

Computational Graphics Software for Interactive Docking and Visualization of Ligand–Protein Complementarity

Saravana G. Baskaran, Thayne P. Sharp, and Kim A. Sharp*

Cite This: <https://dx.doi.org/10.1021/acs.jcim.0c01485>

Read Online

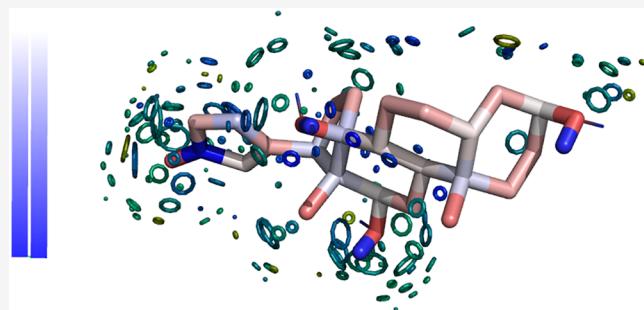
ACCESS |

Metrics & More

Article Recommendations

Supporting Information

ABSTRACT: The Dockeye software is designed to complement automated docking protocols by allowing the user's chemical know-how and experience of what makes for good protein–ligand binding, knowledge that is not easily encoded into automated algorithms, to guide the docking. It allows the interactive manipulation of the ligand placement against a protein target. Real-time intuitively comprehensible feedback about the location, spatial density, and the extent of both favorable and unfavorable atomic interactions between ligand and protein is provided through a carefully designed graphical object. It is also a tool for the graphical analysis of the interactions of known protein–ligand complexes. Comparative docking of 58 protein–ligand complexes with Dockeye and Autodock Vina shows how this software can be used synergistically with automated docking programs to significantly improve the task of discovery of ligand placement.



INTRODUCTION

Docking of a ligand or drug candidate against a protein target is a central part of the computer aided drug development (CADD) pipeline. Docking may be used to discover the placement of the ligand in the binding site when the latter is known, the binding site if this is unknown, and lead compound screening.¹ The term placement refers to the positions of all the atoms of the ligand with respect to the protein target. These positions can be defined in terms of the Cartesian coordinates of all the ligand atoms in the reference frame of the protein. An alternative and useful way to describe the placement of the ligand is in terms of the position and orientation of the ligand with respect to its geometric center and the ligand's internal coordinates, particularly the dihedral angles around the rotatable bonds of the flexible ligands. The first application, placement discovery, is probably the most tractable of the three applications of docking. The other two applications rely on the success of the first but involve extra challenges. Binding site discovery presupposes that one can correctly discover the best placement for each candidate binding site and correctly rank them against each other. Lead compound screening presupposes that one can correctly discover the best placement for each candidate ligand and correctly rank them against each other. The extra challenge of screening is that one must compare ligands with different structures, chemical properties, and conformational repertoires with each other. Broadly speaking, successful docking relies on two factors:² (1) a good scoring function (A scoring function, whether physics based, knowledge based, or empirical, should provide a reliable quantitative ranking of alternative ligand

placements.); (2) a sufficiently thorough search of the coordinate phase space describing the ligand placement.

The ultimate scoring function is of course the binding free energy, ΔG_{bind} . However, while significant progress has been made in the calculation of ΔG_{bind} ,^{3–12} particularly by using all-atom methods with explicit or implicit solvent models, it is still extremely challenging: The reproduction of the experimental ΔG_{bind} given a high resolution structure of the protein–ligand complex takes orders of magnitude longer than could be afforded in automated docking or interactive docking, and even then, the results are often off by 1–2 kcal/mol or more (i.e., an order of magnitude or more in affinity). Practical docking requires much simpler scoring functions that still retain as much as possible of the important physics governing protein–ligand affinity. Complementarity between the ligand in its placement and the protein binding site is a large determinant of ligand affinity. Shape complementarity is an important element of this. In one of the earliest and most influential docking algorithms, DOCK, the representation of the “inverse” shape of the binding site was a key advance.^{13,14} A variety of approaches have been used to model shape complementarity in docking,^{15–21} and shape complementarity, or lack thereof, is an

Received: December 24, 2020

important filter in virtual screening. Beyond shape complementarity, there must be, more broadly, complementarity of specific physical chemical interactions. This includes (1) direct complementarity of partners for specific bonding such as H-bonds, ionic bonds, salt bridges, and electrostatic and polar complementarity, which may be grouped under the concept of electrostatic complementarity; (2) indirect complementarity involving solvent effects. The solvent contributions to complementarity are indirect in the sense that the energies come principally from ligand–solvent and protein–solvent interactions and *not* from direct ligand–protein interactions. The most well-known of these is the hydrophobic effect. Complementarity involves matching up apolar surfaces in the ligand–protein interface. The “negative design” aspect of not putting a polar or charged group of either the protein or the ligand against an apolar region of the opposing surface is just as important. This imposes a penalty for the loss of hydration of the polar group without any compensating interaction. For example, it would be unfavorable if a ligand H-bond donor group was placed in a hydrophobic pocket because of the loss of H-bonding interactions of this group with water in the unbound state. Since this effect comes from a loss of favorable interactions in the unbound state, not from an unfavorable one in the bound state, it is difficult to incorporate into docking programs or even to represent it visually in a graphical depiction of the bound state. There are a variety of ways to represent these two types of complementarity in docking, either graphically or in terms of implicit solvent energy functions.^{22,19,23–27} This is an active area of research but not the main focus of this work since the software described here is rather agnostic as to what scoring function is used.

For the search component of docking, the ligand’s coordinate phase space is usefully partitioned into three translational degrees of freedom (T) and three rotational degrees of freedom (R) around the geometric center of the molecule and the internal conformational degrees of freedom (I). In this work, we only consider docking to a single protein conformer, so the ligand terms comprise all the degrees of freedom for the search. Docking to rigid targets is still almost universally the standard practice, given current computational capabilities. If, as is common, fixed bond length and angle geometries for the ligand are assumed, then ligand flexibility contributes up to M^N internal conformations, where N is the number of torsional degrees of freedom (rotatable bonds) and M is the number of increments into which the 360° range for the torsion angle is divided. Clearly, with an increase in the number of rotatable bonds, the internal conformational search space can rapidly overtake the six rotational/translational (R/T) degrees of freedom of the placement. On the other hand, for larger ligands, one must search more finely in orientational space because, for a given change in orientation angle, an atom further from the center of the ligand will be displaced a larger distance in space, or put another way, positioning all the atoms of a larger ligand to within some given distance of their true positions requires greater precision in the orientation angle. The search process is often fully automated, and a variety of software packages and algorithms have been developed.^{13,14,28–34} For a recent review, see Pagadala et al.³⁵ Generally, these searches are controlled by initial parameter values, which set, for example, the fineness and exhaustiveness of the search, but there is little to no user-intervention during the search itself.

Automated docking has higher throughput than manual docking, but the results are at the mercy of the accuracy of the scoring function. False negatives, good placements that are erroneously excluded because of imperfect scoring functions, can be missed in automated docking without one even knowing that they were there. Moreover, it is rarely useful for these docking programs to return a single, best-scoring placement. Instead, a short list of the top placements ranked by that program’s scoring function is more useful. These can then be compared against each other using some orthogonal method, the knowledge base of the experienced user, or some other contextual information. An obvious orthogonal method involves molecular graphics programs. Molecular graphics provides a natural tool for user guiding docking. This can range from simply viewing and vetting the results from automated docking to fine-tuning the results of automated docking to fully interactive docking. While this obviously has a lower throughput and the search of the coordinate phase space is more limited, it has some benefits. In principle, molecular graphics enabled interactive docking can leverage the brain’s pattern recognition abilities and the expertise of the user to “see past” the limitations of imperfect docking scoring functions to placements and ligand conformations that are more plausible and make better chemical sense.³⁶ However, this depends on having suitable molecular graphics tools for docking: There must be a lag free, ergonomically efficient way to generate placements, including different ligand conformations. To maximize the limited search of the coordinate phase space, one does not want a user interface that constantly takes the user’s eyes off the screen or requires multiple key/mouse actions to switch between protein, ligand, and global manipulations. In addition, there must be instantaneous, comprehensible, and intuitive feedback about the quality of every placement as it is generated. There have been some very interesting haptic systems applied to docking, including force feedback gloves, joysticks, styli, and track-ball resistance methods.^{36–40} In the experience of one author (K.A.S.), these can provide useful information about the quality of the current placement, but the generation of cues as to how to move the ligand to improve the placement is difficult.

Regarding the crucial role of complementarity in determining a good placement, discussed above, it is a nontrivial task to represent this graphically. Representations of shape and chemical complementarity each have their challenges. Consider the standard choices for molecular graphics representations: ribbon, cartoon, or chain traces are principally used to show the overall fold and topography of the protein and are hardly applicable to the display of complementarity. Stick or line representations do not adequately convey the 3-dimensionality of the interactions nor the quality of the fit at the interface. They can also be very cluttered. Finally, space filling or surface representations obscure the display of the binding interface as much as they show it. There is clearly a need for further development in the use of molecular graphics for quantitative protein–ligand docking applications.

The goal of this paper is to describe software designed specifically for visualization of ligand–protein complementarity and to enable interactive docking. The software is designed to allow the interactive manipulation of a ligand placement against a protein target, while providing real-time feedback about the location, spatial density, and extent of both favorable and unfavorable atomic interactions between ligand and protein; a readout of the total interaction energy and its

subcomponent energy terms; automatic selection of the most favorable ligand conformation for each placement; automatic logging of minimum energy placements for later recall/analysis. It is designed to complement automated docking protocols in two ways. Real-time interactive feedback enables the user's chemical know-how and previous experience of what makes for good protein–ligand binding, knowledge that is not easily encoded into an automated algorithm, to guide the docking. It attempts to present as much information about the interactions, overall and at the atomic level in graphical form, so as to capitalize on the pattern-recognition powers of the brain. The software can also be used in a nondocking mode and provides a tool for the graphical analysis of the interactions of a known protein–ligand complex with the goal of aiding the rational design and improvement of the ligand.

METHODS AND ALGORITHMS

Dockeye Software. The Dockeye software is designed to allow the interactive posing of a ligand against a protein target, while providing real-time feedback about the location, spatial density, and extent of both favorable and unfavorable atomic interactions between ligand and protein; a readout of the total interaction energy and its subcomponent energy terms; automatic selection of the most favorable ligand conformation for each placement; automatic logging of minimum energy placements for later recall/analysis and placement bookmarking. The code, utility programs, examples, and instructions are available from GitHub at https://github.com/kimandsharp/dockeye_multi.

Graphical Driver Front End. A front end module reads in the initial protein and ligand coordinates from an ATM format file.^{41,42} The ATM format file contains the coordinate information from a PDB format file in the same format but has the occupancy and B-factor fields replaced by the atomic radii and charges, respectively. This allows all the atom-specific information to be contained in a single file. Multiple ligand conformations may be contained within one ligand input file. For any given placement, the front end calculates the current coordinates of the protein and all conformations of the ligand, using the graphical object transformation matrices corresponding to the current display state of the molecules in the host graphical rendering program. Coordinates, radii, and charges are then passed to the back-end module. The back-end module calculates the interaction between the ligand and the protein. It returns the identity of the ligand conformer with the lowest total interaction energy, its total interaction energy with the protein, and the component energies. The back end also returns what is called the “Dockeye graphical display object”, or more succinctly, the Dockeye object. The Dockeye object is described and illustrated in more detail below. The front end then refreshes the display of the energy panel, displays the new ligand conformation if the lowest energy conformer has changed since the last placement change, and refreshes the display of the Dockeye object. The energy panel has colored coded bars for the total and component interaction energies and a low-water mark showing the most favorable total interaction energy so far. The more intensive back-end calculations are only invoked if the placement has changed, not for global translations/rotations of the protein/ligand. If the current placement represents a new minimum in total energy, the placement is automatically logged for future analysis or for an additional round of docking. The user can

also bookmark any current placement as a starting point for another docking session. The front end is written in Python.

Interaction Energy Module. The back-end module computes in real time the atomic interaction energies between protein and ligand for any given ligand placement and all the pregenerated ligand conformations. For the tests described here, the energy function is given by a sum of pairwise atomic Lennard-Jones or van der Waals (vdw) and Coulombic electrostatic terms:

$$E_{\text{tot}} = E_{\text{vdw}} + E_{\text{elec}} = \sum \frac{27}{4} \epsilon_{ij} \left(\frac{\sigma_{ij}}{r_{ij}}^9 - \frac{\sigma_{ij}}{r_{ij}}^6 \right) + \sum \frac{q_i q_j}{D r_{ij}} \quad (1)$$

where r_{ij} is the instantaneous distance between ligand atom i and protein atom j . σ_{ij} is the interatomic separation at which the vdw potential is zero, and ϵ_{ij} is the depth of vdw energy well at the optimal interatomic separation $r_{ij} = (3/2)^{1/3} \sigma_{ij}$ for a 9–6 type potential. The repulsive vdw term is softened somewhat from the usual inverse 12th power of distance to the inverse 9th power of distance to compensate to some degree for the use of a rigid protein. Currently, a single depth parameter for all atoms is used for simplicity. σ_{ij} is obtained from the sum of the van der Waals radii assigned to atoms i and j . Since the Dockeye software reads the atomic radii and charges from a ATM format file, the assignment of atomic radii and charges and the generation of the ATM format file from the parent PDB file must be done in a preprocessing step, prior to running Dockeye. For this work, the Bondi set of atomic radii, which are based on element type, were used.⁴³ The values of the vdw depth parameter and the dielectric constant D have default values of 0.1 kcal/mol and 80, respectively, but these may be changed at the startup of the Dockeye front end. The sum is over all pairs of the ligand–protein atoms within a cutoff distance (R_{cut}). R_{cut} can be changed by the user: It should be no larger than that which allows the energy to be evaluated in real time with no appreciable lag as the user adjusts the placement. For the tests described here, a R_{cut} value of 5 Å allowed smooth, lag free docking of ligands with up to 35 atoms and 12 torsional degrees of freedom on either a Mac Airbook with a 1.8 GHz Intel Core i5 Processor or a Linux workstation with a 4 core i7 Intel 2.67 GHz processor and a NVIDIA 960GTX GPU. The code is open source, so by modifying the back end a more sophisticated energy function than eq 1 and inclusion of solvent effects can be included at the user's discretion, computational resources permitting. The back end is currently written in C with a moderate level of optimization for CPUs.

Dockeye Graphical Display Object. The back end, in addition to calculating the total interaction energy and its components, examines every protein–ligand atomic pair interaction within the distance cutoff. For each such interaction, it stores the vector joining the two atoms, the midpoint of this vector, and the energy of interaction. These are then used to generate the Dockeye graphical display object as follows. For each such atomic pairwise interaction, a color-coded circle is generated. This circle is oriented perpendicular to the vector joining the two atoms with its center at the vector midpoint. The radius of the circle is inversely proportional to the distance between the two atoms, diminishing to zero at R_{cut} . (Figure 1a) The circle is color-coded by the pairwise interaction energy on a red (positive or unfavorable) to green (~zero) to blue (negative or favorable) scale. The back end

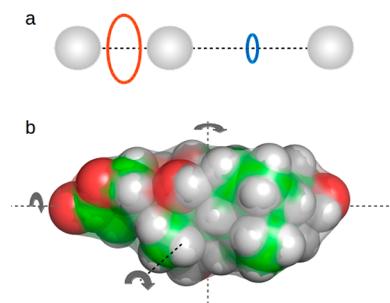


Figure 1. (a) Generation of the Dockeye object. A close unfavorable interaction between the left and middle atoms is indicated by the larger red-spectrum circle. A more distant, favorable interaction between the right and middle atoms is indicated by the smaller blue-spectrum circle. Position, size, orientation, and color of the circles can succinctly convey atomic level information relevant to docking without explicit display of the atoms. (b) Solvent excluding volume of the drug Digoxin is roughly ellipsoidal with approximate symmetry around the principal axes, particularly the horizontal one.

passes this collection of color-coded circles, which form the Dockeye object to the front end to be rendered.

Graphical Rendering. The graphical rendering of the Dockeye object, the energy panel, and the current best ligand conformation is currently done using the PyMOL graphics program, which also displays the protein target and the initial ligand conformation as a reference. While Dockeye is not inherently tied to any particular molecular graphics implementation, hosting it within PyMOL took advantage of some of the latter's attractive features. The Python front end of Dockeye can be run directly by PyMOL's in-built Python interpreter. Dockeye is imported as a Python module upon start up, and it functions as an extension of PyMOL's commands. Also, PyMOL has excellent ergonomics for manipulating two molecules/objects independently using a mouse. Upon switching to edit mode, global translation/rotation/zooming of the ligand–protein assembly is activated using the normal 3-button mouse actions. Upon depressing the shift key, translation/rotation/zooming with the same mouse buttons applies only to the object under the cursor, i.e., either ligand or protein. This facile ability to connect mouse actions to global, protein, or ligand motions without interruption or looking away from the screen, while a modest-sounding feature, is a great help when doing interactive docking.

Input File Preparation. Separate PDB format files for the protein and ligand are required for input. Dockeye, like many modeling programs, requires hydrogen coordinates to be generated for the ligand and the protein and for atomic radii and charges to be assigned. Assignment of radii and charges can be done by a number of different modeling packages, and there are numerous atomic charge sets or force field charge sets from which to choose. Thus, these capabilities were not built into the Dockeye program itself. In this work, as an example application, we tested Dockeye's ability to complement automated docking using Autodock Vina,²⁹ hereafter referred to as Vina. We therefore used the Autodock tools (ADT)⁴⁴ to add hydrogens and assign atomic charges to the ligand and protein. This enabled us to use exactly the same atomic charges with both docking methods. A Python utility program to place the ADT-generated charges into the B-factor field of the Dockeye input PDB file is provided with the software package. This utility program also assigns standard element-based atomic radii.

Flexible Ligands. When docking ligands that can adopt multiple conformations, they must be pregenerated as it is too expensive to generate multiple conformations on the fly and perform energy calculations on them all while examining new placements in real time. Multiple conformations are placed sequentially in the ligand input PDB file, delimited by the MODEL/ENDMDL records. There are a number of modeling programs that can parse a ligand structure to determine the rotatable bonds and then generate conformers. Since the ligand conformations are pregenerated externally to Dockeye, it is assumed that all these conformers are plausible candidates for binding; i.e., they are free of significant internal strain and steric clashes and represent conformations that would occur with appreciable frequency in solution. Thus, inside Dockeye, all conformers are treated on an equal footing; the preference is based only on the interaction energy with the protein. For the Dockeye applications described here, we used the ligand preparation facility of ADT to identify rotatable bonds, the information being written out in ADT's pdbqt format file.

When docking a flexible ligand interactively, the computational demand scales with the total number of ligand–protein interactions that must be evaluated for each placement. For Dockeye, this scales as $N_t = N_a \times N_c$, the product of the number of ligand atoms N_a and the number of pregenerated conformers loaded into Dockeye. The practical upper limit for Dockeye is thus governed by how large N_t can be and still give smooth, lag-free manipulation of the placements. N_c , not the ligand size or number of rotatable bonds per se, limits the application. If one wants to dock with an exhaustive set of conformers at say the level of 3 rotamers per torsional degree of freedom, 6 torsional degrees of freedom would result in $3^6 = 729$ conformers, which for ligands of 30 atoms or less is the current practical limit for Dockeye with the current workstations. For these cases, a Python utility program is provided with Dockeye. This program reads the torsional degrees of freedom in the pdbqt format file produced by ADT, generates the 3^{N_c} conformer coordinates, and places them in a PDB file in the required format. For larger ligands and/or ligands with more torsional degrees of freedom, rather than attempt exhaustive enumeration, it is preferable to select a set of conformers having low energy in the unbound, solution state. With larger, more flexible ligands, simply generating conformers by rotation of torsional angles results in many with high internal energy from steric clashes or other unfavorable interactions, which would not be good candidates for binding in the first place. The generation of lower solution energy conformers can be done independently of Dockeye in a number of ways: by Monte Carlo methods, molecular dynamics, and enhanced sampling using molecular mechanics packages such as CHARMM⁴⁵ or AMBER.⁴⁶

The space-filling shape of many ligands is approximately ellipsoidal and, therefore, somewhat symmetric about one or more of the principal geometric axes (Figure 1b). Shape complementarity is often a major factor in determining tight ligand binding. Experience has shown that for promising placements one or more of the three alternatives where the ligand is rotated 180° about a major axis are often also sterically permitted, competitive alternatives. This complicates the process of docking. It is possible but time-consuming for the user to manually flip the ligand around each of the three axes for each promising placement. Alternatively, a Python utility program provided with Dockeye will generate, for each ligand conformation, its three corresponding versions flipped

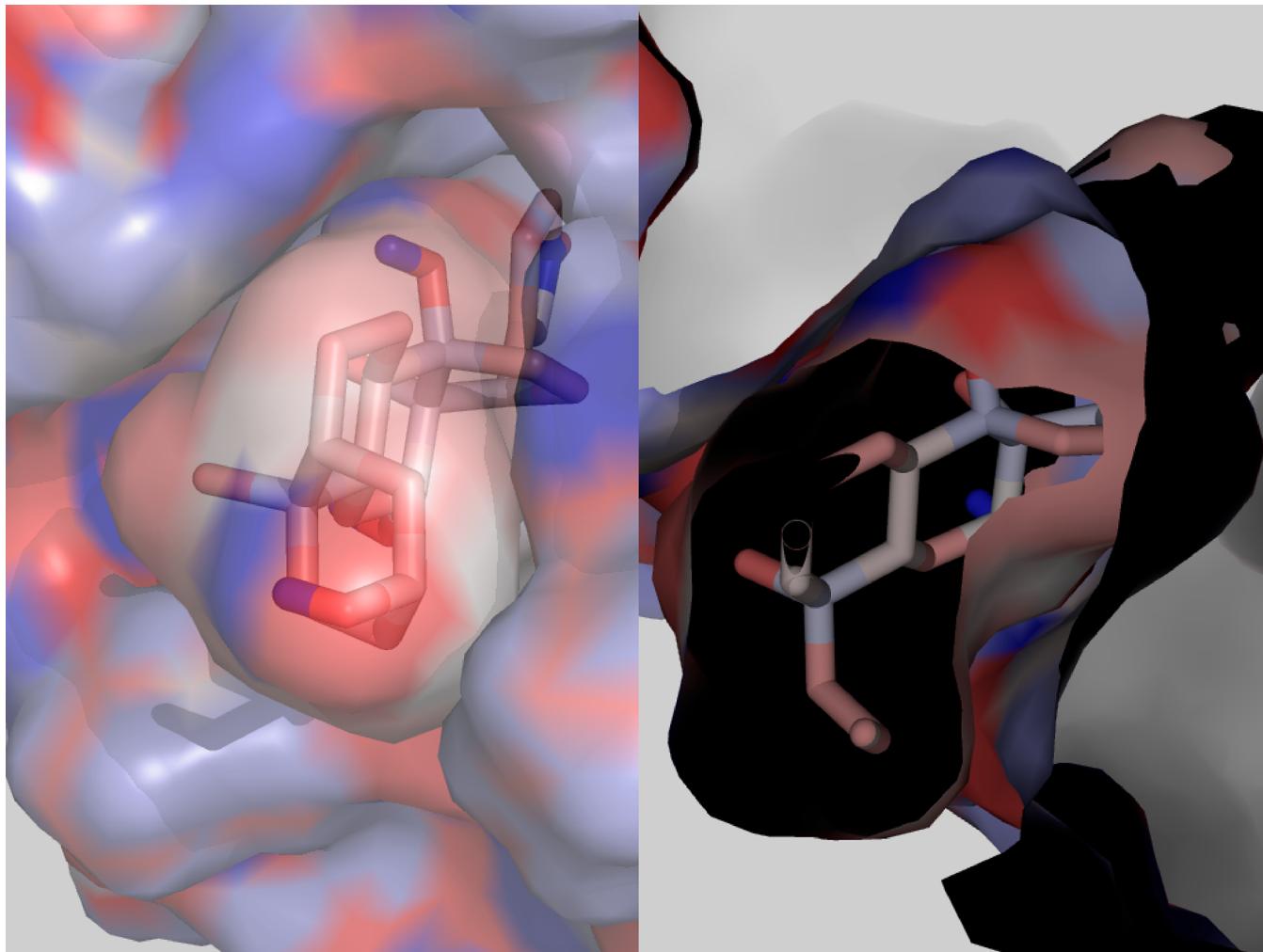


Figure 2. Crystal structure of the ligand Digoxin bound with high affinity to the antibody D2610 ($K_D = 67 \text{ pM}$). Two strategies for visualizing the ligand–protein shape complementarity. Left Panel: transparent molecular surfaces, color-coded by partial atomic charge (blue: +ve; red: -ve). Right Panel: Clipping or slicing through the two surfaces.

180° around the principal geometric axes (flipmers). At the cost of now evaluating a total of 4 times as many pregenerated conformers for each placement, Dockeye will automatically examine and then select the best of the four flipped orientations for every placement. Even if the ligand is not particularly symmetric, this optional ability to automatically examine the 4 “flipped” orientations can facilitate the user’s exploration of placements.

Comparison of Placements. To compare two placements, the root-mean-square distance (rmsd) between the corresponding pairs of atoms in the two placements was calculated. For convenience, we use the abbreviations D_{DE} for the rmsd between a Dockeye placement and the experimental placement, D_{VE} for the rmsd between a Vina placement and the experimental placement, and D_{DV} for the rmsd between a Dockeye placement and a Vina placement. In addition, the distance between two placements in rotational, translational, and internal coordinate (R/T/I) space was calculated as follows. The quaternion method of Kearsley⁴⁷ was used to obtain the rigid body rotation and translation required to superimpose one placement onto the other so as to minimize the rmsd. From the rotation quaternion, the axis of rotation and the magnitude of rotation about this axis, $\Delta\theta$, were obtained. The distance between placements in R/T space is

then $(|\Delta\mathbf{r}|, \Delta\theta)$, where $|\Delta\mathbf{r}|$ is the magnitude of the translation vector between placements in Ångstroms. The rmsd and R/T distances are informative in different ways. The rmsd quantifies the difference in atom positions, which is relevant to the atomic level interactions the ligand makes with the protein. The R/T distance from the experimental placement, on the other hand, can be informative about the success or failure of docking. A large value of rmsd can arise for different reasons. For example, if $|\Delta\mathbf{r}|$ is large, this indicates that docking has missed the correct target site. On the other hand, if $|\Delta\mathbf{r}|$ is small and $\Delta\theta$ is large, this indicates that the correct site was identified but not the correct orientation within this site. Furthermore, in this case, if $\Delta\theta$ is close to 180° (within say 10–15°), this may be a confounding effect of pseudo or actual symmetry in the ligand (Figure 1b), so that placements related by 180° flips are of very similar quality. Knowing this, one could routinely check all four 180° flipped versions of every placement. This four-way check can be automated in Dockeye, as described in the previous section. For rigid ligands, the residual rmsd after the rigid body superposition is always zero. For a flexible ligand, however, if the two placements have different ligand conformations, the residual rmsd is not zero. Consequently, this residual rmsd is a measure, in terms of differences in atomic positions in Cartesian space, of the

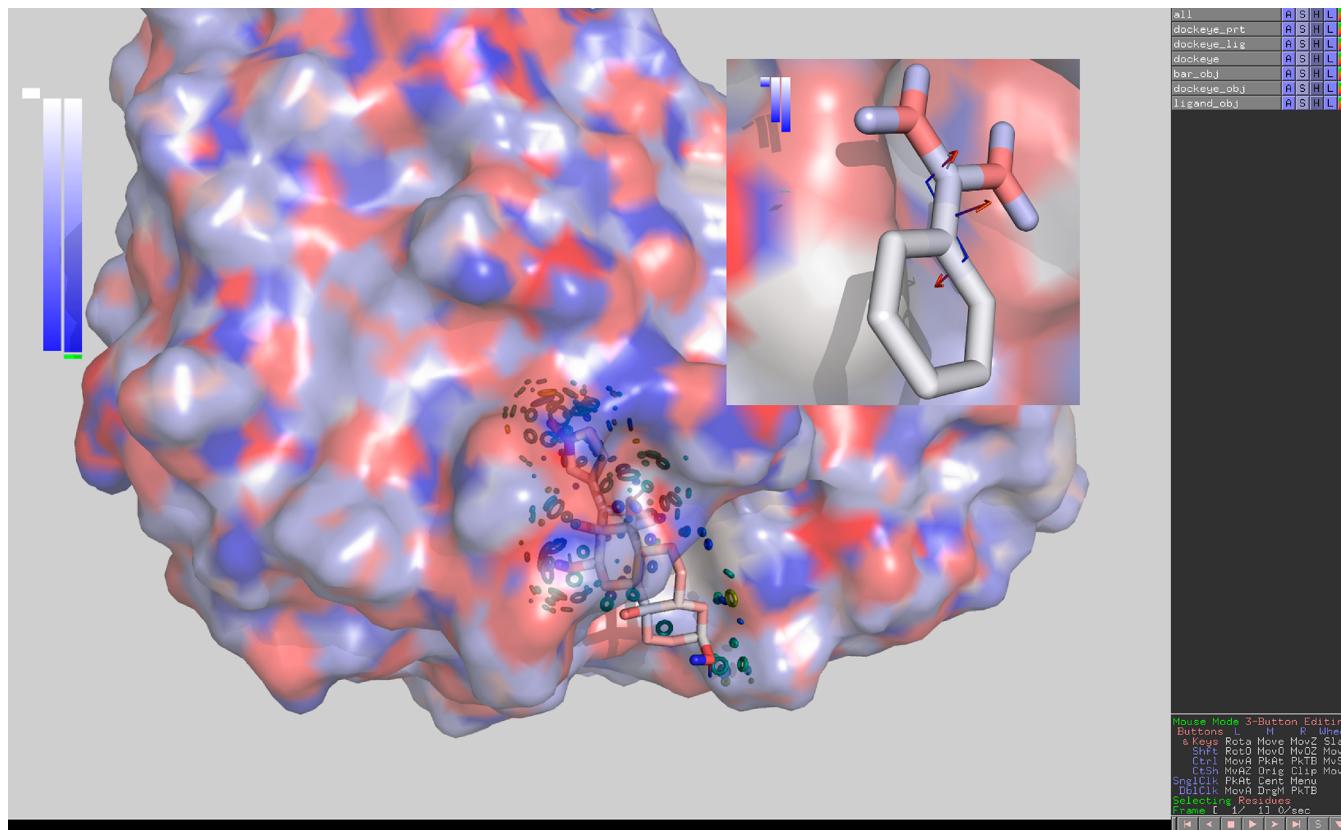


Figure 3. Screen shot of Dockeye running within PyMOL. The interaction energy panel (upper left) shows, from the left, the electrostatic, van der Waals, and total interaction energies, respectively. The lowest energy so far is indicated by the green horizontal bar. The ligand Digoxin in stick representation is surrounded by the atomic interaction energy circles of the Dockeye object. The protein target, antibody D2610, is rendered as a partially transparent molecular surface. Both molecules are color-coded by partial atomic charges (blue: +ve; red: -ve). PyMOL's object panel (upper right) allows one to toggle on and off the displays of the Dockeye object, the energy bars, and the alternative ligand conformation as well switch the back-end interaction energy calculation on or off, as required. The inset figure shows, for the ligand benzamidine, an alternative representation of the interaction with the protein target: The single arrow and double arrow-torque bar indicate the net translational and rotational forces on the ligand, respectively. Arrows are color-coded from blue to red, indicating the increasing magnitude of the force/torque.

distance between conformations. Regarding a comparison with the experiment, the residual rmsd is one way to quantify the error in the ligand conformation.

Dockeye Workflow.

- (1) Preparation of separate PDB format files for ligand and protein (If the files are obtained from the PDB database, crystallographic water, ions, and crystallization compounds are removed.)
- (2) Addition of hydrogens and the assignment of charges and radii to the ligand and protein
- (3) Identification of rotatable bonds in the ligand and generation of conformers and flipmers
- (4) Initialization of Dockeye inside PyMOL: (i) execution of the Python front end, which imports the Dockeye module; (ii) loading of protein and ligand PDB files (During loading, a flag controls whether the electrostatic term is included. If this is false, then only shape complementarity contributes to the docking. At this point, the default dielectric constant, van der Waal's depth parameter, and interaction cutoff distance may be overridden.)
- (5) Switch PyMOL to edit mode to enable interactive docking
- (6) Interactive docking and bookmarking as needed

- (7) Postdocking placements are extracted from the Dockeye log file using a Python utility program provided with the software
- (8) Placements are then compared quantitatively using the initial rmsd, $|\Delta r|$, $\Delta\theta$, and the residual rmsd

RESULTS

The core function of the Dockeye software is to render, in real time, a quantitative and graphical readout of the complementarity between a ligand and its target protein in order to guide docking and ligand design. The concept of complementarity between the ligand and binding site is central in computer aided drug design, in terms of shape complementarity and complementary of physical chemical interactions. Good shape complementarity is achieved by minimizing void volume at the ligand–protein interface while avoiding steric overlap of the atoms. Steric overlap of the atoms would be prohibitive of strong binding. The minimization of the void volume promotes favorable packing interactions (attractive van der Waals interactions). The close approach of ligand and protein atoms also allows for complementary physical chemical interactions, namely, the formation of anhydrous interfaces between apolar surfaces required for the hydrophobic effect, approach of H-bond donors and acceptors to within H-bonding distance, and more generally the approach of

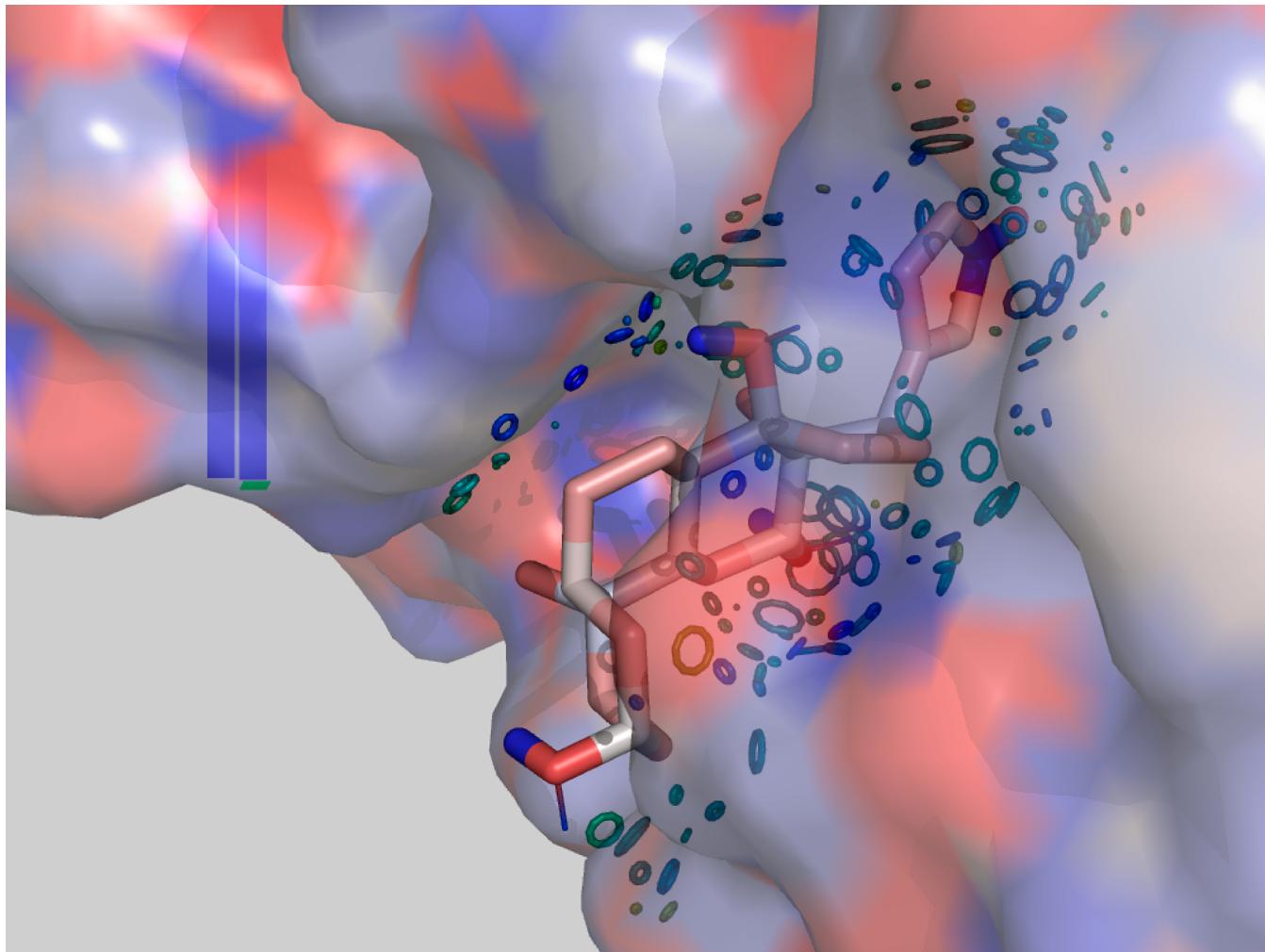


Figure 4. Close-up of the ligand and Dockeye object for a high affinity placement: The experimental placement of the ligand Digoxin bound to the antibody D2610 with $K_D = 67 \text{ pM}$. Both molecules are color-coded by partial atomic charges (blue: +ve; red: -ve). Characteristics of a high affinity placement are a ligand surrounded by a dense sheath of favorable atomic interactions (circles at the blue end of the spectrum, indicating negative energy contributions) with few or no unfavorable interactions (circles at the red end of the spectrum). The bottom center of the image shows, as wire frame, an alternative lower energy ligand conformation automatically selected by Dockeye, in this case enabling a ligand H-bond donor to make a bond with the protein.

oppositely charged atoms for favorable electrostatic interactions. Good chemical complementarity, however, may be achieved without having overall shape complementarity, providing the relevant protein and ligand atoms can approach closely enough. From the perspective of molecular graphics, shape is most advantageously displayed in terms of the molecular surface. The molecular surface is defined as the boundary formed by a water-sized sphere as it rolls over the van der Waals surface of the atoms⁴⁸ (Figure 1b). However, it is a challenge to visualize the extent or lack thereof of shape complementarity of a protein–ligand complex. The surfaces themselves tend to obscure the features of interest: the distance between the two molecules' molecular surfaces and whether they overlap (steric clashes) or have significant separation. This is difficult even with partially transparent rendering of the surfaces (Figure 2a) or by slicing and clipping the image (Figure 2b), even more so for whatever part of the interface happens to be at the back in the current view.

An alternative representation of shape is through the van der Waals (vdw) potential, eq 1, which is effectively synonymous with molecular shape at the level of atomic forces. The

repulsive term penalizes steric overlap and provides the space filling properties, while the attractive term favors well packed interfaces without large cavities. Dockeye calculates and provides numerical and graphical readout of the net vdw interaction energy between ligand and protein in real time as the ligand placement is manipulated with the mouse (Figure 3). The value of total vdw interaction energy by itself does not, however, easily provide information about what part of the ligand is making good or bad interactions, especially during real-time placement manipulation. For this, Dockeye generates what is called the “Dockeye graphical display object” or, more succinctly, the Dockeye object. As described in the Methods and Algorithms section, the Dockeye object is a richly encoded representation of the atomic level interactions between ligand and protein (Figures 3–7). For each atomic interaction between ligand and protein, a color-coded circle of varying size is placed midway between the two interacting atoms (Figure 1a). The closer the two atoms, the larger is the radius. The circle is color-coded on the basis of the sign and magnitude of the pairwise interaction energy from blue (negative or favorable) to green to red (positive or unfavorable). In

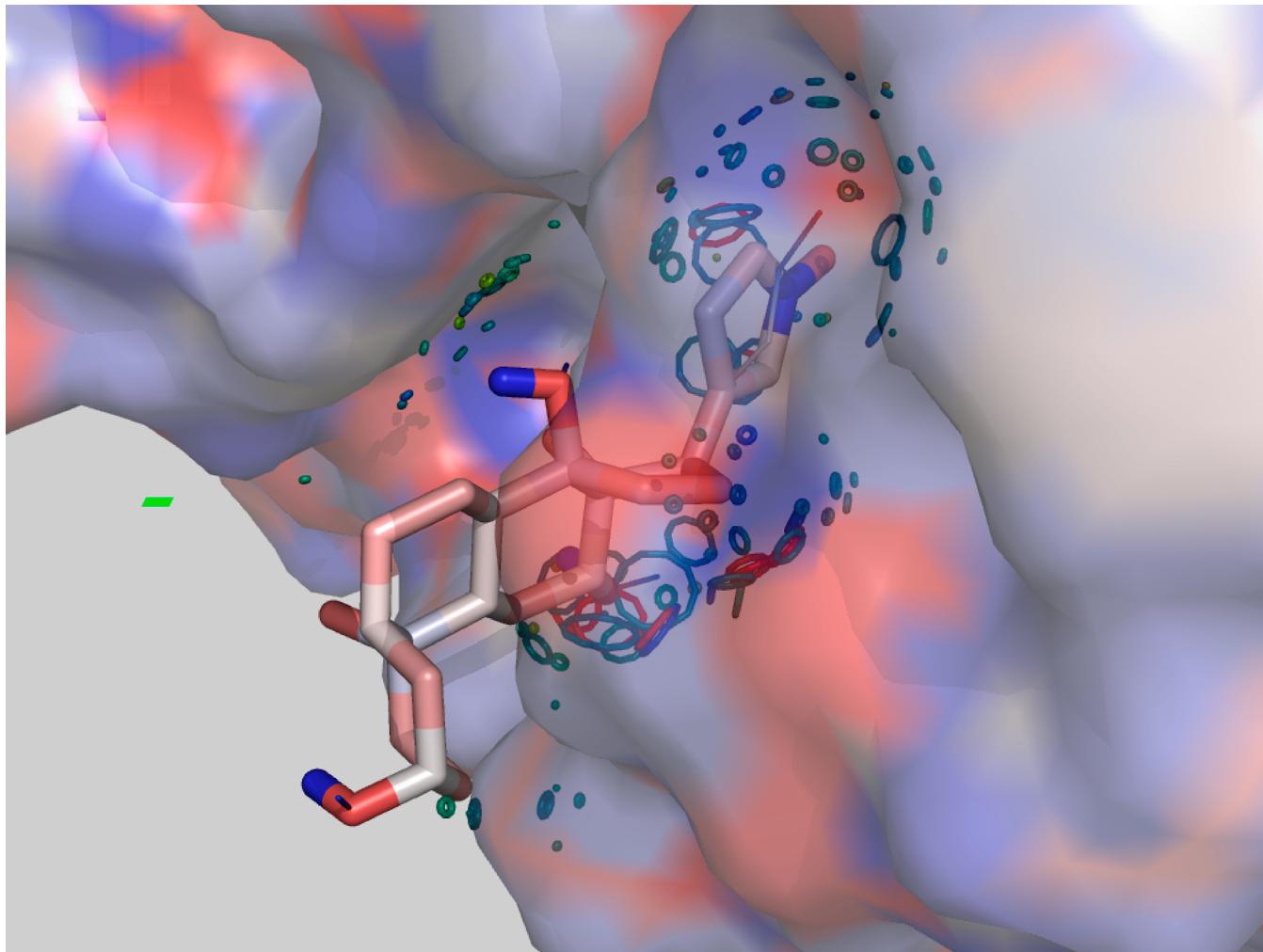


Figure 5. Close-up of the ligand and Dockeye object for a poor placement: Unfavorable interactions on the lower side of the ligand are indicated by the red atomic interaction circles. Also, there is an absence of favorable interactions over a significant fraction of the ligand on its upper side. Both provide concrete visual cues as to how to improve the placement or to modify the ligand by removing/adding groups. Both molecules are color-coded by partial atomic charges (blue: +ve; red: -ve).

aggregate, the collection of circles that make up the Dockeye object convey multiple aspects of the interaction. The number of circles is a measure of the spatial density of the interactions. If many of these are of larger size and at the blue end of the spectrum, this indicates good complementarity (**Figure 4**).

If one wants to reveal just the shape complementarity, this can be done by switching off the electrostatic term and using only the vdw interaction to generate the Dockeye object. The Dockeye object is graphically lean, so that the interactions over the entire ligand/protein interface can be visualized at once or with minimal rotation of the view (**Figure 4**). At the same time, the distribution or absence of circles around the ligand quickly reveals which parts of the ligand contribute the most to complementarity and which parts contribute little or none (**Figure 5**). Circles on the red end of the spectrum pinpoint where there are unfavorable interactions such as bumps, steric overlaps, or charge–charge repulsion. Alternatively, Dockeye can display the net translational force on the ligand from the protein as a vector arrow color-coded by the magnitude of this force on a blue-to-red scale. In a similar manner, the net rotational force on the ligand is displayed as a torque-bar/double arrow, color-coded in the same manner by the magnitude of the torque (**Figure 3**, inset). This representation

displays aggregate, molecular level information about the ligand–protein interaction, while the interaction circles display more granular information at the atomic level.

Since the Dockeye object is updated in real time as the placement is adjusted, this information can easily be used to guide interactive docking. A short video clip illustrating this is provided in the [Supporting Information](#). For ligand design, the analysis of the distribution of the interaction circles can be used to suggest places where groups could be added or removed from the ligand to improve its binding (**Figure 5**). The Dockeye object can also be used in nondocking mode to aid in the analysis of protein–ligand complexes of known structure.

As illustrated in **Figure 6**, one can easily identify, even among many atoms and in a crowded display, significant interactions. By focusing on the interaction circle elements that indicate more significant interactions (larger circles at the two ends of the color spectrum), one can locate the interaction pairs responsible using the fact that the circle element is positioned orthogonal to the vector joining the two atoms at the midpoint. In **Figure 6**, easy identification of a strong H-bond interaction between the ligand and protein is illustrated.

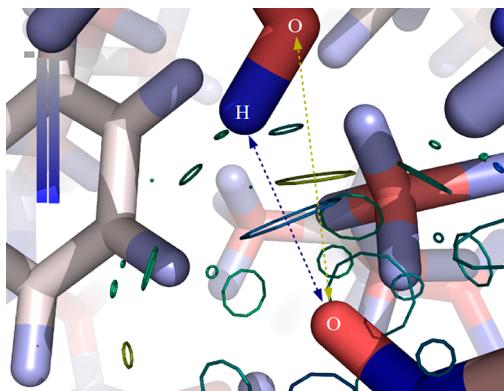


Figure 6. Circle elements in the Dockeye object enable quick identification of important atomic interactions. The large blue and yellow circles indicate two close interactions, the former favorable and the latter unfavorable, respectively. Since the circles are positioned orthogonal to the interaction at the midpoint, this reveals the interactions to be between the ligand oxygen and the hydrogen bond donor and acceptor groups of a tyrosine hydroxyl. Both molecules are color-coded by partial atomic charges (blue: +ve; red: -ve).

Figure 7 is a snapshot from docking a ligand of modest affinity, benzamidine/trypsin ($K_D = 17 \mu\text{M}$). Compared to the well docked high affinity ligand in Figure 4 (Digoxin, $K_D = 67 \text{ pM}$), there are less circles and, on average, they are smaller; additionally, fewer are at the blue (favorable energy) end of the color spectrum. This snapshot also illustrates how Dockeye automatically examines all four 180° flipped versions of a placement: It has exchanged the benzyl ring and H-bonding ends of the ligand to move the H-bonding groups from the solvent into the binding pocket.

Dockeye is designed to be complementary to automated docking programs. While the latter have higher throughput than manual docking and have little or no user input during the actual docking, Dockeye has a relatively low throughput, but user input is used continuously. In automated docking, the expertise is in effect prepackaged, for example, by improving the algorithm and scoring function through cycles of training and testing against a set of known protein–ligand complexes. In contrast, for interactive docking, the expertise resides largely in the user, based on their previous experience, knowledge, and even intuition about the requirements for high affinity binding.

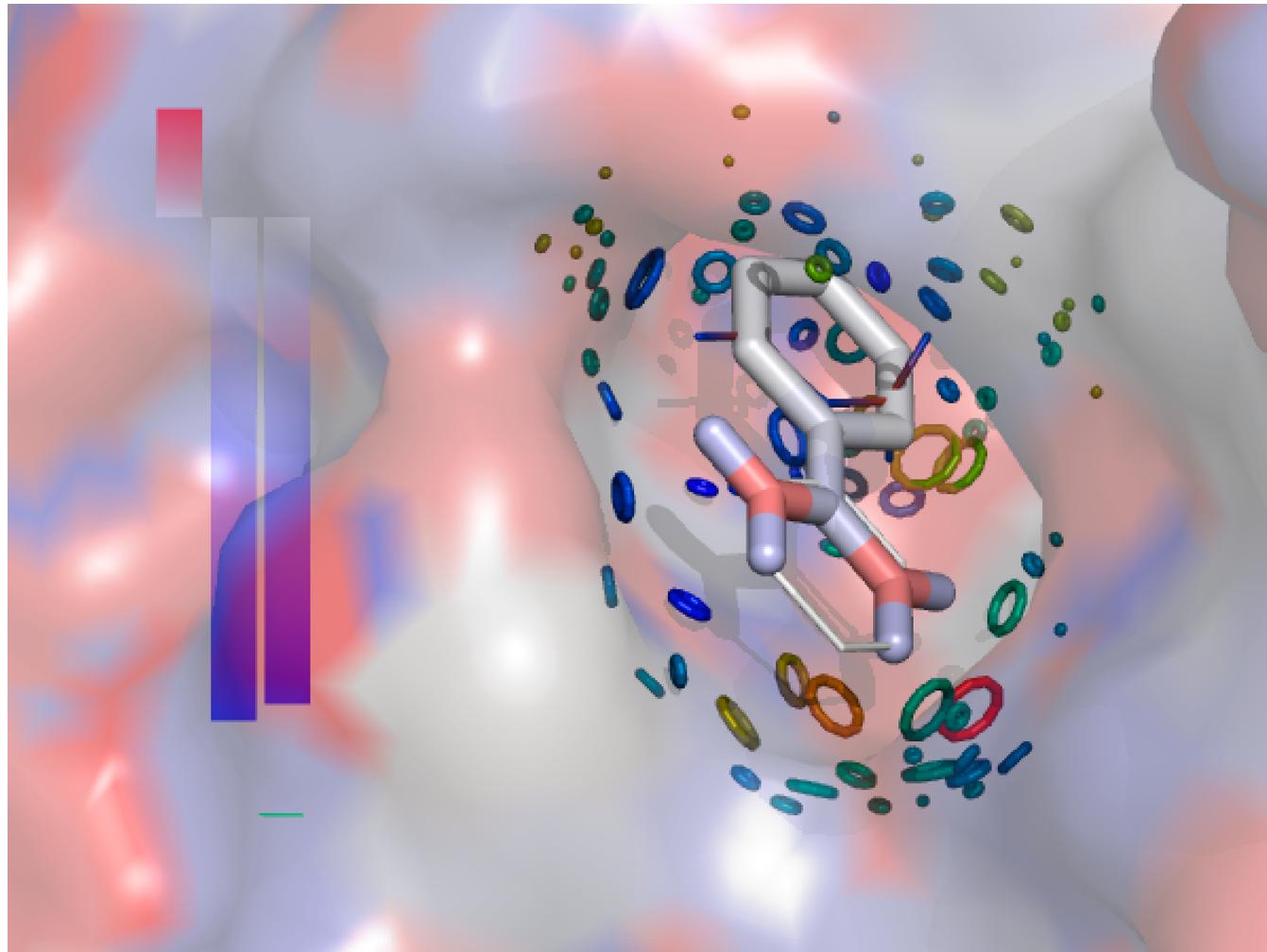


Figure 7. Close-up of the ligand and Dockeye object while docking a moderate affinity ligand, benzamidine, against trypsin, with an experimental K_D of 17 μM . The original reference ligand conformation is shown in stick representation, while the current best conformation is shown as a wire frame. Note that Dockeye has automatically screened the four 180° rotated (flipped) candidates for each conformer and, in this case, has exchanged the positions of the aromatic ring and H-bonding groups relative to the initial orientation, thus moving the H-bonding groups out of the solvent and toward the suitable H-bond acceptors on the target protein. Both molecules are color-coded by partial atomic charges (blue: +ve; red: -ve).

Table 1. Docking of Rigid Ligands^a

PDB ID	ligand code ^c	distance: ^b Dockeye placement to PDB			distance: ^b Vina placement closest to PDB				distance: ^b Vina placement closest to Dockeye			
		rmsd, Å	Δrl , Å	Δθ	Vina rank	rmsd, Å	Δrl , Å	Δθ	Vina rank	rmsd, Å	Δrl , Å	Δθ
185L	IND	0.9	0.4	24	1	0.5	0.4	8	1	0.8	0.3	27
187L	PXY	0.7	0.5	14	1	0.7	0.5	15	1	0.1	0.1	6
188L	OXE	1.1	0.9	22	1	0.9	0.6	23	1	0.6	0.3	16
1A9Q	HPA	0.9	0.8	15	1	0.5	0.4	14	1	0.9	0.6	20
1ACJ	THA	0.5	0.3	13	4	0.3	0.1	8	4	0.3	0.2	6
1AI7	IPH	0.6	0.3	10	3	2.2	0.8	172	4	2.4	0.4	169
1APB	FCA	1.2	0.8	29	1	0.8	0.2	8	1	1.2	0.8	22
1B42	M1A	4.3	1.0	121	6	1.3	0.7	21	7	0.6	0.3	5
1BCU	PRL	0.6	0.4	2	1	0.9	0.6	11	1	1.0	0.7	9
1BAP	ARA	0.7	0.6	4	1	0.9	0.5	6	1	0.7	0.2	9
1BKY	1MC	5.5	3.3	172	3	1.2	0.9	21	1	0.5	0.3	10
1C3X	8IG	2.5	1.8	33	4	2.9	1.3	56	4	1.9	1.4	28
1D6N	PPO	1.2	0.5	33	6	1.4	0.4	42	6	0.5	0.2	16
1E5A	TBP	5.6	4.2	147	5	5.0	3.4	67	5	2.5	1.6	67
1ENU	APZ	4.6	0.6	180	1	0.2	0.2	3	4	2.7	0.3	57
1F0Q	EMO	0.6	0.3	8	1	1.0	0.4	14	1	0.9	0.7	9
1F3E	DPZ	3.7	0.6	172	1	0.4	0.4	2	4	2.9	1.1	55
1FSG	9DG	0.6	0.3	3	4	6.9	6.3	80	4	6.8	6.2	76
1FV9	172	1.0	0.1	23	3	0.7	0.3	13	3	0.6	0.2	12
1GT1	ANC	2.5	0.3	168	5	1.4	1.4	7	4	1.0	0.2	14
1H62	ANB	1.4	0.5	25	8	1.9	0.6	35	8	0.6	0.1	11
1HN2	ANC	2.6	0.6	173	7	0.4	0.4	10	3	0.6	0.4	13
1I37	DHT	0.5	0.2	12	1	0.3	0.2	5	1	0.6	0.3	14
1I80	9HX	0.5	0.3	14	1	0.3	0.3	5	1	0.4	0.3	12
median		1.1	0.5	23		0.9	0.5	13				

^aNo rotatable bonds except for torsions of the type X-X--X-H, where H is hydrogen and X is any heavy (non-hydrogen) atom. ^bDifference between placements expressed both as the root-mean-square distance (rmsd) in angstroms between all atoms and as the magnitude of translation and rigid body rotation ($|\Delta rl|$, $\Delta\theta$) required to minimize the rmsd. ^cResidue name used in the PDB entry. See the Supporting Information for the full ligand names.

Table 2. Docking of Flexible Ligands with 1 Rotatable Bond^a

PDB	ligand code ^d	distance: ^b Dockeye placement to PDB			distance: ^b Vina placement closest to PDB				distance: ^b Vina placement closest to Dockeye			
		rmsd, Å ^c	$ \Delta rl $, Å	$\Delta\theta$	Vina rank	rmsd, Å	$ \Delta rl $, Å	$\Delta\theta$	Vina rank	rmsd, Å	$ \Delta rl $, Å	$\Delta\theta$
1AF2	U	0.9 (0.5)	0.5	19	9	2.3	0.8	71	9	2.7	1.0	86
1BCD	FMS	2.1 (1.3)	0.7	89	3	2.3	1.1	119	1	1.7	0.4	72
1BRA	BEN	0.8 (0.6)	0.4	11	1	1.5	0.1	22	1	1.5	0.4	25
1BTY	BEN	2.5 (0.6)	0.1	178	1	0.6	0.1	11	1	2.5	0.3	173
1DBB	STR	0.4 (0.0)	0.3	9	9	1.6	1.2	18	9	1.7	1.4	17
1DGM	ADN	2.0 (0.0)	1.6	22	2	4.0	2.5	99	2	4.0	2.5	99
1EO6	IPB	1.0 (0.4)	0.9	10	5	2.3	1.7	40	5	2.8	2.5	35
1EOC	4NC	0.4 (0.0)	0.4	0	9	5.3	3.2	125	9	5.4	3.3	125
1G3M	PCQ	0.5 (0.3)	0.3	7	6	1.5	0.7	30	6	1.4	0.8	23
1HNN	SKF ^e	1.5 (1.2)	0.4	40	1	1.3	0.3	26	2	1.0	0.9	5
median		1.0	0.4	15		1.9	1.0	35				

^aNumber of rotatable bonds involving torsions of the type X-X--X-X, where X is any heavy (non-hydrogen) atom. ^bDifference between placements expressed both as the root-mean-square distance (rmsd) in angstroms between all atoms and as the magnitude of translation and rigid body rotation ($|\Delta rl|$, $\Delta\theta$) required to minimize the rmsd. ^cAll-atom RMSD between placements. The number in the brackets is the residual RMSD after optimal rigid body alignment. A nonzero value quantifies the difference in ligand conformations between the two placements. ^dResidue name used in the PDB entry. See the Supporting Information for the full ligand names. ^eVina generated only 6 placements.

To illustrate the complementary nature of the two approaches to docking, we performed docking on a set of 58 protein-ligand complexes using both the automated docking program Vina Autodock²⁹ and Dockeye. Vina Autodock uses a knowledge-based scoring function, while Dockeye uses the physics-based scoring function of eq 1. The test complexes were selected from the PDDBind database, drawn from the

Core list of 4852 complexes.^{49,50} A selection of complexes with rigid ligands and ligands of varying degree of flexibility was chosen. Of the 58 complexes, 24 had rigid ligands (0 rotatable bonds), 10 had 1 rotatable bond, 11 had 2 rotatable bonds, and 13 had 3 or more rotatable bonds (Tables 1–4 and S1). Beyond ligand flexibility, no other selection criteria for the complexes, such as experimental affinities, were used. In this

Table 3. Docking of Flexible Ligands with 2 to 3 Rotatable Bonds^a

PDB	ligand code ^d	distance: ^b Dockeye placement to PDB			distance: Vina placement closest to PDB				distance: Vina placement closest to Dockeye			
		rmsd, Å ^c	Δrl , Å	Δθ	Vina rank	rmsd, Å	Δrl , Å	Δθ	Vina rank	rmsd, Å	Δrl , Å	Δθ
2 Rotatable Bonds												
184L	I4B	0.5 (0.0)	0.4	9	3	1.2	0.7	26	3	1.4	1.0	35
1A69	FMB	1.0 (0.4)	0.7	16	1	0.6	0.2	8	1	1.0	0.7	18
1ADD	1DA	0.3 (0.3)	0.2	2	1	0.9	0.3	6	1	0.8	0.2	5
1AI4	DHY	1.0 (0.8)	0.5	6	3	4.9	3.3	161	3	5.2	3.6	155
1BN3	AL6	1.5 (1.2)	0.5	17	2	1.5	0.4	12	2	0.6	0.4	11
1BNN	AL1	1.5 (1.2)	0.5	20	2	1.5	0.4	12	2	0.9	0.6	16
1BUG	URS	1.9 (1.1)	0.8	88	6	6.7	5.3	143	6	6.4	4.9	145
1C5N	ESI	2.6 (1.4)	0.6	66	1	2.1	0.8	51	1	1.7	0.8	29
1E9H	INR	0.3 (0.0)	0.3	4	1	0.8	0.1	1	1	0.9	0.2	3
1FTL	DNQ	3.1 (0.6)	0.4	164	1	3.1	0.1	172	1	0.8	0.4	6
1KR3	113	0.5 (0.4)	0.3	3	7	1.4	1.2	12	2	1.5	1.2	25
median		1.0	0.5	16		1.5	0.4	12				
3 Rotatable Bonds												
186L ^e	N4B	0.5 (0.0)	0.3	17	1	1.9	0.3	179	1	1.8	0.3	166
1A9U	SB2	0.8 (0.7)	0.3	4	1	1.8	1.1	7	1	1.7	1.0	7
1AIS	MNP	1.2 (0.9)	0.6	10	4	1.0	0.7	17	4	1.6	1.2	21
1BNNT	AL2	1.2 (1.1)	0.4	6	1	1.8	0.6	3	1	1.7	0.7	26
1BNU	AL3	0.7 (0.5)	0.3	5	1	1.8	0.6	11	1	1.7	0.5	14
1BNV	AL7	1.6 (1.2)	0.5	16	1	1.2	0.5	9	1	0.9	0.3	11
1DZK	PRZ	1.3 (1.0)	0.9	9	7	1.7	0.4	48	7	1.7	0.8	49
1E1V	CMG	1.1 (0.7)	0.8	2	9	0.9	0.5	5	9	0.9	0.5	3
1FCX ^f	184	0.4 (0.0)	0.2	9	1	0.8	0.2	6	1	0.7	0.2	7
1FCY ^f	564	0.4 (0.0)	0.4	2	1	0.6	0.1	1	1	0.7	0.3	3
median		0.9	0.4	7		1.5	0.5	8				

^aNumber of rotatable bonds involving torsions of the type X-X--X-X, where X is any heavy (non-hydrogen) atom. ^bDifference between placements expressed both as the root-mean-square distance (rmsd) in angstroms between all atoms and as the magnitude of translation and rigid body rotation (|Δrl|, Δθ) required to minimize the rmsd. ^cAll-atom RMSD between placements. The number in the brackets is the residual RMSD after optimal rigid body alignment. A nonzero value quantifies the difference in ligand conformations between the two placements. ^dResidue name used in the PDB entry. See the Supporting Information for the full ligand names. ^eVina generated only 4 placements. ^fVina generated only 2 placements.

Table 4. Docking of Flexible Ligands with 8 or More Rotatable Bonds^a

PDB	ligand code ^d	distance: ^b Dockeye placement to PDB			distance: Vina placement closest to PDB				distance: Vina placement closest to Dockeye			
		rmsd, Å ^c	Δrl , Å	Δθ	Vina rank	rmsd, Å	Δrl , Å	Δθ	Vina rank	rmsd, Å	Δrl , Å	Δθ
8 Rotatable Bonds												
1A4R	GDP	1.7 (0.5)	0.7	21	1	1.8	0.6	13	1	1.7	0.3	14
9 Rotatable Bonds												
1ATL	0QI	1.8 (1.2)	0.9	29	9	1.4	0.6	15	9	1.3	0.5	17
12 Rotatable Bonds												
1CNX	EG2	1.9 (1.5)	0.6	3	16	1.6	0.3	6	16	1.9	0.5	8

^aNumber of rotatable bonds involving torsions of the type X-X--X-X, where X is any heavy (non-hydrogen) atom. Docking was performed with the 30 lowest energy conformers obtained from a Monte Carlo sampling of 1500 conformers in the unbound state. ^bDifference between placements expressed both as the root-mean-square distance (rmsd) in angstroms between all atoms and as the magnitude of translation and rigid body rotation (|Δrl|, Δθ) required to minimize the rmsd. ^cAll-atom RMSD between placements. The number in the brackets is the residual RMSD after optimal rigid body alignment. A nonzero value quantifies the difference in ligand conformations between the two placements. ^dResidue name used in the PDB entry. See the Supporting Information for the full ligand names.

definition of ligand flexibility, a rotatable bond was defined as a torsional degree of freedom that involved four connected non-hydrogen (heavy) atoms, schematically X-X-X-X. After addition of polar hydrogens in the preparation stage of Vina, some ligands gained additional torsional degrees of freedom, for example, from the rotation of the polar H of a hydroxyl group around the X-OH bond. The classification of ligand flexibility in Tables 1–4 refers to the number of all-heavy atom type torsions. Kollman type charges were added to the ligands in the preparation stage of Vina.⁵¹ The identical hydrogen

atom addition and charge assignment were used for the ligand preparation for Dockeye in order to use the same ligand model with both docking methods. Exhaustive enumeration of all conformers at the level of 3 rotamers per torsional degree of freedom was done using the utility Python program provided with Dockeye. To demonstrate that Dockeye can be used with larger and more flexible ligands, we also docked three ligands with more degrees of freedom, GTP, a hydrolase inhibitor, and a carbonic anhydrase inhibitor with 8, 9, and 12 torsional degrees of freedom, respectively; the largest of these ligands

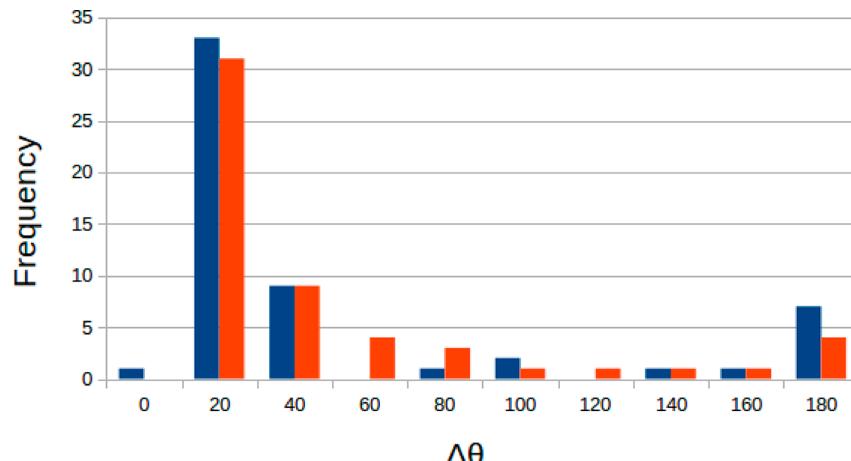


Figure 8. Frequency of errors in the orientation of the placement, relative to the experiment. Blue: Dockeye; Red: Vina.

had 42 atoms. For these three ligands, sampling of the conformers in the unbound, solution state was performed using the Monte Carlo algorithm.⁵² Conformer energies were computed using a molecular mechanics force field with an implicit solvent model.⁵³ Conformers were generated by random rotations around rotatable bonds, and the new conformer was accepted or rejected on the basis of the change in total energy according to the Monte Carlo protocol using a temperature of 298 K. 5 000 samples were generated, and the 30 ligand conformers with the lowest total energy (Internal plus solvation) were used for docking.

The subtype of the docking task we examined here is placement discovery, distinct from site discovery. This corresponds to a situation where the binding site is known from knowledge of where a substrate or inhibitor is bound, the identity of active site residues in an enzyme, or comparable information, but it is not known, obviously, from the actual structure of the ligand complex itself. The question then is can the experimental placement and ligand conformation be discovered? Both Vina and Dockeye were given the same information, namely, the general location of the binding site. This was used in Vina to position the center of its $20 \times 20 \times 20$ Å search box. Default values were used for all other Vina parameters. The output from Vina is N different placements, which differ from each other by at least 1.5 Å rmsd, ranked in descending order of quality by the Vina scoring function. The default is $N = 9$. For Dockeye, the ligand was positioned randomly 50 Å away from the center of the protein, far enough that its interaction energy was zero.

The general location of the binding site was indicated to the user by highlighting a few residues in the region. Interactive docking was continued until no improvement in interaction energy could be achieved after a reasonable time spent docking.

Vina docking of all compounds was done after the results of Dockeye were compiled, so the interactive docking was done completely independently of the automated docking. The results of the two approaches were then compared with each other and with the experimental placement (obtained from the PDB entry indicated). Distances between placements were measured by the root-mean-square distance (rmsd) between atoms and in rotational/translational coordinate space by the magnitude of translation and rotation ($|\Delta r|$, $\Delta\theta$) required to superimpose one placement optimally on the other. The

distance of the Dockeye placement and each of the nine Vina placements was compared to the crystal structure placement. The results are given in Tables 1–4. For Vina, the distance is given for the placement closest to the crystal structure, along with the rank (from 1 to 9) of that placement. The distance between the Dockeye placement and all nine Vina placements was also computed using the above two measures. Results are given for the Vina placement closest to the Dockeye's placement, along with its Vina rank (last four columns, Tables 1–4). Thus, a difference in the two rankings in these tables indicates that different Vina placements were the closest in the two comparisons. For summary values, medians rather than mean values were used because the distributions of distances are very skewed upward. For example, a few placements that are off by about 180° give very large errors.

Both docking methods gave similar results with median errors in rmsd ranging from 0.9 to 1.9 Å, depending on the data set, and with median errors in $|\Delta r|$ and $\Delta\theta$ in the ranges of 0.5–1.0 Å and 7–35°, respectively. Interestingly, there was no discernible trend to increased error with increasing ligand flexibility, as might have been expected due to the increase in degrees of freedom to search. We attribute this to the fact that more flexible ligands tended to be larger: The mean number of ligand atoms for 0, 1, 2, and 3 rotatable bonds was 15, 16, 21, and 23 atoms, respectively (Table S1). Larger ligands in one way are no harder to dock into binding pockets because the number of plausible placements is more restricted sterically. Even for the much more flexible ligands in Table 4, with 8–12 rotatable bonds, the docking accuracy is comparable with that of less flexible ligands. Accuracy was also comparable between the two docking methods. However, for the most flexible ligand, EG2, with 12 rotatable bonds, it was necessary to reach much further down the ranking of Vina's top scorers, to the 16th best placement.

A notable form of placement error is indicated by $\Delta\theta$ values between placement and crystal structure close to 180°. Docking of several ligands suffered from this error. A plot of the frequency of $\Delta\theta$ shows the majority of values cluster near the low-end range of 20–40° with larger angle errors decreasing in frequency until one approaches the 180° mark (Figure 8). One reason for this has already been discussed, namely, that many ligands have an approximate ellipsoidal symmetry shape (Figure 1b), meaning the placements flipped

Table 5. Summary of Ligand Docking Accuracy

category	ligand flexibility (number of rotatable bonds)				
	rigid (N = 24)	1 bond (N = 10)	2 bonds (N = 11)	3 or more bonds (N = 13)	all (N = 58)
top scored Vina pose is closest to the experiment (Vina True Positive)	12 (50%)	3 (30%)	5 (45%)	8 (62%)	28 (48%)
top scored Vina pose is not the one closest to the experiment (Vina False Positive)	12 (50%)	7 (70%)	6 (55%)	5 (38%)	30 (52%)
the Vina pose closest to the Dockeye pose is also closest to the experimental pose (Consensus True Positive)	17 (71%)	8 (80%)	10 (91%)	13 (100%)	48 (83%)
the closest Vina and Dockeye poses are less than 1.5 Å rmsd apart ($D_{DV} < 1.5 \text{ \AA}$), but both are more than 2.0 Å rmsd from the experimental pose ($D_{DE}, D_{VE} > 2.0 \text{ \AA}$) (Consensus False Positive)	0 (0%)	0 (0%)	1 (9%)	0 (0%)	1 (2%)
a Vina pose is within 1.5 Å rmsd of the experimental pose ($D_{VE} < 1.5 \text{ \AA}$), but it is not the closest pose to the Dockeye pose (Dockeye False Positive)	6 (25%)	1 (10%)	1 (9%)	0 (0%)	8 (15%)

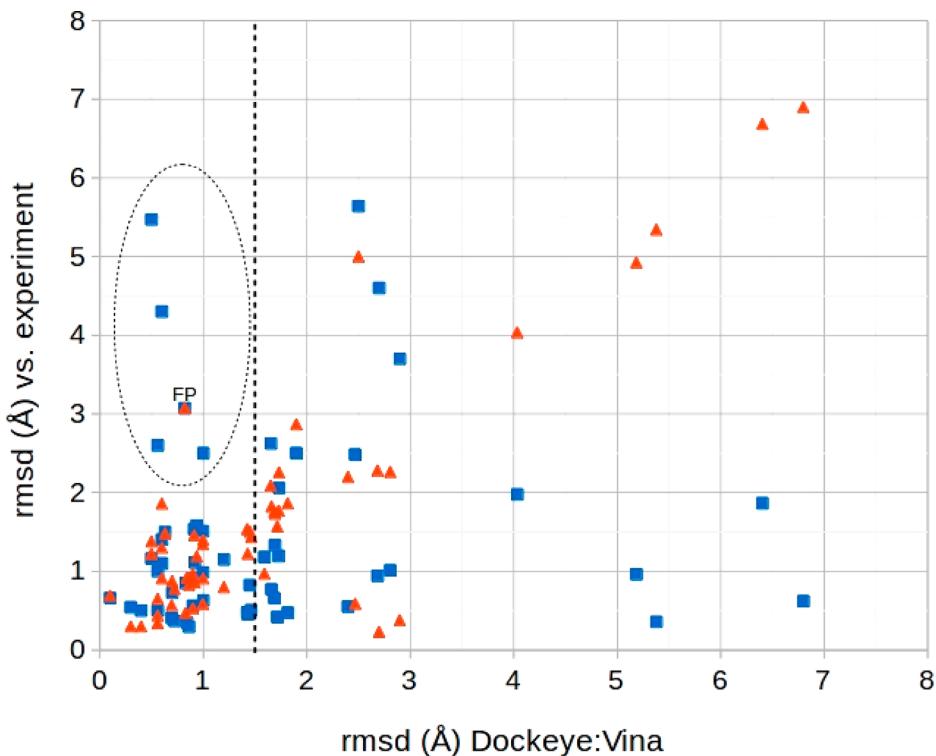


Figure 9. Plot of rmsd error in the placement (blue: Dockeye D_{DE} ; red: best Vina placement D_{VE}) against the distance between Dockeye placement and the closest Vina placement, D_{DV} . The dotted line indicates the cutoff used for good agreement ($< 1.5 \text{ \AA}$). The cluster of points in the lower left rectangle ($D_{DV} < 1.5 \text{ \AA}$, rmsd error $< 2 \text{ \AA}$) represents accurate true positive cases. Dockeye points inside the ellipse fall into the “False Negative” category: they do not agree with a very good Vina placement, except for the one “False Positive” case (labeled FP) where Vina and Dockeye agree but are both wrong.

by 180° are comparable in terms of steric constraints and vdw interactions.

Dockeye addresses this by explicitly scoring all four 180° flipped versions of every placement. The comparison of docking with and without this feature indicates that the 180° error would be made about 2–3 times more frequently without conformer flipping (data not shown). Complexes 1C3X, 1D6N, and 1F0Q are three examples where conformer flipping reduced the rmsd error by about 2–4 Å. Of course, this only addresses limitations in the search for the correct placement. An error because the incorrectly flipped placement has a better score, e.g., in complexes 1BKY and 1ENU, will remain, as this arises from the inaccuracy of the scoring function.

It is emphasized again that the aim here of using the two docking approaches, interactive and automated, was not to see which, if either, was better but to see if the combination of the two could improve docking. Since in a real application the true

placement would not be known, we asked whether the agreement between the Dockeye and Vina results would be predictive of a better agreement with the experiment. We also considered the converse question: if, when Dockeye and Vina disagree, are they both wrong or is just one wrong? Additionally, Vina docking produces a set of placements (the default is 9) ranked by the Vina scoring function. The problem is how to deal with this set. Could Dockeye be used to pick the most accurate placement, especially when Vina judged the placements very close in quality? To these ends, the docking results in Tables 1–4 were classified into the following cases:

- (1) Vina True Positive: The top scored Vina pose is the closest pose to the experiment
- (2) Vina False Positive: The top scored Vina pose is not the one closest to the experiment (In other words, this category contains good poses that would be missed in

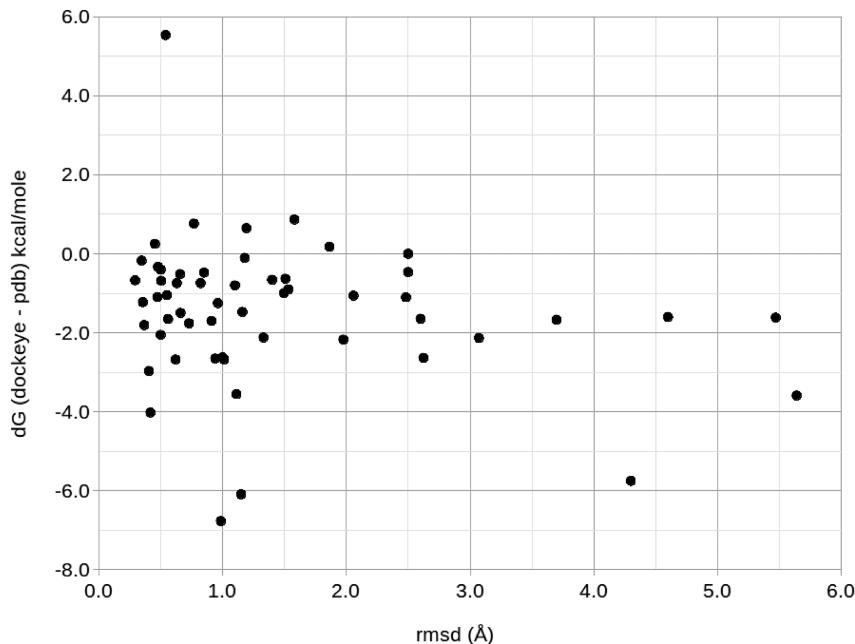


Figure 10. Difference in Dockeye's scoring function: best docked placement minus experimental placement, plotted against rmsd error in docked placement, D_{DE} . Mean difference is -1.4 kcal/mol .

- the absence of Dockeye if one was guided just by the Vina scoring function.)
- (3) Consensus True Positive: The Vina pose closest to the Dockeye pose is also closest to the experimental pose
 - (4) Consensus False Positive: The closest Vina and Dockeye poses are less than 1.5 Å rmsd apart ($D_{DV} < 1.5 \text{ Å}$), but both are more than 2.0 Å rmsd from the experimental pose (D_{DE} and $D_{VE} > 2.0 \text{ Å}$)
 - (5) Dockeye False Positive: A Vina pose is within 1.5 Å rmsd of the experimental pose ($D_{VE} < 1.5 \text{ Å}$), but this pose is not the closest one to the Dockeye pose

An additional category would be “True Negative” cases where Dockeye disagreed significantly with all nine Vina placements and all ten placements were wrong. However, there were not enough occurrences to make this condition a good indicator of anything, other than the obvious fact that at least one of the methods made an error.

The categorization of the docking results into cases 1–5 is summarized in **Table 5** and illustrated in **Figure 9**.

Vina, as is typical for automated docking programs, outputs a certain number of “best” candidate poses it finds, ranked by its scoring function. The Vina default used here is to output the nine best candidates (except for the ligand in entry 1CNX, which has the largest number of rotatable bonds, 12, for which the 20 best candidate poses were output). In a real application where one does not know the experimental pose, the only basis for choice is the automated docking program’s scoring function. If one chose on the basis of solely this, from **Table 5**, Vina’s unassisted true positive rate for pose discovery is about 48% (28 of 58 cases, row 1, **Table 5**). If the pose closest to the experimental pose is not the top ranked of the Vina’s poses, then the highest scored pose is not the best pose; the best pose is missed, and an inferior one is chosen. Thus, one has a false positive. From the results in **Tables 1–4**, 30 of the 58 cases fall into this category (row 2, **Table 5**). Therefore, we estimate the probability of this kind of false positive rate, automated docking with Vina alone and taking the top scored

pose, at about 52%. Of course, this estimate is likely reasonable only for ligands with size and composition similar to those in **Tables 1–4**. However, we can significantly increase the true positive rate and, thus, reduce the false positive rate by choosing the Vina pose closest to the Dockeye pose, regardless of its Vina scored rank. This Consensus True Positive rate, as we call it, is now 83%, nearly twice as high as the unassisted rate (row 3, **Table 5**). Of these, when the Vina and Dockeye poses agree well ($D_{DV} < 1.5 \text{ Å}$), in 34 of the cases, 29 (88%) of both the docked placements are within 2.0 Å of the experiment (D_{DE} and $D_{VE} < 2.0 \text{ Å}$), which we call accurate Consensus True Positives, **Figure 9**.

Cases where a pair of Vina and Dockeye poses agreed well, within 1.5 Å rmsd, but were both seriously in error ($> 2 \text{ Å}$ rmsd from the experiment), which we call a Consensus False Positive, were rare. There was one example among 58 cases or 2% (row 4, **Table 5**). Finally, there are cases where the agreement with Dockeye selects an inferior Vina pose, while missing a good Vina pose (within 1.5 Å rmsd of the experiment), which we call a Dockeye False Positive. This occurred in 8 of the 58 cases or 15% of the time (row 5, **Table 5**). Put another way, the data in **Table 5** show that, using the criteria of similarity between the two independent docking methods, the overall false positive rate was decreased to less than 17%, about a factor of 3.

In order to implement interactive, lag free docking on standard workstations and laptops, Dockeye uses one of the simplest and minimalist scoring or energy functions possible: A simple sum of pairwise interactions with 3 terms (repulsive and attractive vdw terms and a Coulombic electrostatic term) with a rather short cutoff. In contrast, Vina uses a knowledge-based scoring function.²⁹ While the goal of this work was not to test and refine a scoring function for docking, it is useful to compare the values of the Dockeye scoring function for the experimental placement and the best docked placement (**Figure 10**). Over the 55 complexes studied, the value of the Dockeye scoring function for its best placement was on average

–1.4 kcal/mol lower than that for the experimental placement, while the median rmsd error in the placement was about 1 Å.

Since presumably the experimental placement has a lower free energy than any other placement, this means, not surprisingly, that the scoring function is in error, to an amount of at least 1.4 kcal/mol. (It could be larger since we cannot know for certain if there are placements of lower Dockeye energy still that were not discovered.) There is, however, no discernible trend for placements with a larger rmsd error in atomic positions to have a larger error in the energy scoring function. On the other hand, for only six of the complexes did the experimental placement have a lower Dockeye energy than the best discovered placement. Since in these six cases the docking failed to discover a demonstrably better placement, the search was not extensive enough for these six cases. However, this is not a large percentage of the total cases examined, from which we can infer that the phase space exploration is in general quite good.

■ DISCUSSION AND CONCLUSIONS

Reliable placement discovery requires two things: sufficient exploration of the placement/ligand conformation phase space and an accurate scoring function. The results of this comparative study of an automated vs interactive docking, shown in Tables 1–4 and Figures 8 and 9, discussed above, indicate that the phase space exploration is sufficiently thorough in most of the 58 cases examined. The two quite different methods turn up closely related placements in many of the 58 cases, and when they do, it is a reliable indicator of the accuracy of the placement/conformation vs the experimental complex structure. The results for the most flexible ligands (with 3 or more rotatable bonds) are as good as those for the less flexible and rigid ligands, which also argues that the exploration of the ligand conformational space is sufficient. The scoring function is widely acknowledged to be a limiting factor for accurate docking. Ideally, the perfect scoring function would reproduce the experimental binding free energy (ΔG_{bind}) given the experimental ligand–protein structure and, by extension, would give the actual (although not measurable) ΔG_{bind} for any other given placement or conformation, so that the experimental placement by definition would be the one with the lowest free energy. Docking would then be a matter of finding the lowest energy placement/conformation. Even the most realistic and computationally intensive methods for calculating ΔG_{bind} have only partial success and are far too expensive for docking. Thus, most docking programs use very reduced models that can be implemented in the real time needed by the docking algorithm. In the application described here, we used one of the simplest and computationally minimal models possible: a simple sum of pairwise interactions with a rather short cutoff. Nevertheless, it appears that, by combining the results from two different docking approaches, each with different, albeit rather simplistic scoring functions (the Dockeye scoring function is physics based while the Vina AutoDock scoring function is knowledge based²⁹), improved results can be obtained compared to a single approach. We also reiterate that the Dockeye software is not tied to any particular scoring function. This function can easily be modified in the open source software by the user, the main limitation being the computation resources required for lag-free docking. The utility of Dockeye resides, rather, in the rapid and effective presentation of graphical cues to guide interactive docking. While our judgment that this is done in an

intuitively accessible way is somewhat subjective, it is supported by the fact that testing of Dockeye with good results was performed by two of the authors who had none to minimal prior experience with molecular graphics docking. We expect the synergy of Dockeye and automated docking software in the hands of less naive users to be even greater.

■ ASSOCIATED CONTENT

SI Supporting Information

The Supporting Information is available free of charge at <https://pubs.acs.org/doi/10.1021/acs.jcim.0c01485>.

Key to protein and ligand names (PDF)

Video demo of interactive docking (MP4)

■ AUTHOR INFORMATION

Corresponding Author

Kim A. Sharp — Department of Biochemistry and Biophysics, Perelman School of Medicine at the University of Pennsylvania, Philadelphia, Pennsylvania 19104-6073, United States;  orcid.org/0000-0002-0338-0382; Email: sharpk@upenn.edu

Authors

Saravana G. Baskaran — Platelet Biogenesis, Watertown, Massachusetts 02472, United States

Thayne P. Sharp — Harriton High School, Bryn Mawr, Pennsylvania 19010, United States

Complete contact information is available at:

<https://pubs.acs.org/10.1021/acs.jcim.0c01485>

Notes

The authors declare no competing financial interest.

■ ACKNOWLEDGMENTS

K.A.S. acknowledges support from the E.R. Johnson Research Foundation and NIH instrumentation grant number S10OD023592 for computational support.

■ ABBREVIATIONS AND SYMBOLS

ADT, Autodock tools; CADD, computer aided drug development; CPU, computer processing unit; GPU, graphics processing unit; D_{DE} , root-mean-square interatomic distance between the Dockeye (D) and experimental (E) placements; D_{VE} , root-mean-square interatomic distance between the VINA (V) and experimental (E) placements; D_{DV} , root-mean-square interatomic distance between the Dockeye (D) and VINA (V) placements; PDB, Protein Data Bank; R/T/I, rotational, translational, and internal degrees of freedom; rmsd, root-mean-square distance; R_{cut} , distance cutoff for protein–ligand interactions; vdw, van der Waals; ΔG_{bind} , Gibbs free energy of binding; $|\Delta r|$, magnitude of the translation vector between two placements; $\Delta\theta$, rotation angle between two placements

■ REFERENCES

- (1) Klebe, G. Virtual Screening: Scope and Limitations. In *Virtual Screening in Drug Discovery*; Alvarez, J., Shoichet, B., Eds.; CRC Press, 2005; Chapter 1.
- (2) Perola, E.; Walters, W. P.; Charifson, P. S. An Analysis of Critical Factors Affecting Docking and Scoring. In *Virtual Screening in Drug Discovery*; Alvarez, J., Shoichet, B., Eds.; CRC Press, 2005; Chapter 3.
- (3) Wang, J.; Morin, P.; Wang, W.; Kollman, P. A. Use of MM-PBSA in Reproducing the Binding Free Energies to HIV-1 RT of TIBO Derivatives. *J. Am. Chem. Soc.* 2001, 123, 5221–5230.

- (4) Jones-Hertzog, D.; Jorgensen, W. L. Binding Affinities for Sulfonamide Inhibitors with Human Thrombin Using Monte Carlo Simulations with a Linear Response Method. *J. Med. Chem.* **1997**, *40*, 1539.
- (5) Sulea, T.; Purisima, E. O. The Solvated Interaction Energy Method for Scoring Binding Affinities. *Methods Mol. Biol.* **2012**, *819*, 295–303.
- (6) Zhou, R.; Friesner, R. A.; Ghosh, A.; Rizzo, R. C.; Jorgensen, W. L.; Levy, R. M. New Linear Interaction Method for Binding Affinity Calculations Using a Continuum Solvent Model. *J. Phys. Chem. B* **2001**, *105*, 10388–10397.
- (7) Abel, R.; Wang, L.; Mobley, D. L.; Friesner, R. A. A Critical Review of Validation, Blind Testing, and Real-World Use of Alchemical Protein-Ligand Binding Free Energy Calculations. *Curr. Top. Med. Chem.* **2017**, *17*, 2577–2585.
- (8) Hansson, T.; Marelius, J.; Aqvist, J. Ligand Binding Affinity Prediction by Linear Interaction Energy Methods. *J. Comput.-Aided Mol. Des.* **1998**, *12*, 27–35.
- (9) Mobley, D.; Liu, S.; Cerutti, D.; Swope, W.; Rice, J. Alchemical Prediction of Hydration Free Energies for SAMPL. *J. Comput.-Aided Mol. Des.* **2012**, *26*, 551–562.
- (10) Wang, L.; Wu, Y.; Deng, Y.; Kim, B.; Pierce, L.; Krilov, G.; Lupyan, D.; Robinson, S.; Dahlgren, M. K.; Greenwood, J.; Romero, D. L.; Masse, C.; Knight, J. L.; Steinbrecher, T.; Beuming, T.; Damm, W.; Harder, E.; Sherman, W.; Brewer, M.; Wester, R.; Murcko, M.; Frye, L.; Farid, R.; Lin, T.; Mobley, D. L.; Jorgensen, W. L.; Berne, B. J.; Friesner, R. A.; Abel, R. 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. *J. Am. Chem. Soc.* **2015**, *137*, 2695–2703.
- (11) Head, M. S.; Given, J. A.; Gilson, M. K. Mining Minima: Direct Computation of Conformational Free Energy. *J. Phys. Chem. A* **1997**, *101*, 1609–1618.
- (12) Chen, W.; Chang, C.-E.; Gilson, M. K. Calculation of Cyclodextrin Binding Affinities: Energy, Entropy, and Implications for Drug Design. *Biophys. J.* **2004**, *87*, 3035–3049.
- (13) Kuntz, I. D.; Blaney, J. M.; Oatley, S. J.; Langridge, R.; Ferrin, T. E. A Geometric Approach to Macromolecule-Ligand Interactions. *J. Mol. Biol.* **1982**, *161*, 269–288.
- (14) DesJarlais, R. L.; Sheridan, R. P.; Seibel, G. L.; Dixon, J. S.; Kuntz, I. D.; Venkataraghavan, R. Using Shape Complementarity as an Initial Screen in Designing Ligands for a Receptor Binding Site of Known Three-Dimensional Structure. *J. Med. Chem.* **1988**, *31*, 722–729.
- (15) Helmer-Citterich, M.; Tramontano, A. PUZZLE: A New Method for Automated Protein Docking Based on Surface Shape Complementarity. *J. Mol. Biol.* **1994**, *235*, 1021–1031.
- (16) Tsai, J.; Gerstein, M.; Levitt, M. Keeping the Shape but Changing the Charges: A Simulation Study of Urea and Its Iso-Steric Analogs. *J. Chem. Phys.* **1996**, *104*, 9417–9430.
- (17) Das, S.; Krein, M. P.; Breneman, C. M. Binding Affinity Prediction with Property-Encoded Shape Distribution Signatures. *J. Chem. Inf. Model.* **2010**, *50*, 298–308.
- (18) Nicholls, A.; McGaughey, G. B.; Sheridan, R. P.; Good, A. C.; Warren, G.; Mathieu, M.; Muchmore, S. W.; Brown, S. P.; Grant, J. A.; Haigh, J. A.; Nevins, N.; Jain, A. N.; Kelley, B. Molecular Shape and Medicinal Chemistry: A Perspective. *J. Med. Chem.* **2010**, *53*, 3862–3886.
- (19) Axenopoulos, A.; Daras, P.; Papadopoulos, G. E.; Houstis, E. N. SP-Dock: Protein-Protein Docking Using Shape and Physicochemical Complementarity. *IEEE/ACM Trans. Comput. Biol. Bioinf.* **2013**, *10*, 135–150.
- (20) Williams, G. Shape Complementarity at Protein Interfaces via Global Docking Optimisation. *J. Mol. Graphics Modell.* **2018**, *84*, 69–73.
- (21) Yan, Y.; Huang, S.-Y. Pushing the Accuracy Limit of Shape Complementarity for Protein-Protein Docking. *BMC Bioinf.* **2019**, *20*, 696.
- (22) Gilson, M.; Honig, B. The Inclusion of Electrostatic Hydration Energies in Molecular Mechanics Calculations. *J. Comput.-Aided Mol. Des.* **1991**, *5*, 5–20.
- (23) Novotny, J.; Sharp, K. Electrostatic Fields in Antibodies and Antibody/Antigen Complexes. *Prog. Biophys. Mol. Biol.* **1992**, *58*, 203–224.
- (24) Dean, P. M.; Chau, P.-L.; Barakat, M. T. Development of Quantitative Methods for Studying Electrostatic Complementarity in Molecular Recognition and Drug Design. *J. Mol. Struct.: THEOCHEM* **1992**, *256*, 75–89.
- (25) Chau, P.-L.; Dean, P. M. Electrostatic Complementarity between Proteins and Ligands. 1. Charge Disposition, Dielectric and Interface Effects. *J. Comput.-Aided Mol. Des.* **1994**, *8*, 513–525.
- (26) Sobolev, V.; Wade, R. C.; Vriend, G.; Edelman, M. Molecular Docking Using Surface Complementarity. *Proteins: Struct., Funct., Genet.* **1996**, *25*, 120–129.
- (27) Walls, P. H.; Sternberg, M. New Algorithm to Model Protein Protein Recognition Based On Surface Complementarity: Applications to Antibody Antigen Docking. *J. Mol. Biol.* **1992**, *228*, 277–297.
- (28) Lorber, D. M.; Shoichet, B. K. Flexible Ligand Docking Using Conformational Ensembles. *Protein Sci.* **1998**, *7*, 938–950.
- (29) Trott, O.; Olson, A. J. AutoDock Vina: Improving the Speed and Accuracy of Docking with a New Scoring Function, Efficient Optimization and Multithreading. *J. Comput. Chem.* **2010**, *31*, 455–461.
- (30) Repasky, M. P.; Shelley, M.; Friesner, R. A. Flexible Ligand Docking with Glide. *Curr. Protoc. Bioinf.* **2007**, *18*, 8.12.1–8.12.36.
- (31) Ewing, T. J. A.; Makino, S.; Skillman, A. G.; Kuntz, I. D. DOCK 4.0: Search Strategies for Automated Molecular Docking of Flexible Molecule Databases. *J. Comput.-Aided Mol. Des.* **2001**, *15*, 411–428.
- (32) McGann, M. FRED Pose Prediction and Virtual Screening Accuracy. *J. Chem. Inf. Model.* **2011**, *51*, 578–596.
- (33) Coconati, S.; Forli, S.; Perryman, A. L.; Harris, R.; Goodsell, D. S.; Olson, A. J. Virtual Screening with AutoDock: Theory and Practice. *Expert Opin. Drug Discovery* **2010**, *5*, 597–607.
- (34) David, L.; Luo, R.; Gilson, M. K. Ligand-Receptor Docking with the Mining Minima Optimizer. *J. Comput.-Aided Mol. Des.* **2001**, *15*, 157–171.
- (35) Pagadala, N. S.; Syed, K.; Tuszyński, J. Software for Molecular Docking: A Review. *Biophys. Rev.* **2017**, *9*, 91–102.
- (36) Ouh-young, M.; Pique, M.; Hughes, J.; Srinivasan, N.; Brooks, F. P. Using a Manipulator for Force Display in Molecular Docking. In *Proceedings: 1988 IEEE International Conference on Robotics and Automation*; 1988; Vol. 3; pp 1824–1829; DOI: [10.1109/ROBOT.1988.12330](https://doi.org/10.1109/ROBOT.1988.12330).
- (37) Lavery, R. DNA Flexibility Under Control: The Jumna Algorithm and Its Application to BZ Junctions. In *Unusual DNA Structures*; Wells, R. D., Harvey, S. C., Eds.; Springer: New York, NY, 1988.
- (38) Brooks, F. P.; Ouh-Young, M.; Batter, J.; Kilpatrick, P. J. Project GROPEHaptic Displays for Scientific Visualization. *ACM SIGGRAPH* **1990**, *24*, 1 DOI: [10.1145/97880.97899](https://doi.org/10.1145/97880.97899).
- (39) Lai-Yuen, S. K.; Lee, Y.-S. Interactive Computer-Aided Design for Molecular Docking and Assembly. *Comput.-Aided Des. Appl.* **2006**, *3*, 701–709.
- (40) Nagata, H.; Mizushima, H.; Tanaka, H. Concept and Prototype of Protein-Ligand Docking Simulator with Force Feedback Technology. *Bioinformatics* **2002**, *18*, 140–146.
- (41) Gilson, M. K.; Sharp, K. A.; Honig, B. H. Calculating the Electrostatic Potential of Molecules in Solution: Method and Error Assessment. *J. Comput. Chem.* **1988**, *9*, 327–335.
- (42) Sharp, K. A.; Nicholls, A.; Honig, B. *DelPhi: A Macromolecular Electrostatics Software Package*; Dept. of Biochemistry and Molecular Biophysics, Columbia University: New York, 1990.
- (43) Bondi, A. *Molecular Crystals, Liquids and Glasses*; John Wiley and Sons: New York, 1968.
- (44) Sanner, M. F. Python: A Programming Language for Software Integration and Development. *J. Mol. Graphics Modell.* **1999**, *17*, 57–61.

- (45) Huang, J.; MacKerell, A. D. CHARMM36 All-Atom Additive Protein Force Field: Validation Based on Comparison to NMR Data. *J. Comput. Chem.* **2013**, *34*, 2135–2145.
- (46) Case, D.; Pearlman, D.; Caldwell, J. W.; Cheatham, T.; Wang, J.; Ross, C.; Simmerling, T.; Darden, T.; Merz, K.; Stanton, A.; Chenn, J.; Vincent, M.; Crowley, V.; Crowley, V.; Tsui, V.; Gohlke, H.; Radmer, R.; Duan, Y.; Pitera, J.; Massova, I.; Seibel, G.; Singh, U.; Weiner, P.; Kollman, P. A. AMBER 7; UCSF, 2002.
- (47) Kearsley, S. On the Orthogonal Transformation Used for Structure Comparison. *Acta Crystallogr., Sect. A: Found. Crystallogr.* **1989**, *A45*, 208–210.
- (48) Connolly, M. Solvent-Accessible Surfaces of Proteins and Nucleic Acids. *Science* **1983**, *221*, 709–713.
- (49) Wang, R.; Fang, X.; Lu, Y.; Yang, C.-Y.; Wang, S. The PDBbind Database: Methodologies and updates. *J. Med. Chem.* **2005**, *48*, 4111–4119.
- (50) Wang, R.; Fang, X.; Lu, Y.; Wang, S. The PDBbind Database: Collection of Binding Affinities for Protein-Ligand Complexes with Known Three-Dimensional Structures. *J. Med. Chem.* **2004**, *47*, 2977–2980.
- (51) Besler, B. H.; Merz, K. M.; Kollman, P. A. Atomic Charges Derived from Semiempirical Methods. *J. Comput. Chem.* **1990**, *11*, 431–439.
- (52) Mezei, M.; Mehrotra, P. K.; Beveridge, D. L. Monte Carlo Determination of the Free Energy and Internal Energy of Hydration for the Ala Di peptide at 25 °C. *J. Am. Chem. Soc.* **1985**, *107*, 2239.
- (53) Jean-Charles, A.; Nicholls, A.; Sharp, K.; Honig, B.; Tempczyk, A.; Hendrickson, T.; Still, C. Electrostatic Contributions to Solvation Energies: Comparison of Free Energy Perturbation and Continuum Calculations. *J. Am. Chem. Soc.* **1991**, *113*, 1454–1455.