_{off} and k_{on} of complex formation [44] and MD simulation of binding kinetics can become a useful tool to get insight of a mechanism of protein–ligand binding [67]. Sixth, MD penetrates into the docking technology in different ways from prior generation of an ensemble of protein conformations to take into account protein flexibility performing docking into discret rigid pre-generated protein conformations (ensemble docking) to the so-called solvent mapping for the identification of binding sites and hot-spots on protein surfaces [18, 24, 30]. Some additional recent examples of ensemble docking can be found in [2, 3, 5, 22, 23, 82, 86, 106]. All these methods require large computational resources and supercomputing power should be demanded along this road.

Quantum chemistry (QC) plays an important role in modeling for drug design. However, for the energy calculation in docking proper QC methods have not been used until recently. The latter is connected with the inability of QC programs, based on *ab initio* including DFT methods, to perform calculations of molecular systems containing thousands of atoms. Even the model of a protein–ligand complex reduced to 5–6 hundred atoms containing only the ligand and the active site of the target cannot be used for docking with QC *ab initio* methods due to the time limitation and limited capacity of multi-processor performance. Nevertheless, some recently developed QC semiempirical methods together with the localized molecular orbitals method can be used for post-docking refinement of the energy of the different conformations of the protein–ligand system (see below the “Quasi-docking” section). The QM/MM approximation where the most important part of the molecular system is treated with QC methods (QM abbreviation) and the rest of the system is described at a level of classical molecular mechanics (MM abbreviation). For example, QM/MM was used in [4] for the post-docking optimization of the protein–ligand system using the old AM1 semiempirical method, and the best ligand poses were found at the preliminary step by docking with the AutoDock program. Despite the good results demonstrated in [4] eclecticism of this attempt are obvious because docking is performed in the frame of an empirical oversimplified AutoDock force field (a set of classical potentials describing inter- and intra-molecular interactions), MM part is calculated with another, Charmm, force field and for QM part the AM1 method badly describing hydrogen bonds and dispersion inter-molecular interactions [102] is used. Different approaches of QC used for the binding affinity refinement in post-docking processing can be found in [15, 41, 77]. So, QC-docking and QC molecular dynamics for the accurate protein–ligand binding free energy calculations are still waiting for their realization, and this will happen in near future due to supercomputer facilities. QC methods are also used either for creation of whole force fields or for improving existing force fields, by modifications of atomic charges on the base of QC calculations. The most widely recognized example of the latter is the GAFF force field [104] where atomic charges are calculated with QC, and Van der Waals parameters are taken from the empirical potentials. Some examples of this approach are also presented in [8, 103], but there are much more. However, we should note that atomic charges are the part of a whole set of force field parameters describing inter- and intra-molecular interactions, and these parameters must be obtained in a self-consistent manner.

The molecular models of the target protein and ligands are very important for the proper docking performance, as well as MD calculations. 3D structures of proteins and protein–ligand complexes stored in Protein Data Bank [6, 13] do not usually contain hydrogen atoms. Hydrogen atoms are added to the protein structures with the help of a particular programs. The resulted 3D structures obtained after such different programs performance are different from one another, and this influences strongly on docking [47] due to the difference in the spatial positions of the added hydrogen atoms and due to difference in the protonation states of some amino acids residues. The best way to define protonation state is based on QC calculations but usually this is done with standard rules for separated amino acids. Certainly, results of ligand docking also strongly depends on their protonation states, as well as on enantiomery, tautomerism, and in general QC methods (taking into account electron correlation at sufficiently high *ab initio* level) should be used for a proper preparation of the respective enantiomers and tautomers together with low energy stereoisomers of a given ligand. The latter is closely connected to the determination of the lowest energy conformation of the unbound ligand in water solution defining the ligand internal stress energy which is an important term of the binding free energy. Unfortunately, in many cases QC methods still are not used for these purposes. Some examples are presented in [10, 11, 40, 68].

Nowadays machine learning invades many areas of science, and docking is not an exception. For example, convolutional neural networks (CNNs), which are commonly used in the image recognition [50], are applied in [71] to the construction of the scoring function to discriminate between correct and incorrect ligand binding poses and known binders and nonbinders, and it is shown that the CNN scoring function outperforms the AutoDock Vina one. The more sophisticated use of deep learning is presented in [49] where RAVE method (the reweighted autoencoded variational Bayes for enhanced sampling) [74] is used for calculating absolute protein–ligand binding free energies. Other recent examples of machine learning application to docking and scoring are presented in [58, 59, 69, 99]. In connection with this dawn of new era of machine learning application we should make a remark that databases of experimentally measured protein–ligand binding affinities contain many hidden uncertainties and sometimes errors, and before their use for machine learning, these databases should be strongly and cleverly filtered.

Considering that reviews on classical (non–supercomputer) docking programs are published regularly [14, 66, 96, 107], we review here mainly works devoted either to development of supercomputer docking programs, or to the application of supercomputers to virtual screening of large databases of ligands and refinement of docking results. We restrict this review by only docking of small molecular weight ligands into proteins. The protein–protein docking is out of the scope of this work, but all problems discussed here especially protein flexibility and docking accuracy are key features of such docking programs. Two examples of a protein–protein docking program, supercomputer docking and webserver, are presented in [62, 110]. Firstly, we take a look at the docking problem and consider examples of classical docking programs and their peculiarities. Secondly, we present a review of publications on docking in conjunction with supercomputers. Finally, some concluding remarks are presented.

1. Docking: General View

For brevity we will call as inhibitors those molecules (ligands) that bind to a given target proteins. Inhibitors block functioning of the target proteins, but in general ligands upon binding can activate target proteins or change their work. Usually a low molecular weight ligand contains

several dozen atoms. Target proteins contain several thousand atoms, and 3D structures of many target proteins, Cartesian coordinates of all protein atoms, can be free downloaded from the Protein Data Bank (PDB) Web-site (<https://www.rcsb.org>) [6, 13]. This database contains now more than 150 thousand entries of biological macromolecular structures (protein, DNA, RNA) and their complexes with ligands.

Docking is used to discover non-covalent inhibitors of a given target protein. This means that a ligand and a protein are bound with one another by weak inter-molecular interactions, this binding is reversible and the protein-ligand binding free energy ΔG_{bind} is directly related to the binding constant k_b associated with protein $[P]$, ligand $[L]$ and their complex $[PL]$ concentrations in water solvent:

$$k_b = \frac{[PL]}{[P] \times [L]} = e^{\frac{-\Delta G_{bind}}{RT}}, \quad (1)$$

where the dimensionality of the binding constant is mol/L , R is the universal gas constant, and T is absolute temperature in K . The binding free energy is defined as $\Delta G_{bind} = G_{PL} - G_P - G_L$, where G_{PL} , G_P and G_L are Gibbs free energies of the protein-ligand complex, the protein and the ligand, respectively. In its turn the Gibbs free energy of a molecular system X can be calculated through the respective configuration integral $Z_X = -RT \ln(G_X)$ which is calculated over the phase space of the molecular system X . If the low energy relief of the molecular system X is known, the configuration integral can be calculated. Moreover, if the global energy minimum is notably (comparing with kT) stands out by its energy in respect with energies of other low energy minima, the molecular system spends most time in the global minimum, and the configuration integral and corresponding Gibbs free energy of the system is defined by only the global energy minimum value and by the form of the respective energy well.

In many cases target proteins can be considered as rigid systems and the global energy minima search should be performed for only the protein-ligand complex and for the unbound ligand while the energy of the unbound protein is calculated for only one its configuration defined by atom coordinates taken for PDB. PDB-files usually contain only coordinates of heavy (non-hydrogen) atoms. Several thousand hydrogen atoms should be added to the protein structure using the appropriate program. Note that different programs add hydrogen atoms by different ways which result, first, in different protonation (charge) states of some molecular groups, the so-called titrated groups, and, second, in various hydrogen atoms positions, i.e. the covalent bonds of these hydrogen atoms can be differently oriented in space. These differences in spatial positions of added hydrogen atoms lead to different docking results and to different results of binding energy calculations [47].

One of the most important challenge of docking is the choice of the method of the molecular energy calculation which should include both inter- and intra-molecular interactions, as well as take into account the interaction of molecules with the water solvent. Quantum mechanics (QM) or, what is the same, quantum chemistry should be used to describe molecular interactions properly. However, these methods could not be used until recently for the molecular systems containing several thousand atoms. Instead QM methods classical potentials, force fields, describing interatomic interactions are used. There are plenty of force fields created for the classical molecular dynamics simulations and docking: AMBER [104], CHARMM [7], MMFF94 [32], OPLS-AA [39], etc., which describe molecular interactions sufficiently satisfactorily but not without some faults in different particular cases.

However, even such a dramatic simplification as a changeover from QM methods to force fields is insufficient for the realization of the docking procedure as a global energy minimum search. The calculation of all pairwise interactions between all ligand atoms and all protein atoms for every protein–ligand complex configuration in the frame of any force field mentioned above is computationally too costly. Apart from strong simplification of force fields a model of preliminary calculated grid of potentials is implemented in many docking programs: potentials of ligand probe atoms interactions with the all protein atoms are calculated and stored at nodes of the grid covering the active site of the target protein. Then, in the course of the global energy minimum search the energy of the ligand in any position in the active site of the protein is calculated as the sum of the grid potentials over all ligand atoms. This grid approximation gives a large acceleration because all resource–intensive operations are performed at the grid calculation stage before the global optimization. However, the grid approximation restricts docking and its accuracy in several aspects. Firstly, it is impossible to make sufficiently accurate local optimization of the protein–ligand energy with variation of ligand atoms positions. Secondly, it is impossible to take into account mobility of protein atoms in the docking procedure. Thirdly, water solvent plays an important role in molecular energy calculations due to the high dielectric constant of water (78.4 at the room temperature) and respective strong screening of Coulomb interactions; in order to avoid to take into account thousands of explicit water molecules, implicit solvent models are used instead. In the frame of these models water solvent is described by a homogeneous dielectric continuum surrounding a molecule of solute and the nonlocal interactions of atom charges of the solute molecule with polarization charges on the solvent excluded surface (SES) cannot be sufficiently accurately reduced to predetermined local potentials at grid nodes. Fourthly, the fast increase of computing power will soon lead to the opportunity to utilize QM methods in docking, but such self–consistent methods cannot be used in the frame of the grid approximation.

The will to avoid the grid approximation, to use implicit solvent models, to take into account flexibility of ligand and protein simultaneously and by this to increase the docking accuracy leads to necessity to draw larger computer power for docking of one ligand and to the transition from notebook docking to supercomputer docking. Before the discussion of supercomputer docking in the next section we describe shortly some examples of popular classical docking programs to outline their limitations and to contrast them with supercomputer docking programs. The more or less exhaustive reviews of such docking programs are presented in [14, 66, 96, 107].

1.1. AutoDock

This is the most popular docking package [14]. It is a free opensource code actually including two main docking engines employing a pre–calculated grid of potentials of probe ligand atoms interactions with the target protein: AutoDock Vina [98] and AutoDock [26, 56], and also auxiliary programs Raccoon2 (a graphical tool), AutoDockTools for the preparation of molecules and for the analysis of docking results and AutoLigand for predicting optimal sites of ligand binding. AutoDock Vina and AutoDock use stochastic methods for the global energy minimum search and the docking algorithms work mainly for the rigid protein model, although a possibility of taking into account protein flexibility is also implemented in AutoDock [55]. However the latter can be used only for very limited number of degrees of freedom: ligands should have less than 10 torsions and additional six degrees of freedom of two protein side chains. Actually ligand flexibility is described by only ligand torsions while bond lengths and valence angles are kept

fixed. AutoDock Vina uses an oversimplified empirical energy function and the conformational search is based on a gradient local optimization method. The energy function of AutoDock Vina does not contain the Coulomb term and consequently it considers electrostatic interactions only implicitly through the hydrophobic and the hydrogen bonding terms, it uses spherical potentials for hydrogen bonds and treats hydrogen atoms only implicitly. AutoDock uses more sophisticated but also a simplified empirical force field including directed hydrogen bond terms, explicit polar hydrogen atoms and Coulomb and van der Waals interactions and desolvation effects are treated in a very simplified local potential form. AutoDock uses the Lamarckian genetic algorithm for the global energy optimization [55]: a population of trial individuals, ligand conformations, is created, and then in successive generations these individuals mutate, exchange conformational parameters through the crossingover procedure, and compete on their energy selecting individuals with lowest energy, and the “Lamarckia” feature allows individual to search their local conformational space, finding local minima, and then pass this information to later generations.

1.2. DOCK

This is the oldest docking program which was created more than 35 years ago. The initial version of this program perform rigid docking by matching ligand and receptor shapes on a steric overlap. Subsequently a force field, the energy minimization, solvent, ligand and protein flexibility, multiprocessor performance were implemented in it [1, 12]. The orientation search of ligand poses is performed by matching ligand heavy atoms to matching points inside the docking domain. The matching points are generated by filling the docking domain with spheres of varying radii (1.4 – 2.5 Å); centers of these spheres become matching points. The preliminary calculated grid of potentials of probe ligand atom interactions with all protein atoms is used. There are different energy (scoring) functions: from simplest bump filtering that counts the number of van der Waals clashes between ligand and protein atoms and discard ligand poses that exceed a predetermined threshold to the energy score defined by Lennard–Jones and Coulomb potentials of the AMBER force field together with desolvation scoring based on the Generalized Born or the Poisson–Boltzmann implicit solvent models for the polar solute–solvent interaction plus the solvent accessible surface area (SASA) approximation for the non–polar term of the desolvation energy. The latter is a linear form of SASA. In the process of finding low–energy minima DOCK6 uses a combination of global sampling and a local energy minimization performed by the simplex method: rigid docking, including docking of different rigid ligand conformers, plays a central role in the performance of DOCK6, and the found ligand poses can be used for the further local energy minimization and *MD* simulation. The Anchor–and–Grow algorithm for flexible ligand docking is a distinguishable core feature of the DOCK6 program. A ligand is disassembled into non–flexible fragments connected by rotatable bonds. The largest fragments (anchors) are used to generate a set of possible anchor poses within the binding site. From these poses of each anchor a search for possible conformations of the ligand flexible parts is performed, and rigid fragments are flexibly attached to the anchor in layers until the whole ligand is reconstructed. When each ligand fragment is grown, the conformer with minimal energy is selected before proceeding to the next growing step. A key advantage of this algorithm is that the ligand conformation is left within the confinements of the binding site, radically reducing the conformational space of the search.

1.3. SOL

The main features of this program are [91, 95]: (i) rigid protein model; (ii) protein active site is represented by a grid of potentials describing interactions (electrostatic, van der Waals interactions) of all protein atoms with all possible types of ligand probe atoms in the frame of the MMFF94 force field practically without serious simplifications; (iii) inclusion of desolvation effects on the base of a simplified form of Generalized Born approximation [76, 85] allowing to store the corresponding desolvation potentials at nodes of the preliminary calculated grid; (iv) genetic algorithm is used for the global energy optimization; (v) ligand stress energy is taken into account in the frame of MMFF94 during the global optimization, i.e. the objective energy function of the global optimization consists of two terms: the ligand energy in the field of all protein atoms and the ligand stress energy. The latter is the energy of the ligand deformation. The desolvation energy is defined by the difference of the solvation energy of the bound protein–ligand complex and solvation energies of the unbound protein and ligand. The effect of desolvation is due to the fact that when a ligand binds to a protein, some of the surface protein and ligand atoms stop interacting with the solvent, water.

The docking region is defined by the position of the grid of preliminary calculated potentials. The grid is created in the cube (the docking cube) covering the whole protein active site. By default the cube edge is equal to 22 \AA , and the cube contains $101 \times 101 \times 101$ uniformly distributed grid nodes. The size and the position of the cube center are user defined. During docking performed by the global optimization a ligand can be randomly positioned at any pose inside the docking cube and it can have any conformation defined by random variations of torsions keeping fixed ligand bond lengths and valence angles. The grid is calculated by the SOLGRID module before the global optimization process. Opposite to many popular empirical force fields such as AMBER, CHARMM, OPLS-AA, the MMFF94 force field is considered as a physical *ab initio* force field because all its parameters are fitted to the results of *ab initio* quantum–chemical calculations for a broad set of organic molecules. MMFF94 has a well–defined procedure of the atom typification applicable to an arbitrary organic compound. This typification defines the force field parameters for any target protein and for almost any ligand. The atom typification, as well as the hydrogen atoms addition are made by the APLITE program. For SOLGRID calculations the MMFF94 atom typification is somewhat simplified up to 27 atom types (instead of 99 types) to keep the potentials in the computer RAM memory. So, the grid contains in its nodes 27 van der Waals and 27 Coulomb potentials of interactions of a respective ligand probe atom with all protein atoms and respective desolvation potentials. Note, that van der Waals potentials defined by equations of the MMFF94 force field undergo some broadening ($\approx 0.3 - 0.5 \text{ \AA}$) which can be defined in the set of SOLGRID input parameters. Broadening imitates restricted mobility of protein atoms. SOLGRID consumes from 1 to several hours at one computing core depending on the target protein size, and a binary file of about 200 Mb is created. When the docking procedure is executed by the SOL module the ligand energy in the field of the protein is calculated as a sum of grid potentials of all ligand atoms corresponding to each ligand pose in the docking cube. A potential in the position of a given ligand atom is obtained by the interpolation of potentials in eight neighbouring grid nodes.

The SOL docking program works in accordance with the genetic algorithm as follows. First, the initial population of individuals (ligand poses in the docking cube) is created by random translations and rotations of a ligand as a rigid body and by a random generation of ligand torsions in their domains. The population size is the main input parameter of SOL, and it is

equal to 30000 by default. Then the total energy, the sum of the ligand grid and stress energies, is calculated for each of individuals and the latter are ranked in respect with their total energies. The individuals (ligand poses) with lowest values of the total energy are selected into the so-called mating pool where they take part in the creation of the population of the next generation through the crossover, mutations and direct translation keeping the population size fixed. Several individuals with the lowest energies are translated to the next generation without any change to be sure that the best ligand poses will not be lost in the next generation. Niching is used to choose diverse individuals in the mating pool. The niching procedure is implemented as a positive energy penalty given to a ligand pose which is close to the ligand pose that has been already chosen into the mating pool. Niching helps to avoid the degeneration of the population when all individuals are collected in one local energy minimum. After a given number of generations (1000 by default) the genetic algorithm stops and the individual (the ligand pose) with the lowest energy is taken from the last generation. This is the solution of the global optimizations problem: finding the ligand pose with the lowest total energy. Several (50 by default) independent runs of the genetic algorithm are performed to get some confidence in the reliability of the global optimization. All 50 ligand poses are clustered in respect with their positions: two ligand poses belong to one cluster if RMSD between their corresponding atoms is less than 1 Å. If population of the first cluster containing the ligand poses with lowest energies is sizeable and the number of clusters is small, the ligand pose with the lowest energy from the first cluster is considered a reliable solution of the global optimization problem. If there are many clusters and there is only one ligand pose in the first cluster, the solution of the global optimization problem is considered as unreliable and optimization procedure should be repeated with higher parameters of the genetic algorithm, mainly with larger population size and with more generations. SOL executes docking of one ligand with default genetic algorithm parameters for several hours at one computing core depending on the number of ligand atoms. First application of SOL in drug design has been made during discovery of new thrombin inhibitors [79]: more than 6000 ligands have been docked by SOL using X-Com grid technology, developed at the Research Computer Center of Moscow State University [25, 81]. Later the MPI multi-processor implementation of SOLGRID and SOL programs [91] make it possible to generate the grid of potentials, as well as to dock one ligand for less than 1 minute using several dozen computing cores. On the other hand, for the virtual screening of large ligand databases (dozens and hundreds of thousands of molecules) SOL can run on the Lomonosov supercomputer [100] distributing ligands over hundreds and thousands computing cores and docking one ligand per one core. Certainly some auxiliary scripts and programs are created to queue up respective jobs and to analyze the docking results. There is a number of success stories of the application of these three docking programs and other ones in computer aided structural based drug design and some of them are presented in [96].

2. Supercomputer Docking Programs

In spite of existence of many docking programs, only a relatively small subset of molecular docking codes have been ported to use many-core high performance architectures, such as GPUs or Intel's Xeon Phi. The employment of supercomputer for docking pursues two main goals: to screen as many ligands as possible during a short period of time, and to dock one ligand as accurately as possible. The increase effectiveness of these multi-processor calculations is also an important technical problem.

2.1. Faster and Larger

Modern chemistry opens its almost unlimited space of small molecular weight ligands for drug design creativity. Millions of on-shelf compounds are now available for different libraries and databases. Much more virtual molecules can be generated, but the possibility of their synthesis is questionable in many cases. All these ligands can be used as a starting pool of candidates for a given, especially new, target protein at the initial stage of the drug development giving a diversity of hit compounds and their optimization results in a diversity of lead compounds. In this section we present several examples of adaptation of existing docking programs to supercomputers for the virtual docking screening of thousands and millions of ligands.

2.2. HSP-DOCK

One of the challenges of supercomputer docking is to create the infrastructure that would allow anyone the ability to dock as many ligands as possible and as quickly as possible. One solution of this problem is presented in [97] where using profiling optimization with the help of a specially designed software “wrapper” called HSP-DOCK authors are able to dock large numbers of ligands efficiently with the widely used DOCK6 program on a large, non-shared memory NICS Kraken supercomputer system: a cluster computer made up of SMP (symmetric multi-processor) nodes [73]. When compared with the standard distribution of MPI multi-threading with DOCK6, HSP-DOCK showed near-linear scaling to the number of cores available to it, without necessitating the modification of the DOCK algorithm code itself. Here are some details of this docking experiment. Four different target proteins were chosen. A library of 1.4 million lead-like molecules from ZINC database was docked against each target. In this case lead-like compounds have the number of torsions ≤ 7 , and molecular weight of less than 350 *Da*. The targeted site of each protein was based on known biological interactions and refined further using MOE’s site finder utility providing precise docking decoys along with propensity for ligand-binding scores. The resulting dataset was analyzed for electrostatics (*ES*), van der Waals (*VDW*) and GRID (the summation of *ES* + *VDW*) scores. Docking is benchmarked on two different computer setups including a small academic cluster and a massively parallel supercomputer: the Medical University of South Carolina Computational Biology Resource Center (MUSC CBRC using MPI-DOCK) cluster of 46 nodes and the National Institute of Computer Science (NICS) Kraken Cray XT5 with 112896 cores (using HSP-DOCK). It is found that the DOCK6 standard MPI version does not scale well on tens of thousands of cores, or when the library of ligands grows into the millions. MPI scaling becomes inefficient because I/O contention and saturation of the master process causes the cluster to spend more time waiting for I/O tasks than carrying out computations. HSP-DOCK dynamically schedules workload distribution to individual nodes to optimize output routines, reducing I/O thrashing and increasing time spent in computation. The MUSC CBRC mean time to dock one ligand per one core was 88s to be compared with 20 s on Kraken. It takes ~ 17 h using 32000 cores of Kraken to dock 50 million flexible ligands into a single target protein. However such large scale docking leads to the need of the posterior analysis of the docking results. Examples presented in this study show that after docking of 1213412 compounds into one target protein there are 57017 hits, i.e. ligands with a score better than a specified threshold. For other investigated targets the number of hits is of the same order: 1 – 5 % of all successfully docked compounds. The necessity of further elaborations of several dozen thousand hits either with more accurate docking program or with bioinformatics

becomes obvious. One may wonder to perform more accurate docking from the beginning, but it needs much more computing resources and another organization of such massive docking. So, the example presented in [97] demonstrates the possibility of fast and effective virtual screening of very large databases of compounds using the DOCK6 docking program without any changes of the code with the help of a software “wrapper” HSP-DOCK on the supercomputer Kraken Cray XT5 of the National Institute of Computer Science.

2.3. BUDE

Another example of a supercomputer multi-core docking program is the Bristol University Docking Engine (BUDE) [52]. The special thrill of BUDE is a high effective realization for a supercomputer performance and compatibility with different types of hardware. It is adapted for modern day accelerators and it has highly optimized computational kernels for the docking and scoring functions using OpenCL, which are capable of sustaining a significant fraction of the hardware's peak performance. Exploiting OpenCL enables performance portability across a wide range of different computer architectures, including CPUs, GPUs and accelerators. BUDE has undergone extensive optimization, and can sustain over 40% of floating point peak performance on a wide range of different architectures. The main features of BUDE is the energy minimization algorithm which is based on the Evolutionary Monte Carlo (EMC) techniques described in [29]. This genetic algorithm based optimization is performed in the space of the six degrees of freedom – the rigid protein and ligand approximation and a tuned empirical free-energy force field is employed for predicting the binding pose and for estimating the protein-ligand binding energy. In the course of a single ligand conformation docking BUDE evaluates the binding energies for hundreds of thousands of ligand poses, and each pose is evaluated independently. The optimized version of BUDE is several times faster than the previous non-optimized version of BUDE on eight different hardware with either GPU or CPU: NVIDIA GTX 680, NVIDIA GTX 780 Ti, NVIDIA Tesla K20c, AMD Radeon HD7970, AMD Radeon R9290X, AMD FirePro S10000, Intel Xeon Phi SE10P and Intel E5 – 2687W (x2). For example the optimizations developed for the GTX 780 Ti have demonstrated the biggest improvement of the Nvidia GPUs with a 4.5× increase over the baseline version of BUDE. The optimized BUDE code achieves 44 billion atom-atom *interactions/s* on the GTX 680, taking 37 s to dock the 128 conformations (~ 0.3 s per conformation). It achieved a sustained performance of 1.43 TFLOPS when measured across the entire BUDE run, representing 46% of the peak single precision performance of the device. The obvious drawback of this work is a too simplified docking model but it shows a direction of possible improvements of performance of any docking program.

2.4. VinaLC

This program is specially designed for the multi-core supercomputer performance with the goal to carry out virtual screening of very large ligand databases by docking [108]. VinaLC is the realization of the AutoDock Vina docking program [98] that is enhanced by a mixed parallel scheme, where within each node multithreading is used, and across different nodes an MPI parallel scheme is applied. VinaLC was developed and tested on petascale supercomputers at Lawrence Livermore National Laboratory. It scales up to 15408 CPU cores demonstrating overheads of less than 4% and it is able to dock 17 million flexible ligands on 15408 cores during 24 hours. The docking accuracy of VinaLC defined by a model implemented in AutoDock Vina

was investigated in [108] using the Directory of Useful Decoys (DUD) dataset [35] and the reassuring results were obtained.

AutoDock Vina was implemented and tested later [38] on four different HPC infrastructures available to academic researchers in the Netherlands: an 8 – *core* virtual machine on the Dutch Academic HPC cloud, a local cluster at the Academic Medical Center of the University of Amsterdam with 128 cores, the Hadoop cluster for scientific research consisting of 90 *data/compute* nodes (adding up to more than 1400 cores) and has a distributed file system with a capacity of 630 *TB*, and the Dutch eScience grid which includes a dozen PBS (portable batch system) clusters all around the Netherlands (>10000 cores). These HPCs were employed to perform virtual screening of about 100000 compounds from ZINC [36] and other libraries investigating parallelism, docking time, etc.

Some details of a more recent use of AutoDock Vina for virtual screening are presented in [20] (docking on the Salomon (Czech Republic) supercomputer) and in [33] (docking on the petaflops-scale Anselm supercomputer <https://www.it4i.cz>). Another example of the adaptation of the docking program from AutoDock family was presented in [21], where AutoDock4.lga.MPI was used to dock 1 million compounds. Details of MPI-realization of the AutoDock4.lga.MPI program were presented in [17], but in [21] particular attention was paid to the optimal organization of docking of millions of ligands on thousands of processors when file preparation and result analysis define the effectiveness of the whole virtual screening job. Among others the clusterization of ligand poses found in independent runs plays an important role in the revelation of best pose and finding best binders among millions of candidate molecules. As a result 1 million ligands having from 0 to 32 torsions (but only 5 % of the database are ligands with at least 10 torsions) have been docked on the Jaguar *Cray XT5* Supercomputer at Oak Ridge National Laboratory using 65 thousand processors. This job took almost 24 hours, but nearly the whole database was screened in 10 hours and only docking of extremely flexible ligands increased the total time.

As predecessor of supercomputer docking, the grid technology should be mentioned. One example of such distributed docking calculations on large scale grids is presented in [37] where two docking programs, AutoDock and FlexX [72] were used for virtual screening millions of ligands against two targets to struggle with malaria and avian influenza.

2.5. More Accurate

Some efforts have been undertaken in attempts to understand main reasons of unsatisfactory reliability of most of docking programs and to improve the docking accuracy. The quick increase of available supercomputer resources gives a fortunate opportunity for such investigations. We present here main features of two docking programs of a new generation and then their use as a tool for the employment of quantum-chemical methods for docking.

2.6. Generalized Docking Programs

2.6.1. FLM

FLM – massive parallel low energy minima search [48, 60, 92]. The MPI-based FLM docking program performs multi-processors local optimizations of the energy of the protein-ligand complex from random initial ligand positions and conformations. As a result the spectrum of all or almost all low energy minima is found. Among these minima the global energy minimum

should present and in respect with the docking paradigm the best ligand pose should correspond to this global minimum. The term “generalized” is used for this program because the result of its performance is not only the global energy minimum but the whole spectrum of low energy minima. Obviously, the good accuracy of ligand positioning is an indispensable prerequisite of high accuracy of the protein–ligand binding energy calculation. FLM works in the frame of the *ab initio* MMFF94 force field [32] without simplifications and a preliminary calculated grid of protein–ligand interaction potentials and without the use of fitting parameters. There are two versions of this docking program: faster *FLM-0.05* where energy is calculated without taking into account the interaction of protein–ligand complex with water solvent, and slower *FLM-0.10* where the PCM implicit solvent model is implemented. The initial ligand positions and conformations in the docking space are generated by random translations and rotations of the ligand as a rigid body and by internal rotations (torsions) of molecular groups around ordinary single bonds which are not included in the cyclic structures, bond lengths and values of valence angles are kept fixed. The local energy optimization is performed by the well-known gradient based L-BFGS method. The optimization is made with variation of Cartesian coordinates of all ligand atoms keeping all protein atoms fixed. The docking space is defined by a sphere of a given radius and the location restricting the position of the geometrical center of the ligand. The number of low energy minima kept in the docking procedure is defined by the input parameter which can be any integer number. The scalability of the FLM performance with the growth of the number of CPU cores is very good up to several thousand cores because each local optimization is performed on a single core. One of peculiarities of the FLM program is the time unlimited performance when the job time and the number of cores are restricted only by a supercomputer work organization. The goal of the FLM performance is not only to find the global energy minimum with high reliability but also to find a given number of low energy minima without skipping any minimum. Exercises with a test set of protein–ligand complexes where FLM was used for docking native ligands into the proteins with which these ligands were crystallized showed that as a rule the global energy minimum was found already after several dozen of thousands of local optimizations, but for the detection of several thousand low energy minima it was needed to perform several hundred thousand, sometimes (for ligands containing about 15 – 20 torsions) more than a million local energy optimizations and in total $\sim 10^9$ energy calculations of the protein–ligand complex were needed for generalized docking of one ligand. It was demonstrated that *FLM-0.05* could find the global energy minimum near the crystallized native ligand position only for a small portion of test complexes. However *FLM-0.10*, where the implicit solvent model was implemented, could find the global energy minimum near the crystallized native ligand pose for more protein–ligand complexes than *FLM-0.05* could do. So, the use of implicit solvent models together with force fields is obviously better for good docking positioning. FLM can be exploited also for finding the low energy minima spectrum of the unbound ligand. This operation is needed for the determination of the global energy minimum of the unbound ligand defining the ligand stress energy which is an important term in the expression for the protein–ligand binding energy. It turned out that FLM could be used for the comparison of different force fields in the docking procedure and for the investigations of effectiveness of different docking algorithms. Special energy minima indexes have been introduced which are convenient for the description of the low energy minima spectrum of protein–ligand complexes [48, 60, 61, 92–94]. In particular, the index of the minimum Near Native (INN) is useful for the indication of the docking accuracy. Its meaning is as follows. All low energy minima

found by a generalized docking program can be ranked by their energy in the ascending order. After this ranking every minimum gets its own integer index which is equal to its number in this ranked list of minima. The lowest energy minimum obtains the index equal to 1. INN indicates the index of the minimum corresponding to the ligand pose differing from the crystallized native ligand position with RMSD (the root mean-square deviation over all ligand atoms) less than 2 \AA . If the docking program finds several minima close to the crystallized native ligand position, INN will be equal to the index of the minimum with the lowest energy (the lowest index from all possible ones). When INN equals to 1, the docking paradigm is satisfied: the found global minimum of the energy of the protein–ligand complex is near the crystallized native ligand pose. FLM is a very resource-intensive docking program and this is its main drawback: it needs about $10000 - 20000 \text{ CPU} * \text{hours}$ for generalized docking of one ligand depending on its size and flexibility.

2.6.2. SOL-P

SOL-P uses the tensor train global optimization algorithm [61, 93, 94]. SOL-P is also the generalized docking program. It performs the search for the global energy minimum, as well as for other low energy minima of the protein–ligand complex in the frame of the MMFF94 force field [32] without any simplifications and without using fitting parameters; it does not use neither the preliminary calculated grid of potentials of ligand probe atom interactions with the target protein, nor premeditated points of ligand positioning in the active site of the protein. The main peculiarity of this program is the novel tensor train (TT) docking algorithm. This algorithm is more effective than the widely used popular genetic algorithm [109]. SOL-P docks flexible ligands into the rigid protein (the early versions of SOL-P have been designated as SOL-T [61]), as well as into the protein with moveable atoms. In the latter case the space for the global energy minimum search is described by the translations and rotations of the ligand as a rigid body, internal torsions of the ligand and by Cartesian coordinates of selected protein atoms. In the case of docking with moveable protein atoms the INON index (Index Near Optimized Native) has been introduced instead of INN [93, 94]. The definition of INON is almost the same as one of INN but the ligand pose corresponding to an energy minimum is compared by RMSD with the optimized native ligand pose obtained by the local energy optimization from the initial crystallized native ligand position. SOL-P for a rigid protein is much faster than FLM: SOL-P needs only about $5 - 100 \text{ CPU} * \text{hours}$ for docking a flexible ligand depending on its size and the number of torsions, but FLM carries out a more thorough search of low energy minima. The effectiveness of finding of the global energy minimum by FLM and SOL-P is almost the same. TT docking algorithm is based on the recently developed methods of the tensor analysis and in particular on the TT global optimization algorithm. Tensors, multi-dimensional arrays, appear in the docking problem as follows. The energy of the protein–ligand complex in the frame of the MMFF94 force field is a continuous function depending on the variables describing all degrees of freedom (d) of the molecular system. If the discretization of each of these variables is carried out by a set of discretization points (n), e.g. equidistant points, the continuous energy function is transformed into d -dimensional tensor with n^d elements. If the discretization is sufficiently fine, the solution of the continuous global optimization problem coincides with the solution of the discrete global optimization problem. The TT global optimization algorithm determines the largest in magnitude element of a tensor. Docking is the global minimization problem but it can be easily converted into the global maximization problem applied for example to the functional:

$$f(x, E_*) = \exp^{100 \operatorname{arccot}(E(x) - E_*)} \quad (2)$$

where $E(x)$ is the dimensionless MMFF94 energy for the given conformation x of the protein–ligand complex, E_* is the global minimum found on the previous iteration. The global maximization problem is solved on the base of TT–Cross method [65] which is used to obtain approximation of the tensor $A(i_1, \dots, i_d)$ in the tensor train (TT) representation:

$$A(i_1, \dots, i_d) \approx \sum_{\alpha_1=1, \dots, \alpha_{d-1}=1}^{r_1, \dots, r_d} G_1(i_1, \alpha_1) G_2(\alpha_1, i_2, \alpha_2) \dots G_{d-1}(\alpha_{d-2}, i_{d-1}, \alpha_{d-1}) G_d(\alpha_{d-1}, i_d). \quad (3)$$

The numbers r_1, \dots, r_{d-1} are called TT–ranks of the tensor. The 3–dimensional tensors $G_i \in \mathbb{R}^{r_{i-1} \times n_i \times r_i}$ are called cores or carriages of the tensor train. Operations on tensors in the TT format are reduced to standard matrix rules. If TT–ranks are reasonably small, then the TT decomposition has several suitable properties allowing a fast tensor element evaluation, effective storage and others [63, 64]. Note that TT–Cross method exploits the well–known matrix cross interpolation [31] algorithm. There are three main parameters defining the TT–docking algorithm performance, and their optimal values are determined during the SOL–P validation [93, 94]: the number of discretization points for each variable corresponding to each degree of freedom, the optimal value is 2^{16} , the maximal TT–rank, the optimal rank is 4, and the number of iterations of the TT–algorithm, the optimal number is 15. SOL–P is written on *C++* with usage of BLAS and LAPACK libraries and it uses the parallel MPI. Details of the TT–docking algorithm and the organization of performance of the SOL–P program are presented in [61, 93, 94].

SOL–P is successfully used for the docking flexible ligands into the proteins with moveable atoms. The results of such docking are described in [93, 94] where results of successful docking with 157 degrees of freedom are presented. The main distinction of this realization from the widely used ensemble docking is that the global energy minimum search is performed with variations of all variables describing ligand and protein degrees of freedom simultaneously and equally. Actually there are no special restrictions on the number of mobile protein atoms, but these atoms can move only in respective cubes with edges of about 1 \AA and centered at the position of these atoms in the structure taken from PDB. Such investigations have shown that movements of some protein atoms in the docking process improves positioning accuracy of docking [93, 94]. The effectiveness of the TT–docking algorithm makes it possible to dock by SOL–P flexible ligands, oligopeptides, with a large number of torsions (more than 15 – 20) [87]. Such flexible ligands cannot be docked usually by classical docking programs.

2.6.3. Quasi–docking

Quasi–docking is a combination of a force field and quantum chemical methods [89]. The validation of the FLM generalized docking program and investigations with it the spectra of low energy minima of different test sets of protein–ligand complexes lead to the idea of the quasi–docking procedure. Quasi–docking has been used for the comparison of quality of docking with different force fields and quantum–chemical methods [88, 89]. Quasi–docking consists of two steps. At the first step sufficiently wide range of low energy minima of a protein–ligand complex is found with the help of the FLM docking program, the energy is being computed in the frame of the MMFF94 force field [32]. At the second step energies of all these minima

are recalculated with other force fields or quantum–chemical methods. It has been shown that when the protein–ligand energy is calculated with the CHARMM force field and the respective GBSW implicit solvent model ligand positioning accuracy is better than in the case of the energy calculated with the MMFF94 force field and with the respective PCM implicit solvent model [89]. On the other hand the docking accuracy with CHARMM is worse than with either PM6–D3H4X [101, 102], or PM7 [83] quantum–chemical methods used for the energy calculations with the COSMO implicit solvent model [43]. These quantum–chemical methods were developed rather recently comparing with older AM1 or PM3 methods, which had been widely used during latest 30 years. PM6–D3H4X and PM7 methods overcome the main limitations of older semiempirical methods describing well dispersion interactions and hydrogen bonds at the level of DFT *ab initio* methods [34]. These quantum–chemical semiempirical methods and the COSMO solvent model are implemented in the MOPAC (<http://OpenMOPAC.net>) package [84] where the module MOZYME gives an opportunity to calculate sufficiently quickly the whole protein–ligand complexes containing thousands of atoms. So, quasi–docking actually gives an opportunity to perform docking with quantum–chemical methods which demonstrate the best docking accuracy. A sufficiently wide low energy minima spectrum should be found at the first step of quasi–docking and its optimal width, when the energy is calculated with MMFF94 without solvent, is determined to perform minimal quantity of quantum–chemical recalculations [48]. Certainly, each step of the quasi–docking procedure essentially exploits supercomputing.

Conclusion

Materials presented in this review show that supercomputers are used for docking more and more widely. To some extent this is amazing because the vast majority of docking programs are still focused on working on laptops and workstations. However, the needs to increase effectiveness of molecular modeling application to drug design, to perform docking more accurately and to screen millions of virtual and on–the–shelf compounds force to attract supercomputer resources for docking. The necessity to use large computing resources for docking appeared more than 10 years ago when grid technologies were used for virtual screening of large libraries of compounds using docking. Currently two directions of supercomputer docking are formed. First, the adaptation of existing docking programs to supercomputers to perform effectively docking of hundreds of thousands or millions of ligands. Two problems should be solved on this road: to reach high effectiveness of multi–core parallel calculations of a great amount of ligands, and to perform a fast and effective analysis of the very large arrays of results yielded by docking. Second direction of the supercomputer docking focuses on a more accurate docking of a single ligand. Along this road novel docking programs are created which are free from simplifications and approximations inherent in classical docking programs, and the novel programs can be used to obtain with maximal accuracy the spectrum of low energy minima of the protein–ligand system, which forms its configuration space and actually defines the free energy of the system. This road is much more tortuous and hard because one should overcome both computing and physical–chemical problems. The outlines of quantum–chemical docking is appearing ahead along this road. However, efforts to achieve this goal will be justified by attaining much higher efficiency in the use of docking programs for the drug development.

Acknowledgments

The work was financially supported by the Russian Science Foundation, Agreement No. 15-11-00025-II. The research is carried out using the equipment of the shared research facilities of HPC computing resources at Lomonosov Moscow State University, including the Lomonosov supercomputer [100].

This paper is distributed under the terms of the Creative Commons Attribution-Non Commercial 3.0 License which permits non-commercial use, reproduction and distribution of the work without further permission provided the original work is properly cited.

References

1. Allen, W.J., Balius, T.E., Mukherjee, S., *et al.*: Dock 6: Impact of new features and current docking performance. *Journal of Computational Chemistry* 36(15), 1132–56 (2015), DOI: 10.1002/jcc.23905
2. Arcon, J.P., Defelipe, L.A., Modenutti, C.P., *et al.*: Molecular dynamics in mixed solvents reveals proteinligand interactions, improves docking, and allows accurate binding free energy predictions. *Journal of Chemical Information and Modeling* 57(4), 846–863 (2017), DOI: 10.1021/acs.jcim.6b00678
3. Basciu, A., Mallocci, G., Pietrucci, F., *et al.*: Holo-like and druggable protein conformations from enhanced sampling of binding pocket volume and shape. *Journal of Chemical Information and Modeling* 59(4), 1515–1528 (2019), DOI: 10.1021/acs.jcim.8b00730
4. Beierlein, F., Lanig, H., Schrer, G., *et al.*: Quantum mechanical/molecular mechanical (QM/MM) docking: An evaluation for known test systems. *Molecular Physics – MOL PHYS* 101, 2469–2480 (2003), DOI: 10.1080/0026897031000092940
5. Bekker, G.J., Araki, M., Oshima, K., *et al.*: Dynamic docking of a medium-sized molecule to its receptor by multicanonical MD simulations. *The Journal of Physical Chemistry B* 123(11), 2479–2490 (2019), DOI: 10.1021/acs.jpcc.8b12419
6. Berman, H.M., Westbrook, J., Feng, Z., *et al.*: The protein data bank. *Nucleic Acids Research* 28(1), 235–42 (2000), DOI: 10.1093/nar/28.1.235
7. Best, R.B., Zhu, X., Shim, J., *et al.*: Optimization of the additive CHARMM all-atom protein force field targeting improved sampling of the backbone phi, psi and side-chain chi(1) and chi(2) dihedral angles. *Journal of Chemical Theory and Computation* 8(9), 3257–3273 (2012), DOI: 10.1021/ct300400x
8. Bikadi, Z., Hazai, E.: Application of the PM6 semi-empirical method to modeling proteins enhances docking accuracy of AutoDock. *Journal of cheminformatics* 1, 15–15 (2009), DOI: 10.1186/1758-2946-1-15
9. Bolcato, G., Cuzzolin, A., Bissaro, M., *et al.*: Can we still trust docking results? An extension of the applicability of DockBench on PDBbind database. *International journal of molecular sciences* 20(14), 3558 (2019), DOI: 10.3390/ijms20143558

10. ten Brink, T., Exner, T.E.: Influence of protonation, tautomeric, and stereoisomeric states on protein–ligand docking results. *Journal of Chemical Information and Modeling* 49(6), 1535–1546 (2009), DOI: 10.1021/ci800420z
11. ten Brink, T., Exner, T.E.: pka based protonation states and microspecies for protein–ligand docking. *Journal of Computer–Aided Molecular Design* 24(11), 935–942 (2010), DOI: 10.1007/s10822-010-9385-x
12. Brozell, S.R., Mukherjee, S., Balius, T.E., *et al.*: Evaluation of dock 6 as a pose generation and database enrichment tool. *Journal of Computer–Aided Molecular Design* 26(6), 749–773 (2012), DOI: 10.1007/s10822-012-9565-y
13. Burley, S.K., Berman, H.M., Bhikadiya, C., *et al.*: RCSB Protein Data Bank: biological macromolecular structures enabling research and education in fundamental biology, biomedicine, biotechnology and energy. *Nucleic Acids Research* 47(D1), D464–D474 (2018), DOI: 10.1093/nar/gky1004
14. Chen, Y.C.: Beware of docking! *Trends in Pharmacological Sciences* 36(2), 78–95 (2015), DOI: 10.1016/j.tips.2014.12.001
15. Chung, J.Y., Hah, J.M., Cho, A.E.: Correlation between performance of QM/MM docking and simple classification of binding sites. *Journal of Chemical Information and Modeling* 49(10), 2382–2387 (2009), DOI: 10.1021/ci900231p
16. Cole, D.J., Tirado-Rives, J., Jorgensen, W.L.: Molecular dynamics and monte carlo simulations for protein–ligand binding and inhibitor design. *Biochimica et biophysica acta* 1850(5), 966–971 (2015), DOI: 10.1016/j.bbagen.2014.08.018
17. Collignon, B., Schulz, R., Smith, J.C., *et al.*: Task–parallel message passing interface implementation of Autodock4 for docking of very large databases of compounds using high–performance super–computers. *Journal of Computational Chemistry* 32(6), 1202–9 (2011), DOI: 10.1002/jcc.21696
18. De Vivo, M., Cavalli, A.: Recent advances in dynamic docking for drug discovery. *Wiley Interdisciplinary Reviews: Computational Molecular Science* 7(6), e1320 (2017), DOI: 10.1002/wcms.1320
19. Dixit, S.B., Chipot, C.: Can absolute free energies of association be estimated from molecular mechanical simulations? The biotin–streptavidin system revisited. *The Journal of Physical Chemistry A* 105(42), 9795–9799 (2001), DOI: 10.1021/jp011878v
20. Dolezal, R., Nepovimova, E., Melikova, M., *et al.*: Structure–based virtual screening for novel modulators of human orexin 2 receptor with cloud systems and supercomputers. In: *Advanced Topics in Intelligent Information and Database Systems*, pp. 161–171. Springer International Publishing (2017), DOI: 10.1007/978-3-319-56660-3_15
21. Ellingson, S.R., Dakshanamurthy, S., Brown, M., *et al.*: Accelerating Virtual High–Throughput Ligand Docking: current technology and case study on a petascale super–computer. *Concurrency and computation : practice & experience* 26(6), 1268–1277 (2014), DOI: 10.1002/cpe.3070

22. Elokely, K.M., Doerksen, R.J.: Docking challenge: protein sampling and molecular docking performance. *Journal of chemical information and modeling* 53(8), 1934–1945 (2013), DOI: 10.1021/ci400040d
23. Evangelista Falcon, W., Ellingson, S.R., Smith, J.C., *et al.*: Ensemble docking in drug discovery: How many protein configurations from molecular dynamics simulations are needed to reproduce known ligand binding? *The Journal of Physical Chemistry B* 123(25), 5189–5195 (2019), DOI: 10.1021/acs.jpcc.8b11491
24. Feixas, F., Lindert, S., Sinko, W., *et al.*: Exploring the role of receptor flexibility in structure-based drug discovery. *Biophysical chemistry* 186, 31–45 (2014), DOI: 10.1016/j.bpc.2013.10.007
25. Filamofitsky, M.P.: The system X-Com for metacomputing support: architecture and technology. *Numerical Methods and Programming* 5(2), 1–9 (in Russian) (2004)
26. Forli, S., Huey, R., Pique, M.E., *et al.*: Computational protein–ligand docking and virtual drug screening with the AutoDock suite. *Nature protocols* 11(5), 905–19 (2016), DOI: 10.1038/nprot.2016.051
27. Friesner, R.A., Murphy, R.B., Repasky, M.P., *et al.*: Extra precision glide: docking and scoring incorporating a model of hydrophobic enclosure for protein–ligand complexes. *Journal of Medicinal Chemistry* 49(21), 6177–96 (2006), DOI: 10.1021/jm051256o
28. Genheden, S., Nilsson, I., Ryde, U.: Binding affinities of factor Xa inhibitors estimated by thermodynamic integration and MM/GBSA. *Journal of Chemical Information and Modeling* 51(4), 947–958 (2011), DOI: 10.1021/ci100458f
29. Gibbs, N., Clarke, A.R., Sessions, R.B.: Ab initio protein structure prediction using physicochemical potentials and a simplified off-lattice model. *Proteins: Structure, Function, and Bioinformatics* 43(2), 186–202 (2001), DOI: 10.1002/1097-0134(20010501)43:2%3C186::aid-prot1030%3E3.0.co;2-1
30. Gioia, D., Bertazzo, M., Recanatini, M., *et al.*: Dynamic docking: A paradigm shift in computational drug discovery. *Molecules (Basel, Switzerland)* 22(11), 2029 (2017), DOI: 10.3390/molecules22112029
31. Goreinov, S., Tyrtysnikov, E.: The maximal–volume concept in approximation by low-rank matrices. *Contemporary Mathematics* 268, 47–51 (2001)
32. Halgren, T.A.: Merck molecular force field. I. Basis, form, scope, parameterization, and performance of MMFF94. *Journal of Computational Chemistry* 17(5-6), 490–519 (1996), DOI: 10.1002/(SICI)1096-987X(199604)17:5/6%3C490::AID-JCC1%3E3.0.CO;2-P
33. Honegr, J., Dolezal, R., Malinak, D., *et al.*: Rational design of a new class of toll-like receptor 4 (TLR4) tryptamine related agonists by means of the structure– and ligand–based virtual screening for vaccine adjuvant discovery. *Molecules* 23(1) 102 (2018), DOI: 10.3390/molecules23010102
34. Hostaš, J., Řezáč, J., Hobza, P.: On the performance of the semiempirical quantum mechanical PM6 and PM7 methods for noncovalent interactions. *Chemical Physics Letters* 568-569(Supplement C), 161–166 (2013), DOI: 10.1016/j.cplett.2013.02.069

35. Huang, N., Shoichet, B.K., Irwin, J.J.: Benchmarking sets for molecular docking. *Journal of Medicinal Chemistry* 49(23), 6789–6801 (2006), DOI: 10.1021/jm0608356
36. Irwin, J.J., Sterling, T., Mysinger, M.M., *et al.*: ZINC: A free tool to discover chemistry for biology. *Journal of Chemical Information and Modeling* 52(7), 1757–1768 (2012), DOI: 10.1021/ci3001277
37. Jacq, N., Breton, V., Chen, H.Y., *et al.*: Virtual screening on large scale grids. *Parallel Computing* 33(4), 289–301 (2007), DOI: 10.1016/j.parco.2007.02.010
38. Jaghoori, M.M., Bleijlevens, B., Olabarriaga, S.D.: 1001 Ways to run AutoDock Vina for virtual screening. *Journal of Computer–Aided Molecular Design* 30(3), 237–249 (2016), DOI: 10.1007/s10822-016-9900-9
39. Jorgensen, W.L., Tirado-Rives, J.: Potential energy functions for atomic–level simulations of water and organic and biomolecular systems. *Proceedings of the National Academy of Sciences of the United States of America* 102(19), 6665–70 (2005), DOI: 10.1073/pnas.0408037102
40. Kalliokoski, T., Salo, H.S., Lahtela-Kakkonen, M., *et al.*: The effect of ligand–based tautomer and protomer prediction on structure–based virtual screening. *Journal of Chemical Information and Modeling* 49(12), 2742–2748 (2009), DOI: 10.1021/ci900364w
41. Kantardjiev, A.A.: Quantum.ligand.dock: protein–ligand docking with quantum entanglement refinement on a GPU system. *Nucleic acids research* 40(Web Server issue), W415–W422 (2012), DOI: 10.1093/nar/gks515
42. Khavrutskii, I.V., Wallqvist, A.: Improved binding free energy predictions from single–reference thermodynamic integration augmented with hamiltonian replica exchange. *Journal of Chemical Theory and Computation* 7(9), 3001–3011 (2011), DOI: 10.1021/ct2003786
43. Klamt, A., Schuurmann, G.: COSMO: a new approach to dielectric screening in solvents with explicit expressions for the screening energy and its gradient. *Journal of the Chemical Society, Perkin Transactions 2* (5), 799–805 (1993), DOI: 10.1039/P29930000799
44. Klebe, G.: The use of thermodynamic and kinetic data in drug discovery: Decisive insight or increasing the puzzlement? *ChemMedChem* 10(2), 229–231 (2015), DOI: 10.1002/cmdc.201402521
45. Klimovich, P.V., Shirts, M.R., Mobley, D.L.: Guidelines for the analysis of free energy calculations. *Journal of computer–aided molecular design* 29(5), 397–411 (2015), DOI: 10.1007/s10822-015-9840-9
46. Kollman, P.: Free energy calculations: Applications to chemical and biochemical phenomena. *Chemical Reviews* 93(7), 2395–2417 (1993), DOI: 10.1021/cr00023a004
47. Kutov, D.K., Katkova, E.V., Sulimov, E.V., *et al.*: Influence of the method of hydrogen atoms incorporation into the target protein on the protein–ligand binding energy. *Bulletin of the South Ural State University, Ser. Mathematical Modelling, Programming & Computer Software* 10(3), 94–107 (2017), DOI: 10.14529/mmp170308

48. Kutov, D.C., Sulimov, A.V., Sulimov, V.B.: Supercomputer docking: Investigation of low energy minima of protein–ligand complexes. *Supercomputing Frontiers and Innovations* 5(3), 134–137 (2018), DOI: 10.14529/jsfi180326
49. Lamim Ribeiro, J.a.M., Tiwary, P.: Toward achieving efficient and accurate ligand–protein unbinding with deep learning and molecular dynamics through RAVE. *Journal of Chemical Theory and Computation* 15(1), 708–719 (2019), DOI: 10.1021/acs.jctc.8b00869
50. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. *Nature* 521, 436 (2015), DOI: 10.1038/nature14539
51. Lushchekina, S.V., Makhaeva, G.F., Novichkova, D.A., *et al.*: Supercomputer modeling of dual–site acetylcholinesterase (AChE) inhibition. *Supercomputing Frontiers and Innovations* 5(4), 89–97 (2018), DOI: 10.14529/jsfi180410
52. McIntosh-Smith, S., Price, J., Sessions, R.B., *et al.*: High performance in silico virtual drug screening on many–core processors. *The International Journal of High Performance Computing Applications* 29(2), 119–134 (2015), DOI: 10.1177/1094342014528252
53. Michel, J., Foloppe, N., Essex, J.W.: Rigorous free energy calculations in structure–based drug design. *Molecular Informatics* 29(8-9), 570–578 (2010), DOI: 10.1002/minf.201000051
54. Mobley, D.L., Gilson, M.K.: Predicting binding free energies: Frontiers and benchmarks. *Annual Review of Biophysics* 46(1), 531–558 (2017), DOI: 10.1146/annurev-biophys-070816-033654
55. Morris, G.M., Huey, R., Lindstrom, W., *et al.*: Autodock4 and AutoDockTools4: Automated docking with selective receptor flexibility. *Journal of Computational Chemistry* 30(16), 2785–91 (2009), DOI: 10.1002/jcc.21256
56. Morris, G.M., Goodsell, D.S., Halliday, R.S., *et al.*: Automated docking using a Lamarckian genetic algorithm and an empirical binding free energy function. *Journal of Computational Chemistry* 19(14), 1639–1662 (1998), DOI: 10.1002/(SICI)1096-987X(19981115)19:14<1639::AID-JCC10>3.0.CO;2-B
57. Neves, M.A., Totrov, M., Abagyan, R.: Docking and scoring with ICM: the benchmarking results and strategies for improvement. *Journal of Computer–Aided Molecular Design* 26(6), 675–686 (2012), DOI: 10.1007/s10822-012-9547-0
58. Nguyen, D.D., Wei, G.W.: Agl–score: Algebraic graph learning score for proteinligand binding scoring, ranking, docking, and screening. *Journal of Chemical Information and Modeling* 59(7), 3291–3304 (2019), DOI: 10.1021/acs.jcim.9b00334
59. Nogueira, M.S., Koch, O.: The development of target–specific machine learning models as scoring functions for docking–based target prediction. *Journal of Chemical Information and Modeling* 59(3), 1238–1252 (2019), DOI: 10.1021/acs.jcim.8b00773
60. Oferkin, I.V., Katkova, E.V., Sulimov, A.V., *et al.*: Evaluation of docking target functions by the comprehensive investigation of protein–ligand energy minima. *Advances in Bioinformatics* 2015, 126858 (2015), DOI: 10.1155/2015/126858

61. Oferkin, I.V., Zheltkov, D.A., Tyrtysnikov, E.E., *et al.*: Evaluation of the docking algorithm based on Tensor Train global optimization. *Bulletin of the South Ural State University, Ser. Mathematical Modelling, Programming & Computer Software* 8(4), 83–99 (2015), DOI: 10.14529/mmp150407
62. Ohue, M., Shimoda, T., Suzuki, S., *et al.*: Megadock 4.0: an ultra-high-performance protein-protein docking software for heterogeneous supercomputers. *Bioinformatics (Oxford, England)* 30(22), 3281–3283 (2014), DOI: 10.1093/bioinformatics/btu532
63. Oseledets, I.: Tensor-train decomposition. *SIAM Journal on Scientific Computing* 33(5), 2295–2317 (2011), DOI: 10.1137/090752286
64. Oseledets, I., Tyrtysnikov, E.: Breaking the curse of dimensionality, or how to use SVD in many dimensions. *SIAM Journal on Scientific Computing* 31(5), 3744–3759 (2009), DOI: 10.1137/090748330
65. Oseledets, I., Tyrtysnikov, E.: TT-cross approximation for multidimensional arrays. *Linear Algebra and its Applications* 432(1), 70–88 (2010), DOI: 10.1016/j.laa.2009.07.024
66. Pagadala, N.S., Syed, K., Tuszynski, J.: Software for molecular docking: a review. *Bio-physical Reviews* 9(2), 91–102 (2017), DOI: 10.1007/s12551-016-0247-1
67. Pan, A.C., Borhani, D.W., Dror, R.O., *et al.*: Molecular determinants of drugreceptor binding kinetics. *Drug Discovery Today* 18(13), 667–673 (2013), DOI: 10.1016/j.drudis.2013.02.007
68. Park, M.S., Gao, C., Stern, H.A.: Estimating binding affinities by docking/scoring methods using variable protonation states. *Proteins: Structure, Function, and Bioinformatics* 79(1), 304–314 (2011), DOI: 10.1002/prot.22883
69. Pei, J., Zheng, Z., Kim, H., *et al.*: Random forest refinement of pairwise potentials for proteinligand decoy detection. *Journal of Chemical Information and Modeling* 59(7), 3305–3315 (2019), DOI: 10.1021/acs.jcim.9b00356
70. Perthold, J.W., Oostenbrink, C.: Accelerated enveloping distribution sampling: Enabling sampling of multiple end states while preserving local energy minima. *The Journal of Physical Chemistry B* 122(19), 5030–5037 (2018), DOI: 10.1021/acs.jpcc.8b02725
71. Ragoza, M., Hochuli, J., Idrobo, E., *et al.*: Proteinligand scoring with convolutional neural networks. *Journal of Chemical Information and Modeling* 57(4), 942–957 (2017), DOI: 10.1021/acs.jcim.6b00740
72. Rarey, M., Kramer, B., Lengauer, T., *et al.*: A fast flexible docking method using an incremental construction algorithm. *Journal of Molecular Biology* 261(3), 470–489 (1996), DOI: 10.1006/jmbi.1996.0477
73. Rekapalli, B., Vose, A., Giblock, P.: HSPp-BLAST: Highly scalable parallel PSI-BLAST for very large-scale sequence searches. In: 4th International Conference on Bioinformatics and Computational Biology 2012, BICoB 2012, 12–14 March 2012, Las Vegas, Nevada, USA. pp. 37–42 (03 2012)

74. Ribeiro, J.M.L., Bravo, P., Wang, Y., *et al.*: Reweighted autoencoded variational Bayes for enhanced sampling (RAVE). *The Journal of Chemical Physics* 149(7), 072301 (2018), DOI: 10.1063/1.5025487
75. Riniker, S., Christ, C.D., Hansen, N., *et al.*: Comparison of enveloping distribution sampling and thermodynamic integration to calculate binding free energies of phenylethanolamine N-methyltransferase inhibitors. *The Journal of Chemical Physics* 135(2), 024105 (2011), DOI: 10.1063/1.3604534
76. Romanov, A.N., Jabin, S.N., Martynov, Y.B., *et al.*: Surface generalized born method: A simple, fast, and precise implicit solvent model beyond the coulomb approximation. *The Journal of Physical Chemistry. A, Molecules, spectroscopy, kinetics, environment & general theory* 108(43), 9323–9327 (2004), DOI: 10.1021/jp046721s
77. Ryde, U., Sderhjem, P.: Ligand–binding affinity estimates supported by quantum–mechanical methods. *Chemical Reviews* 116(9), 5520–5566 (2016), DOI: 10.1021/acs.chemrev.5b00630
78. Sadovnichii, V.A., Sulimov, V.B.: Supercomputing technologies in medicine. In: *Supercomputing Technologies in Science*, pp. 16–23. Moscow University Publishing (2009)
79. Sinauridze, E.I., Romanov, A.N., Gribkova, I.V., *et al.*: New synthetic thrombin inhibitors: molecular design and experimental verification. *PLoS One* 6(5), e19969 (2011), DOI: 10.1371/journal.pone.0019969
80. Sliwoski, G., Kothiwale, S., Meiler, J., *et al.*: Computational methods in drug discovery. *Pharmacological Reviews* 66(1), 334–395 (2013), DOI: 10.1124/pr.112.007336
81. Sobolev, S.I.: Effective performance of distributed computing environments, pp. 249–258 (in Russian). Moscow State University Press (2008)
82. Spitaleri, A., Decherchi, S., Cavalli, A., *et al.*: Fast dynamic docking guided by adaptive electrostatic bias: The MD–binding approach. *Journal of Chemical Theory and Computation* 14(3), 1727–1736 (2018), DOI: 10.1021/acs.jctc.7b01088
83. Stewart, J.J.: Optimization of parameters for semiempirical methods VI: more modifications to the NDDO approximations and re–optimization of parameters. *Journal of Molecular Modeling* 19(1), 1–32 (2013), DOI: 10.1007/s00894-012-1667-x
84. Stewart, J.J.P.: Mopac2016. <http://OpenMOPAC.net> (2016)
85. Still, W.C., Tempczyk, A., Hawley, R.C., *et al.*: Semianalytical treatment of solvation for molecular mechanics and dynamics. *Journal of the American Chemical Society* 112(16), 6127–6129 (1990), DOI: 10.1021/ja00172a038
86. Strecker, C., Meyer, B.: Plasticity of the binding site of renin: Optimized selection of protein structures for ensemble docking. *Journal of Chemical Information and Modeling* 58(5), 1121–1131 (2018), DOI: 10.1021/acs.jcim.8b00010

87. Sulimov, A., Kutov, D., Ilin, I., *et al.*: Supercomputer docking with a large number of degrees of freedom. In: Devillers, E.J., Geronikaki, A. (eds.) 10th International Symposium on Computational Methods in Toxicology and Pharmacology Integrating Internet Resources 2019, CMPTI 2019, 23-27 June 2019, Ioannina, Greece. p. 24 (2019)
88. Sulimov, A.V., Kutov, D.C., Katkova, E.V., *et al.*: Combined docking with classical force field and quantum chemical semiempirical method PM7. *Advances in Bioinformatics* 2017, 7167691 (2017), DOI: 10.1155/2017/7167691
89. Sulimov, A.V., Kutov, D.C., Katkova, E.V., *et al.*: New generation of docking programs: Supercomputer validation of force fields and quantum-chemical methods for docking. *Journal of Molecular Graphics and Modelling* 78, 139–147 (2017), DOI: 10.1016/j.jmkgm.2017.10.007
90. Sulimov, A.V., Kutov, D.C., Katkova, E.V., *et al.*: Search for approaches to improving the calculation accuracy of the proteinligand binding energy by docking. *Russian Chemical Bulletin, International Edition* 66(10), 1913–1924 (2017), DOI: 10.1007/s11172-017-1966-6
91. Sulimov, A.V., Kutov, D.C., Oferkin, I.V., *et al.*: Application of the docking program SOL for CSAR benchmark. *Journal of Chemical Information and Modeling* 53(8), 1946–56 (2013), DOI: 10.1021/ci400094h
92. Sulimov, A.V., Kutov, D.C., Sulimov, V.B.: Parallel supercomputer docking program of the new generation: Finding low energy minima spectrum. In: 4th Russian Supercomputing Days 2018, RuSCDays 2018, 2425 September 2018, Moscow, Russia. vol. 965, pp. 314–330 (2018), DOI: 10.1007/978-3-030-05807-4_27
93. Sulimov, A.V., Zheltkov, D.A., Oferkin, I.V., *et al.*: Evaluation of the novel algorithm of flexible ligand docking with moveable target-protein atoms. *Computational and Structural Biotechnology Journal* 15, 275–285 (2017), DOI: 10.1016/j.csbj.2017.02.004
94. Sulimov, A.V., Zheltkov, D.A., Oferkin, I.V., *et al.*: Tensor train global optimization: Application to docking in the configuration space with a large number of dimensions. In: 3rd Russian Supercomputing Days. vol. 793, pp. 151–167 (2017), DOI: 10.1007/978-3-319-71255-0_12
95. Sulimov, V.B., Sulimov, A.: Docking: molecular modeling for drug discovery. AINTELL, Moscow (2017)
96. Sulimov, V.B., Kutov, D.C., Sulimov, A.V.: Advances in docking. *Current Medicinal Chemistry* 26(37), 1–25 (2019), DOI: 10.2174/0929867325666180904115000
97. Trager, R.E., Giblock, P., Soltani, S., *et al.*: Docking optimization, variance and promiscuity for large-scale drug-like chemical space using high performance computing architectures. *Drug Discovery Today* 21(10), 1672–1680 (2016), DOI: 10.1016/j.drudis.2016.06.023
98. Trott, O., Olson, A.J.: Autodock Vina: improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading. *Journal of Computational Chemistry* 31(2), 455–61 (2010), DOI: 10.1002/jcc.21334

99. Ustach, V.D., Lakkaraju, S.K., Jo, S., *et al.*: Optimization and evaluation of site-identification by ligand competitive saturation (SILCS) as a tool for target-based ligand optimization. *Journal of Chemical Information and Modeling* 59(6), 3018–3035 (2019), DOI: 10.1021/acs.jcim.9b00210
100. Voevodin, V.V., Antonov, A.S., Nikitenko, D.A., *et al.*: Supercomputer lomonosov-2: Large scale, deep monitoring and fine analytics for the user community. *Supercomputing Frontiers and Innovations* 6(2), 4–11 (2019), DOI: 10.14529/jsfi190201
101. Řezáč, J., Hobza, P.: A halogen-bonding correction for the semiempirical PM6 method. *Chemical Physics Letters* 506(4), 286–289 (2011), DOI: 10.1016/j.cplett.2011.03.009
102. Řezáč, J., Hobza, P.: Advanced corrections of hydrogen bonding and dispersion for semiempirical quantum mechanical methods. *Journal of Chemical Theory and Computation* 8(1), 141–151 (2012), DOI: 10.1021/ct200751e
103. Wang, J.C., Lin, J.H., Chen, C.M., *et al.*: Robust scoring functions for protein-ligand interactions with quantum chemical charge models. *Journal of chemical information and modeling* 51(10), 2528–2537 (2011), DOI: 10.1021/ci200220v
104. Wang, J., Wolf, R.M., Caldwell, J.W., *et al.*: Development and testing of a general amber force field. *Journal of Computational Chemistry* 25(9), 1157–1174 (2004), DOI: 10.1002/jcc.20035
105. Wang, L., Wu, Y., Deng, Y., *et al.*: Accurate and reliable prediction of relative ligand binding potency in prospective drug discovery by way of a modern free-energy calculation protocol and force field. *Journal of the American Chemical Society* 137(7), 2695–2703 (2015), DOI: 10.1021/ja512751q
106. Xie, B., Clark, J.D., Minh, D.D.L.: Efficiency of stratification for ensemble docking using reduced ensembles. *Journal of Chemical Information and Modeling* 58(9), 1915–1925 (2018), DOI: 10.1021/acs.jcim.8b00314
107. Yuriev, E., Holien, J., Ramsland, P.A.: Improvements, trends, and new ideas in molecular docking: 20122013 in review. *Journal of Molecular Recognition* 28(10), 581604 (2015), DOI: 10.1002/jmr.2471
108. Zhang, X., Wong, S.E., Lightstone, F.C.: 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), DOI: 10.1002/jcc.23214
109. Zheltkov, D.A., Oferkin, I.V., Katkova, E.V., *et al.*: TTDock: a docking method based on tensor train decompositions. *Numerical Methods and Programming* 14(3), 279–291 (in Russian) (2013)
110. van Zundert, G., Rodrigues, J., Trellet, M., *et al.*: The HADDOCK2.2 Web Server: User-friendly integrative modeling of biomolecular complexes. *Journal of Molecular Biology* 428(4), 720–725 (2016), DOI: 10.1016/j.jmb.2015.09.014