HHS Public Access

Author manuscript

J Chem Inf Model. Author manuscript; available in PMC 2016 April 27.

Published in final edited form as:

J Chem Inf Model. 2015 April 27; 55(4): 872–881. doi:10.1021/ci500380a.

The Impact of Side-chain Packing on Protein Docking Refinement

Mohammad Moghadasi[†], **Hanieh Mirzaei**[†], **Artem Mamonov**[‡], **Pirooz Vakili**[¶], **Sandor Vajda**[‡], **Ioannis Ch. Paschalidis**[*,§], and **Dima Kozakov**[*,‡]

[†]Division of Systems Engineering & Center for Information and Systems Engineering

[‡]Department of Biomedical Engineering

[¶]Division of Systems Engineering, and Department of Mechanical Engineering

[§]Department of Electrical and Computer Engineering, Division of Systems Engineering, and Department of Biomedical Engineering

Abstract

We study the impact of optimizing side-chain positions in the interface region between two proteins during the process of binding. Mathematically, the problem is similar to side-chain prediction, extensively explored in the process of protein structure prediction. The protein-protein docking application, however, has a number of characteristics that necessitate different algorithmic and implementation choices. In this work, we implement a distributed approximate algorithm that can be implemented on multi-processor architectures and enables trading off accuracy with running speed. We report computational results on benchmarks of enzyme-inhibitor and other types of complexes, establishing that the side-chain flexibility our algorithm introduces substantially improves the performance of docking protocols. Further, we establish that the inclusion of unbound side-chain conformers in the side-chain positioning problem is critical in these performance improvements.

The prediction of the tertiary structure of proteins is an important problem in computational structural biology with applications in protein structure design, protein association, and homology modeling. In general, side-chains are more flexible than the backbone, and positioning them is a critical component of protein structure prediction[1–3].

It is therefore not surprising that side-chain prediction has received significant attention during the last few decades. Most of the existing literature views the problem as an optimization/search problem over possible side-chain conformations. Several works first attempt to reduce the search space by applying the idea of *Dead-End Elimination (DEE)*, which eliminates all side-chain conformations that cannot possibly be in the optimal solution[4,5]. Lee *et al.*[1] proposed an approach based on a *simulated annealing* search. Lee *et al.*[6] also suggested a similar approach using a *mean-field optimization* search. Roitberg and Elber proposed a method that combined the latter two approaches.[7] Bower *et al.*[8] introduced

*To whom correspondence should be addressed yannisp@bu.edu; midas@bu.edu.

heuristics to search over the space of specific energy functions implemented in the *SCWRL* package. The latest version of the package, *SCWRL4.0*[9], implemented a *tree decomposition* algorithm[10] which is an exact method using dynamic programming. Side-chain prediction has also been formulated as a mathematical programming problem. Specifically, it has been formulated as an *Integer Linear Programming (ILP)* problem[11,12] and several strategies have been proposed to solve it[12,13]. A *semi-definite programming* relaxation of the ILP problem was developed by Chazelle *et al.*[14] and a *second-order cone programming* relaxation was proposed by Kingsford *et al.*[11]. The primary application of the work we surveyed above is in side-chain prediction in the context of protein folding. In fact, some works consider the joint folding and side-chain prediction problem; see Loose et al.[15] Side-chain prediction algorithms attempt to resolve the uncertainty in the position of side chains (especially the ones on the protein surface) that computational or experimental determination of the tertiary structure of proteins leave unresolved.

Side-chain prediction is, however, extremely important in the context of protein-protein association. As the two partner proteins approach each other, side-chains in the interface region between the proteins tend to re-orient so as to avoid steric clashes and facilitate the process of binding. Capturing this effect algorithmically has the potential to enhance docking protocols and it is the main motivation behind the work in this paper.

This problem of side-chain prediction in the course of protein docking has a number of characteristics –distinct from its application to folding– that enable the development of specialized and more efficient algorithms. First, side-chains need to be repacked many times in the process of iterative docking algorithms, and hence speed is a primary consideration. Second, accuracy does not have to be extremely high. In fact, it was shown by Wolfson et al.[16] that docking results can be substantially improved even by a very approximate adjustment of side-chains that removes steric clashes. Third, the unbound protein structure provides a good approximation for the bound conformation of many side-chains; it has been shown that over 60% of surface side-chains retain the unbound conformation upon association with the partner protein[17]. Thus, as will be discussed, considering the unbound conformer as one of the potential states substantially improves the results. Fourth, prediction is performed in the presence of a second protein that, in many cases, significantly reduces the potential joint conformations. In this light, the approach we have developed can be seen as accounting for these special conditions.

More formally, we will consider the so-called problem of *Side-Chain Packing (SCP)* defined as follows: given the unbound structures of the receptor and the ligand, and assuming that the backbones remain rigid, predict the interface side-chain conformations that minimize the overall energy of the complex. SCP has been shown to be NP-hard[18] and inapproximable[14] (i.e., there is no polynomial-time algorithm to obtain solutions that are arbitrarily close to optimal).

Some forms of SCP have already been incorporated in docking procedures[19]. In our docking protocol, first a large set of unbound receptor-ligand conformations are sampled using a rigiddocking technique called PIPER[20]. Low energy conformations are retained for further refinement. Refinement techniques[19,21] iteratively move the ligand while keeping the

receptor fixed in order to minimize an approximate energy function[22]. This iterative search aims to find the rotation and orientation of the ligand which locally minimize the ligand-receptor interaction energy. SCP then becomes a component of energy evaluation for each ligand move.

SCP is a combinatorial problem, assuming that side-chain positions are selected from a discrete set of conformations called *rotamers*[23]. This is generally a good approximation within the framework of required accuracy, particularly because in docking we generally work with relative smooth scoring functions that do not heavily penalize minor steric overlaps. In this work we formulate SCP as a *Maximum Weighted Independent Set (MWIS)* problem on an appropriately defined graph. We have developed[24] a fully distributed algorithm that can output near-optimal solutions. We have established that our algorithm obtains an optimal solution to SCP for a special class of problem instances motivated by the structure of SCP arising in docking[24]. In contrast to the aforementioned related work in the context of folding, our method is based on an approximate algorithm and forgoes optimality since state-of-the-art interaction energy models are also approximate. However, our method is fully distributed and requires only message-passing between neighboring nodes of the graphical model of the SCP problem which will be illustrated in Fig. 1. Distributed algorithms are algorithms designed to run over multiple processors, with no tight centralized control. This is appealing in our docking framework since, as mentioned earlier, one has to solve many instances of SCP in the course of docking two proteins. In some large instances of SCP involving numerous residues in a protein complex, the distributed implementation of our algorithm allows us to position the side-chains with near-optimal accuracy, yet, with an average running time significantly smaller than the state-of-the-art centralized algorithms.

The approach we have developed further enables the user to parametrize the method so as to trade-off the quality of the solution against the running time. In the docking application, and especially in the early stages of docking, we are not looking for the most near-native set of rotamers necessarily, but for a good feasible solution which resolves the steric clashes of the interface. In such cases, the accuracy of the algorithm can be set such that desirable timing constraints are met.

Following an earlier observation[17,25], we test the impact of including the unbound conformations of side-chains in the set of possible conformers. Our study of large benchmarks of enzyme-inhibitor and other types of complexes (as defined in Chen et al.[26]) establishes that this inclusion substantially improves side-chain prediction accuracy and the effectiveness of docking protocols. Essentially, we find that the unbound protein structure contains substantial information about the side-chains of the bound state.

Our discussion thus far suggests that SCP in the process of docking exhibits significant special structure which provides us with a number of algorithmic and implementation choices (e.g., exact vs. approximate, distributed vs. centralized, inclusion of unbound conformers, etc.). In this light, our approach is not directly comparable to existing and well-established side-chain prediction methods we surveyed. Still, we do report results comparing the side-chain prediction accuracy of our approach and that of SCWRL4[9], which is considered the state-of-the-art. Several considerations need to be taken into account when

interpreting such results. First, SCWRL4 is available in binary form and does not include the unbound rotamers. Moreover, it is an exact and centralized algorithm, designed with folding applications in mind, and it does not benefit from a multi-processor environment. Our findings can potentially guide the development of alternative approaches for docking applications, including the adaptation of tools like SCWRL4[9].

The remainder of the paper is organized as follows. Section 1 outlines our method for solving the SCP problem. Computational results on benchmark sets and an extensive discussion are included in Section 2. Section 3 contains some concluding remarks.

1 METHODS

1.1 SCP Formulation

In the context of our docking application, we are only interested in positioning the side-chains located in the interface between the receptor and the ligand. Side-chains buried within the proteins are typically well-packed and non-interface surface side-chains have no significant effect on docking. We fix the position and orientation of the ligand with respect to the receptor and define the *interface residues I* as the set of all receptor and ligand residues whose C_a atom is within a small distance (10 Å) from a C_a atom located on the partner molecule. Let U_i denote the set of rotamers for each residue $i \in I$ and denote by $|I|$ the cardinality –number of elements– of I.

The goal of SCP is to choose one rotamer per residue to minimize the free energy of the complex. Let i_r denote the rotamer selected for residue $i \in I$. Then, the overall energy takes the form:

$$E = E_0 + \Sigma_{i \in I} E(i_r) + \Sigma_{i,j \in I : i<j} E(i_r, j_s), \quad (1)$$

where E_0 is self-energy of the two backbones, $E(i_r)$ is the energy of the interaction between rotamer i_r from residue i and the two backbones including the self-energy of the rotamer i_r, and $E(i_r, j_s)$ is the pairwise interaction energy between the selected rotamers i_r and j_s for $i \neq j$.

We next formulate SCP as an MWIS problem on an appropriately defined graph $G = (V, E)$ whose nodes are assigned weights. The MWIS problem amounts to selecting a set of nodes of G that form an independent set, i.e., no two nodes selected are connected by an edge, of maximal total weight.

We construct G as follows. The node set of the graph, V , consists of two types of nodes: *single-rotamer nodes* and *pair-rotamer nodes*. To each rotamer i_r of each interface residue i we assign a single-rotamer node and to each pair of rotamers (i_r, j_s) from two different residues i and j we assign a pair-rotamer node. We associate an energy value with each node: $E(i_r)$ with single-rotamer nodes and $E(i_r, j_s)$ with pair-rotamer nodes. We also assign to each node a nonnegative weight such that higher weights correspond to nodes with lower energies; this can be done by reversing the sign of the energy values and shifting them equally to become nonnegative. Turning to the edge-set of G , each edge represents a "conflict" between a set of rotamers. The term conflict means that the nodes incident to the

edge correspond to two different rotamers of the same residue, e.g., nodes (i_r, j_s), and (i_rt , j_s). Since in SCP we need to select exactly one rotamer per residue, both nodes connected by an edge cannot be selected. From the construction, it follows that SCP is equivalent to the MWIS problem for graph G . A graphical representation of such modeling is shown in Fig. 1 for a system of two residues i and j which have 3 and 2 rotamers respectively.

1.2 Rotamer Selection

We use the the backbone-dependent rotamer library[23] to derive the initial set of rotamers. In addition to the rotamers listed in the rotamer library, we generate more rotamers by considering the standard deviation value σ_1 (also available in the rotamer library) of the dihedral angle χ_1 for each rotamer. Specifically, we split each rotamer of the library into 3 rotamers with the following first dihedral angles: $\chi_1 - \sigma_1$, χ_1, and $\chi_1 + \sigma_1$. We keep the rest of dihedral angles (χ_2, χ_3 and χ_4), if any, as they are, and assign to each new rotamer a probability equal to 1⁄3 of the original rotamer probability. As discussed in the Introduction, we also add one more conformer from the unbound structure of the protein to the set of rotamers. The set of rotamers gained from the expansion of the original library spans the conformational space of the side-chains better, and gives the algorithm a broader search space to seek the optimal side-chain configuration.

Before solving the MWIS formulation, we run a pre-processing subroutine called *rotamer refinement* which refines the set of rotamers for each residue and excludes any infeasible rotamers from the set. This subroutine consists of two phases. *(i)* First, we find the atomic coordinates of each rotamer and define its distance to the backbone as the nearest distance between its heavy atoms and the backbone heavy atoms. We remove from consideration rotamers whose distance to the backbone is smaller than a predefined threshold. These rotamers form steric clashes with the backbone and cannot belong to the optimal solution. *(ii)* Next, we implement another pre-processing step to reduce the number of the rotamers for each interface residue. We use a *Dead-end Elimination (DEE)* algorithm[5], which is based on a refinement of the elimination criterion known as the *Goldstein* criterion. The idea is as follows: a rotamer i_r from residue i can be eliminated from the set if there exists some other rotamer i_s from the same residue such that

$$E(i_r) - E(i_s) + \sum_{j \neq i} \min_{t \in U_j} \{E(i_r, j_t) - E(i_s, j_t)\} > 0, \quad (2)$$

for some other residue j with a U_j set of rotamers. This indicates a situation in which the "best" conformation that includes $i_r \in U_i$ has larger total energy value compared to the "worst" conformation that includes $i_s \in U_i$. In other words, for any feasible solution of SCP that includes rotamer $i_r \in U_i$, replacing i_r by i_s gives us a new feasible solution with lower total energy. In this case, we can eliminate i_r from U_i. DEE stops when it finds no more rotamers to remove. These pre-processing phases can reduce the size of G drastically, thereby speeding up the process of finding an MWIS without sacrificing optimality.

1.3 Our Distributed Algorithm to Solve MWIS

MWIS is an NP-hard problem. We have developed a two-phase algorithm[24,27,28] to find effective solutions: the first phase solves a relaxation of MWIS and the second phase leverages the relaxed solution to construct an effective MWIS feasible solution. This feasible solution indicates which rotamer to pick for each interface residue.

In the first phase, we employ a stronger relaxation than the standard linear programming relaxation of MWIS. In particular, we introduce constraints on the cliques of the graph that are redundant but make the linear programming relaxation of the integer programming problem tighter. We develop a *Gradient Projection (GP)* method (see[24]) for solving the (linear programming) dual problem of this relaxation. Our algorithm only involves message-passing among adjacent nodes of the graph and uses local information. This message-passing approach allows us to solve the problem in a distributed fashion using multiple processors. As we discussed earlier, benefiting from a multi-processor architecture can be useful due to the many and large problem instances one has to tackle in the course of docking two proteins.

Since we solve a relaxation of MWIS in the first phase of the algorithm, we have to round up the solution to a feasible solution for the original problem. To that end, the second phase of the algorithm consists of a greedy estimation procedure that constructs a feasible MWIS solution based on the solution of the relaxation. Our estimation phase is also distributed and works based on passing messages between the nodes of the problem graph.

Our two-phase algorithm produces a near-optimal solution to the problem and has several parameters (e.g., accuracy of the relaxation phase) that can be tuned to trade-off the accuracy of the final solution against the running time. This is useful in the context of our docking application; for instance, in early phases of the docking protocol a less expensive and less accurate version can be used and the accuracy can be tightened in the final stages of docking.

1.4 Partitioning the Interface Residues

The number of nodes in the graph G increases quadratically with the number of interface residues. This can lead to a very large G which is computationally expensive to handle. To reduce the size of the graphs we have to process, we partition the set of interface residues into non-overlapping clusters based on their interaction energy values. We first compute the interaction energy between each pair of residues in the set. If the interaction energy between a pair of residues is greater than a small threshold ε, we say those two residues are interacting. If, however, two residues are too far away, there would be no interaction energy between them. We consider a subset of the residues as *a cluster* if interactions involving these residues are exclusively confined within the cluster. In this sense, the clusters are non-overlapping and the union of clusters forms the whole interface set.

After partitioning the interface set into several clusters, we solve the SCP problem using the MWIS formulation on each cluster separately and in parallel. Note that since the clusters do not energetically interact, breaking the main SCP problems into smaller subproblems does not change the overall solution, yet speeds up the procedure notably.

Based on our statistical analysis over a docking benchmark set composed of tens of receptorligand complexes with thousands of conformations each, we conclude that a significant portion of the clusters contain only 2 residues. Even though our algorithm is an approximate method in general, due to the special structure of the MWIS graph it does find the exact solution for clusters of size two[24]. For larger clusters, it can find an effective feasible solution which is near-optimal[24].

1.5 Off-grid minimization with an optional SCP step

To study the role of SCP in protein docking we have incorporated our side-chain packing approach into off-grid refinement, where it is typically used.

We have implemented a standard *Monte Carlo Minimization (MCM)*-based off-grid refinement protocol, which is used in many refinement approaches[19,29,30]. Off-grid refinement seeks the lowest energy configuration in the vicinity of the initial conformation. The protocol we use performs iterations each consisting of four main steps. (*I*) In Step 1, the ligand position and orientation with respect to the receptor are slightly (randomly) perturbed. (*II*) In Step 2, we slide the proteins back into contact. (*III*) The 3rd step is where SCP is applied and this step is optional; to assess SCP's role in refinement, we will show results for runs without SCP that leave side-chains to their unbound positions, runs with SCP where the whole interface is re-arranged using the algorithm we presented and the standard rotamer library, and runs with SCP using the standard rotamer library to which we add the unbound side-chain conformers. (*IV*) The final step in each iteration of the refinement protocol locally minimizes the energy of the resulting complex using a rigid-body minimization algorithm[22] and allowing the side-chains to slightly move to off-rotamer positions in order to alleviate potential steric clashes. After these four steps are performed, we have a new candidate complex. We decide either to accept or reject this candidate based on the *Metropolis criterion*, namely, if the total energy of the candidate complex is lower than the energy of the complex in the beginning of the iteration, we accept the candidate; otherwise, we accept the candidate with a probability that is inversely exponentially proportional to the energy difference. If the candidate is accepted, then it becomes the complex used to initialize the next iteration; otherwise, we discard the candidate complex and start the next iteration with exactly the same complex we had in the beginning of the current iteration.

1.6 Refinement set generation

To study the effect of off-grid minimization with SCP, we have generated sets of near-native structures using a soft rigid-body approach[20]. For our study set we have used enzyme-inhibitors and other types of complexes from the protein docking benchmark[26]. The following steps were performed for each of the enzyme-inhibitor complexes: (1) We systematically sampled mutual receptor-ligand orientations using an FFT-based approach (PIPER[20]) and obtained 70, 000 lowest energy structures. (2) The 1,000 lowest scoring structures were clustered[31] using a greedy algorithm and the clusters were ranked based on their size (a larger cluster corresponds to higher rank). (3) The highest ranking cluster whose center has a *Root Mean Square Deviation (RMSD)* of all atom positions under 10 Å from the native was selected for refinement. The top 1, 000 lowest energy structures out of the 70, 000 generated at Step 1, which are also within 12 Å RMSD from the selected cluster center,

were selected as the refinement set. A similar protocol was used for other types of complexes, with the exception that clusters were chosen from three FFT sampling runs, each with different weights in the energy function. Details are described in Kozakov et al.[32]. We have used all complexes in the protein docking benchmark[26] for which PIPER[20] was able to produce at least 50 solutions within 5 Å RMSD to the native. Our study set consists of 35 cases of enzyme-inhibitors, and 34 cases of other types of protein complexes.

1.7 Energy Function

For the energy function terms referenced thus far, we have used a state-of-the-art high-accuracy docking energy potential, which combines force-field and knowledge-based energy terms[19,29,33]. In particular, interaction energies are computed as a weighted sum (*w*'s are the corresponding weights):

$$E=w_{VDW}E_{VDW}+w_{SOL}E_{SOL}+w_{DARS}E_{DARS}+w_{COUL}E_{COUL}+w_{HB}E_{HB}+w_{RP}E_{RP},$$

where E_{VDW} is the Lennard-Jones potential, E_{SOL} is an implicit solvation term[34], E_{COUL} is the Coulomb potential, E_{HB} is a knowledge-based hydrogen bonding term, and E_{RP} is a statistical energy term associated with a specific selection of rotamers from the backbone-dependent rotamer library[23]. E_{DARS} is a structure-based intermolecular potential derived from the non-redundant database of native protein-protein complexes using a novel DARS (Decoys as Reference State)[35] reference set. The DARS reference set is formed by generating a large decoy set of docked conformations based only on a shape-complementarity scoring function; we compute the potential by observing the frequency of interactions in these decoys.

In order to calculate E_{RP}, we need to know the probability p^{iu} of each rotamer i_u, which can be approximated by the fraction of time that amino acid residue *i* is found in rotamer *u* in a large dataset. These probabilities are given in the rotamer library. The statistical energy value of such a rotamer is given by $-\log(p^{iu})/p_{i0}$, thus, the more frequent a rotamer, the lower the energy assigned to it. The weights in the energy function are chosen according to the selections in Gray et al.[19].

2 Results and Discussion

2.1 Accuracy of Side-Chain Positioning

We use SCP in predicting the bound-state side-chain conformations of an *unbound* receptor-ligand complex. To assess the accuracy of our algorithm, we test it against a benchmark set consisting of 48 unbound *enzyme-inhibitor (EI)* and 67 *other (OT)* types of protein complexes, and compare our predictions to the native-state conformers which are observed using experimental techniques. We also compare the accuracy of our algorithm with that of the SCWRL4.0 package[23], which, as we commented earlier, is the state-of-the-art side-chain prediction tool. We refer the reader to our earlier discussion on the differences between SCWRL4.0 and our approach and on how the results should be interpreted.

We use standard criteria to evaluate our side-chain prediction approach.[9,17] The first criterion called χ_1 is based on the difference between the first dihedral angle (χ_1) of the set of residues in the predicted structure and the native structure. The second criterion called χ_{1+2} is based on the differences between the first two dihedral angles (χ_1 and χ_2) of the residues in the predicted structure and the corresponding dihedral angles in the native structure. For the χ_1 criterion, an accurate prediction of a residue occurs when the χ_1 angle of a predicted residue is within 40 degrees of its native-state value. For the χ_{1+2} criterion, the prediction of a residue is considered accurate when both the χ_1 and the χ_2 angles of the predicted structure are within 40 degrees of their native-state values. Although the 40 degrees deviation may appear to be large, it is the size of the error considered in standard criteria used for evaluating side-chain prediction algorithms. In addition, as already mentioned, side-chain prediction generally requires relatively limited accuracy in applications to docking.

To show the effect of including the unbound conformer of the side-chains in side-chain prediction, we consider two different cases: (*i*) solving the SCP problem without including the unbound conformers (-UB), and (*ii*) solving the SCP problem with unbound conformers (UB). We compare the overall packing results in the absence and in the presence of the side-chains' unbound conformations to show how the inclusion of the unbound conformers in the rotamer-set can affect the side-chain prediction results. We also provide the SCWRL4.0 predictions to determine the accuracy of our algorithm in comparison with that method. As mentioned in the Introduction, in SCWRL4.0 the unbound side-chains are not considered as possible rotameric states of the residues.

For each complex, we run each algorithm over exactly the same interface set of residues obtained from the *unbound* structure of the complex. We report the number of the interface residues whose predicted conformation is considered accurate based on the χ_1 and χ_{1+2} criteria.

Detailed results are in Fig. 2. We provide the side-chain prediction accuracy of the aforementioned methods for the two different types of protein structures (EI and OT) separately. In the last row of the table, we compare the performance of these methods over the full benchmark by calculating the percentage of all interface residues which are predicted within the accuracy range. A couple of observations are in order. First our method produces slightly less accurate results compared to SCWRL4.0 when the unbound side-chain conformations are not included in the rotamer set. This is, essentially, the small price to pay for an approximate algorithm (vs. the exact approach of SCWRL4.0) which, however, has a number of characteristics that are useful in docking applications (distributed, scalable, tunable speed-accuracy trade-off). The second, and important, observation is that the inclusion of the unbound rotamers improves the accuracy of the predictions. This shows that the unbound structure of proteins carries substantial information about their native docked structure, hence, considering them in the side-chain prediction methods is of great importance in docking applications.

2.2 SCP as a Protein Docking Component

As discussed earlier, our main motivation for this work is to apply SCP in protein-docking refinement protocols. Next, we analyze the effectiveness of our SCP algorithm when we use it as a component of our protein docking refinement procedure. We report on the impact of SCP in the overall performance of the off-grid optimization refinement procedure, and, more specifically, in increasing the number of near-native predictions.

For this purpose, we refine the PIPER[20] outputs using three different modes of the off-grid optimization refinement protocol outlined in Sec. 1.5: (*i*) *REF-SCP*, when the conformations are refined without employing the SCP algorithm, (*ii*) *REF+SCP-UNB* and (*iii*) *REF+SCP +UNB*, when the conformations are refined by the off-grid optimization procedure which uses SCP as a component of energy evaluation without and with, respectively, considering the unbound side-chain conformers in SCP.

For each mode, we calculate the RMSD of each predicted conformation in the set from the native structure. A prediction is considered “accurate” when this RMSD is below 5 Å from the native. The table in Fig. 3 as well as Figs. 4 and 5 report the number of accurate predictions in the refinement set (see Sec. 1.5) for 35 EI and 34 OT complexes. The first column of the tables in Fig. 3 lists the PDB code of the complex. The second column reports the number of accurate conformations (within 5 Å RMSD from the native) out of the top 1, 000 PIPER outputs in the refinement set. These conformations are the input to the refinement stage. The three following columns specify the number of accurate refined conformations generated by the three different modes of off-grid optimization described above – denoted as R-SP, R+SP-UB and R+SP+UB, respectively. The last two rows of each EI and OT table report the total number of accurate predictions over all complexes and the percentage improvement over PIPER. The latter metric is computed by averaging over all complexes the per-complex percentage improvement and it reflects a view of performance which is not biased by the number of accurate complexes for each refinement set. The results show that adding side-chain packing and including the unbound conformers can improve the overall refinement performance by increasing the number of accurate predictions.

Next, we present two other figures for the EI and OT protein benchmarks. In each figure, we plot three curves that indicate the increase/decrease in the number of PIPER accurate conformations using the three settings of the refinement procedure described above. The green, blue and red curves indicate the REF-SCP, REF+SCP-UNB and REF+SCP+UNB mode, respectively. As an example, consider the protein complex *1yvb* which has 376 accurate conformations generated by PIPER as shown in the table of Fig. 3 and is the first protein shown in Fig. 4. The green, blue and red data points show the values of 117, 93 and 117 respectively, for *1yvb*, reflecting the respective gains of the three modes over the PIPER result. The same type of analysis is carried out for the OT benchmark as well, and the results are illustrated in Fig. 5.

As shown in Figs. 4 and 5, in most cases the red curve is superior to the other two curves, indicating that the REF+SCP+UNB method works better than the other two methods. It

follows that the use of SCP including the unbound conformers increases the number of near-native predictions and improves the refinement performance.

2.3 Effect of Parallelization on Running Time

To validate the effect of parallelization on improving the running time of our SCP algorithm, we study how the average running time over the benchmark set of 48 EI and 67 OT unbound proteins (listed in the table of Fig. 2) changes as we increase the number of processors. To get a better sense of the improvement, we categorize the benchmark set based on the size of the interface into two subsets labeled as *Large* and *Small*. The size of the interface refers to the number of interface residues of each protein complex, and is reported in the third column of the table in Fig. 2. The size of the MWIS optimization problem, associated with our SCP algorithm, increases quadratically with the size of the interface. Therefore, the parallel approach is of great importance when it comes to large problem instances. In our setting, the protein complexes with interface size greater then 20 are considered in the “Large” category, and the ones whose interface size is on the range of 20 or less are considered in the “Small” category. We also evaluate the running time over the entire benchmark (labeled as *All* in Fig. 6).

Our results were obtained on a desktop workstation with Intel® Core™ i7-950 Processor (8M Cache, 3.06 GHz, 4.80 GT/s Intel® QPI) and 4 GB of RAM. We report the speedup values in Fig. 6 for the cases of 2-, 4-, 6- and 8-processor runs of the algorithm with respect to the single-processor running time. The speedup value of the *n*-processor setting is computed by dividing the average running time of the algorithm when using *n* processors by the average running time of the single-processor run. Fig. 6 shows that using the multi-processor architecture is generally beneficial in speeding up the packing process, especially for large systems.

3 Conclusion

We considered the problem of side-chain packing in the process of protein-protein docking. Specifically, this is the problem of appropriately positioning side-chains in the interface region between the two proteins. The problem exhibits significant special structure that makes it notably different from the side-chain prediction problem extensively explored in the context of protein folding. These differences, motivated our development of a new approximate but fully distributed approach.

We tested this approach against benchmark sets of enzyme-inhibitor and other types of complexes. We found that the incorporation of side-chain packing in each iteration of protein docking refinement protocols, facilitates the docking process and leads to improved performance. We also established that the inclusion of the unbound conformer as an option in the side-chain packing optimization improves side-chain positioning accuracy and docking performance. The latter, can potentially motivate the adaptation of alternative side-chain prediction approaches. Our side-chain packing software is available under an open source license.

Acknowledgments

Research partially supported by the NIH/NIGMS under grants GM093147 and GM061867, by the NSF under grants CNS-1239021, DBI-1147082 and IIS-1237022, by the ARO under grants W911NF-11-1-0227 and W911NF-12-1-0390, and by the ONR under grant N00014-10-1-0952.

References

(1). Lee C, Subbiah S. Prediction of protein side-chain conformation by packing optimization. J. Mol. Biol. 1991; 217:373–388. [PubMed: 1992168]

(2). Summers N, Karplus M. Construction of side-chains in homology modelling: Application to the C-terminal lobe of rhizopuspepsin. J. Mol. Biol. 1989; 210:785–811. [PubMed: 2693742]

(3). Holm L, Sander C. Database algorithm for generating protein backbone and side-chain coordinates from a Calpha trace: application to model building and detection of coordinate errors. J. Mol. Biol. 1991; 218:183–194. [PubMed: 2002501]

(4). Desmet J, de Maeyer M, Hazes B, Lasters I. The dead-end elimination theorem and its use in protein side-chain positioning. Nature. 1992; 356:539–542. [PubMed: 21488406]

(5). Goldstein R. Efficient rotamer elimination applied to protein side-chains and related spin glasses. Biophys. J. 1994; 66:1335–1340. [PubMed: 8061189]

(6). Lee C. Predicting protein mutant energetics by self-consistent ensemble optimization. J. Mol. Biol. 1994; 25(3):236, 918–939.

(7). Roitberg A, Elber R. Modeling side chains in peptides and proteins: Application of the locally enhanced sampling and the simulated annealing methods to find minimum energy conformations. J. Chem. Phys. 1991; 95:9277–9287.

(8). Bower M, Cohen F, Dunbrack R. Prediction of protein side-chain rotamers from a backbone-dependent rotamer library: a homology modeling tool. J. Mol. Biol. 1997; 267:1268–1282. [PubMed: 9150411]

(9). Krivov G, Shapovalov M, Dunbrack R. Improved prediction of protein side-chain conformations with SCWRL4. Proteins: Struct., Funct., Bioinf. 2009; 77:778–795.

(10). Xu, J. Rapid protein side-chain packing via tree decomposition; Proc. - Annu. Int. Conf. Res. Comput. Mol. Biol. (RECOMB); Cambridge, MA, USA. 2005. p. 423-439.

(11). Kingsford C, Chazelle B, Singh M. Solving and analyzing side-chain positioning problems using linear and integer programming. Bioinformatics. 2005; 21:1028–1036. [PubMed: 15546935]

(12). Eriksson O, Zhou Y, Elofsson A. Side chain-positioning as an integer programming problem. Lect. Notes Comput. Sci. 2001; 21:128–141.

(13). Althaus, E.; Kohlbacher, O.; Lenhof, H.; Muller, P. A combinatorial approach to protein docking with flexible side-chains; Proc. - 4th Conf. Comput. Mol. Biol., ACM; New York, NY. 2000.

(14). Chazelle B, Kingsford C, Singh M. A semidefinite programming approach to side chain positioning with new rounding strategies. INFORMS J. Comput. 2004; 16:380–392.

(15). Loose C, Klepeis J, Floudas C. A new pairwise folding potential based on improved decoy generation and side-chain packing. Proteins: Struct., Funct., Bioinf. 2004; 54:303–314.

(16). Mashiach E, Schneidman-Duhovny D, Andrusier N, Nussinov R, Wolfson H. FireDock: a web server for fast interaction refinement in molecular docking. Nucleic Acids Res. 2008; 36:W229–W232. [PubMed: 18424796]

(17). Beglov D, Hall D, Brenke R, Shapovalov M, Dunbrack R, Kozakov D, Vajda S. Minimal ensembles of side chain conformers for modeling proteinÐprotein interactions. Proteins: Struct., Funct., Bioinf. 2012; 802:591–601.

(18). Pierce N, Winfree E. Protein design is NP-hard. Protein Eng., Des. Sel. 2002; 15:779–782.

(19). Gray J, Moughon S, Wang C, Schueler-Furman O, Kuhlman B, Rohl C, Baker D. Protein-protein docking with simultaneous optimization of rigid-body displacement and side-chain conformations. J. Mol. Biol. 2003; 331:281–299. [PubMed: 12875852]

(20). Kozakov D, Brenke R, Comeau S, Vajda S. PIPER: An FFT-based protein docking program with pairwise potentials. Proteins: Struct., Funct., Bioinf. 2006; 65:392–406.

(21). Shen Y, Paschalidis IC, Vakili P, Vajda S. Protein Docking by the Underestimation of Free Energy Funnels in the Space of Encounter Complexes. PLoS Comput. Biol. 2008; 4:10. e1000191.

(22). Mirzaei H, Beglov D, Paschalidis IC, Vajda S, Vakili P, Kozakov D. Rigid body energy minimization on manifolds for molecular docking. J. Chem. Theory Comput. 2012; 8:4374–4380. [PubMed: 23382659]

(23). Shapovalov M, Dunbrack R Jr. A smoothed backbone-dependent rotamer library for proteins derived from adaptive kernel density estimates and regressions. Structure. 2011; 19:844–858. [PubMed: 21645855]

(24). Moghadasi, M.; Kozakov, D.; Vakili, P.; Vajda, S.; Paschalidis, IC. A new distributed algorithm for side-chain positioning in the process of protein docking; Proc. - 52nd IEEE Conf. Decision Control (CDC); Florence, Italy. 2013. p. 739-744.

(25). Kirys T, Ruvinsky AM, Tuzikov AV, Vakser IA. Correlation analysis of the side-chains conformational distribution in bound and unbound proteins. BMC Bioinf. 2012; 13:236.

(26). Chen R, Mintseris J, Janin J, Weng Z. A protein-protein docking benchmark. Proteins: Struct., Funct., Bioinf. 2003; 52:88–91.

(27). Moghadasi, M.; Kozakov, D.; Mamonov, A.; Vakili, P.; Vajda, S.; Paschalidis, IC. A message passing approach to side chain positioning with applications in protein docking refinement; Proc. - 51st IEEE Conf. Decision Control (CDC); Maui, Hawaii. 2012. p. 2310-2315.

(28). Paschalidis IC, Huang F, Lai W. A Message-Passing Algorithm for Wireless Network Scheduling. IEEE/ACM Trans. Networking. 2014 in print.

(29). Andrusier N, Nussinov R, Wolfson HJ. FireDock: fast interaction refinement in molecular docking. Proteins: Struct., Funct., Bioinf. 2007; 69:139–59.

(30). Kozakov D, Schueler-Furman O, Vajda S. Discrimination of near-native structures in protein-protein docking by testing the stability of local minima. Proteins: Struct., Funct., Bioinf. 2008; 72:993–1004.

(31). Kozakov D, Clodfelter K, Vajda S, Camacho C. Optimal clustering for detecting nearnative conformations in protein docking. Biophys. J. 2005; 89:867–875. [PubMed: 15908573]

(32). Kozakov D, Beglov D, Bohnuud T, Mottarella SE, Xia B, Hall DR, Vajda S. How good is automated protein docking? Proteins: Struct., Funct., Bioinf. 2013; 81:2159–66.

(33). Pierce B, Weng Z. ZRANK: Reranking protein docking predictions with an optimized energy function. Proteins: Struct., Funct., Bioinf. 2007; 67:1078–86.

(34). Schaefer M, Karplus M. A comprehensive analytical treatment of continuum electrostatics. J Phys Chem. 1996; 100:1578–1599.

(35). Chuang G-Y, Kozakov D, Brenke R, Comeau SR, Vajda S. DARS (Decoys As the Reference State) potentials for protein-protein docking. Biophys J. 2008; 95:4217–27. [PubMed: 18676649]

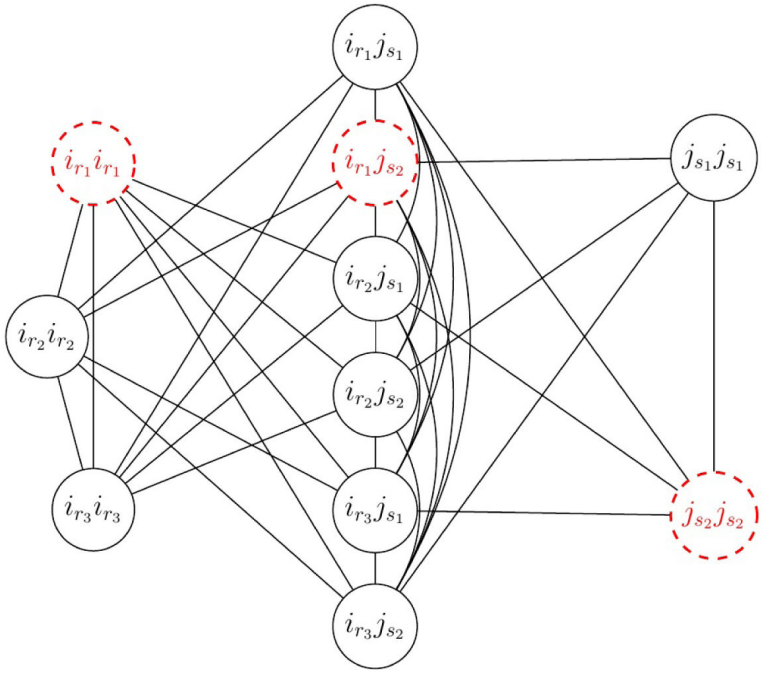

Figure 1.

Construction of the graphical model of a system with 2 residues *i* and *j* with sets of rotamers: $U_i = \{i_{r1}, i_{r2}, i_{r3}\}$ and $U_j = \{j_{s1}, j_{s2}\}$. The optimal set of rotamers of the residues *i* and *j* can be obtained by finding the MWIS of this weighted graph. Let the triple $\{i_{r1}\, i_{r1}, i_{r1}\, j_{s2}, j_{s2}\, j_{s2}\}$ be the MWIS, then the solution to the SCP problem will be rotamers i_{r1} and j_{s2} for residues *i* and *j*, respectively.

PDB	cat	# res	χ_1			χ_{1+2}		
			scwrl	MWIS		scwrl	MWIS	
				-UB	+UB		-UB	+UB
2b42	EI	30	20	14	23	10	9	14
1udi	EI	23	15	16	14	9	11	8
1pxv	EI	21	12	16	17	9	10	12
1n8o	EI	20	15	14	15	12	10	14
1f34	EI	19	11	9	14	7	7	10
2abz	EI	18	10	11	13	8	7	8
1avx	EI	17	13	11	12	11	9	10
1cgi	EI	17	10	12	13	10	8	8
1clv	EI	17	9	11	9	6	7	9
1ml0	EI	17	10	12	14	9	10	10
1bvn	EI	16	9	8	9	6	6	7
1mah	EI	16	13	8	12	10	5	9
2j0t	EI	16	9	7	11	6	5	9
2sic	EI	16	10	8	11	6	5	4
2sni	EI	16	12	11	12	11	4	7
1dfj	EI	15	8	11	12	7	10	11
1ezu	EI	15	10	8	9	5	6	7
1hia	EI	15	9	7	10	5	6	6
1fle	EI	14	5	5	6	5	4	5
1jtg	EI	14	8	10	7	6	6	6
1oc0	EI	14	9	10	8	5	6	4
1r0r	EI	14	10	11	12	9	9	9
1ay7	EI	13	12	9	11	9	7	8
1oyv	EI	13	6	8	10	5	6	8
1bvk	EI	12	6	6	7	3	3	5
1gl1	EI	12	7	8	9	4	7	6
1gxd	EI	12	7	5	4	2	3	2
1ijk	EI	12	10	8	10	7	7	6
1kkl	EI	12	8	6	8	6	5	6
1nw9	EI	12	7	7	9	5	6	5
2mta	EI	11	10	7	8	8	5	7
2uuy	EI	11	8	8	10	6	6	7
1eaw	EI	10	8	9	8	3	6	5
1tmq	EI	10	7	8	8	4	3	2
7cei	EI	10	7	8	7	4	5	5
1e6e	EI	8	7	6	7	6	4	6
1jiw	EI	8	5	6	6	3	5	4
1mlc	EI	8	6	6	7	4	4	5
1oph	EI	7	4	3	5	4	2	4
2o8v	EI	7	5	4	5	3	2	4
1acb	EI	6	4	4	5	2	0	2
1ewy	EI	6	2	3	3	2	3	2
1f6m	EI	6	4	3	2	1	2	1
1yvb	EI	5	3	3	3	3	3	3
2pcc	EI	5	4	4	4	4	3	4
1s1q	EI	4	3	3	2	3	2	1
2oul	EI	4	3	3	3	2	3	3
3sgq	EI	4	2	3	4	2	2	3
1h1v	OT	42	23	22	27	16	17	19
2oor	OT	40	22	25	18	16	18	10
1ffw	OT	29	19	9	13	15	5	7
2g77	OT	29	16	18	15	10	10	11
1h9d	OT	27	16	13	14	11	10	10
2ayo	OT	27	12	15	16	9	13	12
1kxp	OT	26	13	20	17	10	15	13
1r8s	OT	24	13	13	14	10	9	11
2ot3	OT	22	14	16	16	9	12	12
1eer	OT	21	16	14	13	11	9	8
1wql	OT	21	11	9	15	9	7	9

PDB	cat	# res	χ_1			χ_{1+2}		
			scwrl	MWIS		scwrl	MWIS	
				-UB	+UB		-UB	+UB
1i2m	OT	20	12	13	17	10	12	14
1fqj	OT	19	13	12	13	10	5	8
2btf	OT	19	11	8	10	7	4	8
2hrk	OT	19	12	13	15	11	12	15
2z0e	OT	19	15	7	10	13	6	9
1ml0	OT	18	13	12	15	8	11	12
1wdw	OT	18	10	10	13	10	6	10
2b4j	OT	18	7	9	8	5	7	5
2hqs	OT	18	11	13	11	8	11	7
1f51	OT	17	10	10	10	8	6	8
2nz8	OT	17	8	13	14	7	11	12
2vdb	OT	17	12	12	10	10	8	8
1k5d	OT	16	12	7	13	7	5	7
1t6b	OT	16	8	10	12	4	5	8
1b6c	OT	15	12	12	14	11	10	11
1kac	OT	15	7	5	7	6	4	8
2a5t	OT	15	11	7	7	7	5	6
1a2k	OT	14	12	14	14	11	11	11
1grn	OT	14	8	5	8	4	1	3
1k74	OT	14	7	5	7	6	5	7
2j7p	OT	14	7	5	7	5	3	5
1atn	OT	13	9	10	6	6	8	5
1buh	OT	13	7	6	9	5	5	7
1xul	OT	13	7	5	5	7	5	5
3cph	OT	13	10	7	7	5	4	7
2a9k	OT	12	5	4	6	2	2	4
2c01	OT	12	7	7	9	5	4	7
2oza	OT	12	4	2	4	2	1	2
1gla	OT	11	7	7	7	5	5	5
1gpw	OT	11	7	7	7	6	6	6
1j2j	OT	11	9	9	10	7	8	6
1syx	OT	11	6	6	7	3	5	5
1xqs	OT	11	5	9	8	3	8	7
2fju	OT	11	7	8	7	3	2	5
2hle	OT	11	6	5	5	4	3	4
1mq8	OT	10	6	6	8	4	2	6
1ofu	OT	10	7	5	7	4	4	5
2ajf	OT	10	9	8	7	5	5	3
3bp8	OT	10	3	5	9	2	5	9
1efn	OT	9	4	4	6	2	3	5
1klu	OT	9	7	7	7	4	2	3
2cfh	OT	9	5	6	4	4	5	3
1pyh	OT	8	7	4	4	6	4	4
1r1b	OT	8	5	5	4	4	3	4
1rv6	OT	8	1	2	1	1	2	1
2h7v	OT	8	7	6	3	4	4	2
2i9b	OT	8	2	2	3	2	2	1
2oob	OT	8	4	6	6	4	3	5
3d5s	OT	8	6	6	7	3	5	5
1akj	OT	7	3	4	3	2	4	3
1lfd	OT	7	6	4	4	4	3	3
1us7	OT	7	6	6	6	3	1	3
1ak4	OT	6	4	4	6	2	1	4
1azs	OT	4	3	2	3	1	0	1
1jk9	OT	3	1	3	3	1	3	3
1jwh	OT	2	1	1	1	1	0	1
sum		1592	978	942	1040	697	664	754
ACCURACY %			61.4	59.2	65.3	43.8	41.7	47.4

Figure 2.

Comparing SCWRL4.0 and MWIS to native. We compare the performance of sidechain positioning of three modes: (i) **scwrl** shows the prediction accuracy of SCWRL4.0, (ii) **MWIS −UB** denotes the performance of our MWIS algorithm without considering the unbound conformers, and (iii) **MWIS +UB** indicates MWIS performance including the unbound conformers. Moreover, we report the number of the interface residues whose predicted conformation is considered accurate based on the χ_1 and χ_{1+2} criteria. Also, **# res** indicates the number of interface residues for each system.

Enzyme-Inhibitor Benchmark				
	Input	Refinement Output		
PDB	Piper	R-SP	R+SP-UB	R+SP+UB
2b42	106	77	66	91
1udi	163	195	196	197
1n8o	220	283	276	303
1f34	104	124	138	140
2abz	106	103	110	118
1avx	225	239	270	256
1cgi	129	118	127	132
1bvn	210	229	214	242
1mah	271	273	289	298
2j0t	44	61	66	66
2sic	338	364	355	384
2sni	327	353	333	373
1dfj	214	239	259	261
1ezu	119	203	220	221
1fle	128	94	65	95
1jtg	344	417	430	434
1oc0	39	23	29	33
1r0r	382	403	410	422
1ay7	73	82	88	90
1oyv	228	310	295	319
1gl1	169	188	178	205
1ijk	209	224	207	215
2mta	231	204	196	198
2uuy	197	251	198	232
1eaw	222	241	244	249
1tmq	347	390	391	393
7cei	241	314	315	340
1e6e	160	184	189	190
1acb	89	77	73	90
1ewy	62	57	57	57
1yvb	376	493	469	493
2pcc	62	64	58	56
2oul	333	395	403	410
1ppe	518	522	487	558
4cpa	333	343	308	346
Total	7319	8137	8009	8507
Improvement %		9.90	8.43	15.25

Others Benchmark				
	Input	Refinement Output		
PDB	Piper	R-SP	R+SP-UB	R+SP+UB
1ffw	52	54	59	49
2g77	75	72	90	91
2ayo	566	559	519	531
1kxp	396	425	457	502
1wq1	94	82	91	68
1i2m	42	28	67	44
2btf	267	194	232	220
2hrk	154	230	281	265
1ml0	661	695	707	717
1f51	87	106	102	107
1b6c	435	486	479	508
1a2k	299	137	137	162
1grn	234	206	216	229
1k74	478	497	515	497
1akj	183	183	183	189
1buh	144	172	160	166
1gla	108	114	87	109
1gpw	491	515	535	544
1syx	214	162	237	208
1xqs	149	105	98	108
2hle	102	141	162	144
1ofu	573	601	635	617
2cfh	515	549	571	548
1rlb	234	366	352	394
3d5s	666	736	737	745
1azs	718	704	644	688
1jk9	355	367	363	370
1jwh	414	405	281	424
1e96	240	247	252	257
1hel	285	235	259	247
1xd3	334	318	300	323
1z0k	80	87	92	91
1z5y	203	215	173	151
1zhi	113	113	115	112
Total	9961	10106	10188	10425
Improvement %		2.39	4.65	5.54

Figure 3.

We compare three different refinement modes of a refinement algorithm to demonstrate: (i) the effect of side-chain packing on docking refinement, and (ii) the importance of including the unbound conformers. In each case, we report the number of near-native structures (within 5 Å RMSD from the native) amongst the refinement set of size 1, 000. In the table, *R-SP* stands for REF-SCP (refinement without side-chain packing), *R+SP-UB* denotes REF +SCP-UNB (refinement with side-chain packing without unbound conformers) and *R+SP +UB* denotes the REF+SCP+UNB (refinement with side-chain packing and with unbound conformers).

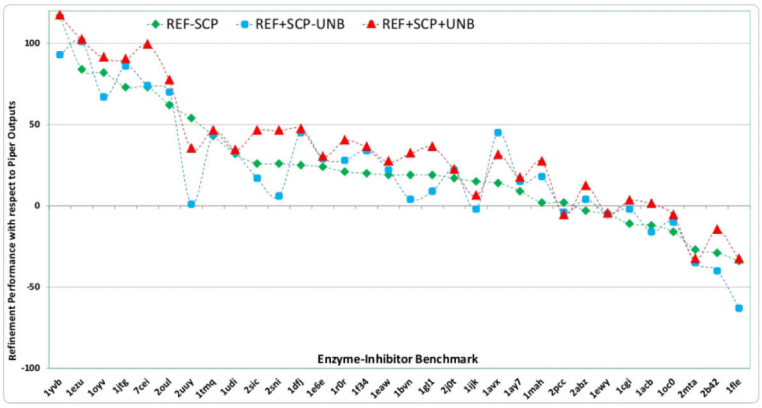

Figure 4.

The effect of different modes of docking on increasing/decreasing the accuracy of PIPER outputs for the EI benchmark. The values on the vertical axis denote the number of additional accurate conformations with respect to PIPER that each refinement mode predicts. The horizontal axis shows the PDB codes of each protein complex. For each mode, these discrete data points are fit to a curve to illustrate the overall performance of each case.

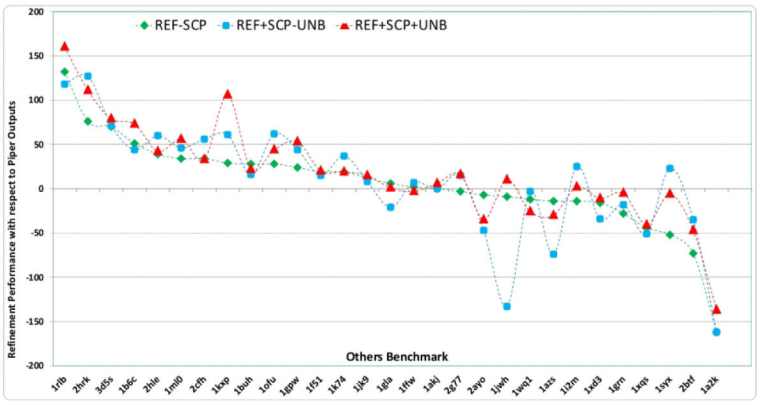

Figure 5.

The effect of different modes of docking on increasing/decreasing the accuracy of PIPER outputs for the OT benchmark. The plots have the same specifications as captioned in Fig. 4.

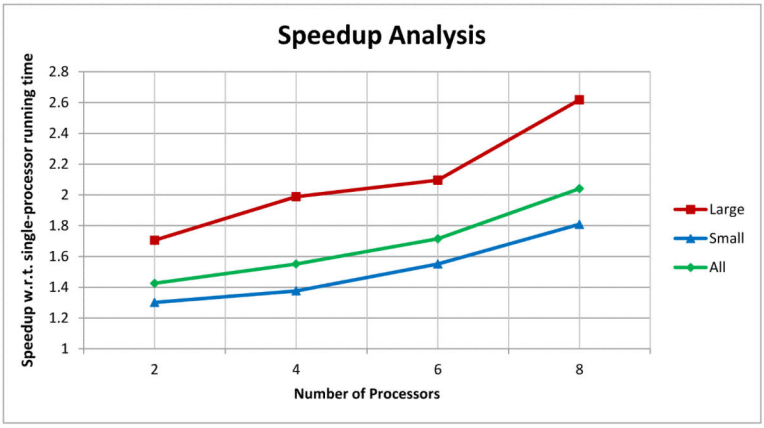

Figure 6.

The speedup with respect to the single-processor run for 2-, 4-, 6- and 8-processor settings. The vertical axis shows the speedup value, and the horizontal axis depicts the number of processors. Different categories of protein ensembles (Large, Small and All) are plotted.