doi:10.1016/j.jmb.2007.07.050 J. Mol. Biol. (2007) 373, 503–519

JMB

Available online at www.sciencedirect.com

ScienceDirect

ELSEVIER

# Protein–Protein Docking with Backbone Flexibility

**Chu Wang, Philip Bradley and David Baker***

*Department of Biochemistry and Howard Hughes Medical Institute, University of Washington, Seattle, WA 98195, USA*

*Received 12 May 2007; received in revised form 7 July 2007; accepted 25 July 2007*
Available online 2 August 2007

Computational protein–protein docking methods currently can create models with atomic accuracy for protein complexes provided that the conformational changes upon association are restricted to the side chains. However, it remains very challenging to account for backbone conformational changes during docking, and most current methods inherently keep monomer backbones rigid for algorithmic simplicity and computational efficiency. Here we present a reformulation of the Rosetta docking method that incorporates explicit backbone flexibility in protein–protein docking. The new method is based on a "fold-tree" representation of the molecular system, which seamlessly integrates internal torsional degrees of freedom and rigid-body degrees of freedom. Problems with internal flexible regions ranging from one or more loops or hinge regions to all of one or both partners can be readily treated using appropriately constructed fold trees. The explicit treatment of backbone flexibility improves both sampling in the vicinity of the native docked conformation and the energetic discrimination between near-native and incorrect models.

Published by Elsevier Ltd.

*Keywords:* protein–protein docking; flexible-backbone docking; loop modeling; conformational change; Monte Carlo minimization

***Edited by M. Sternberg***

## Introduction

Protein–protein interactions play important roles in all cellular activities. Large and complicated protein–protein interaction networks have been mapped in several organisms by methods such as yeast two-hybrid[1] and mass spectrometry,[2] revealing many potentially interacting proteins and complexes. However, the structures of only a small fraction of these potential complexes have been characterized by experimental techniques such as X-ray crystallography, NMR and electron microscopy.[3] Such a gap might be bridged by computational protein–protein docking, which generates a structural model of a protein complex given the structures of its individual components.

Many docking methods treat the interacting proteins as rigid bodies; others allow flexibility only at the side-chain level.[4] The performance of those methods has been extensively evaluated via blind predictions of the structures of more than 20 protein complexes in the Critical Assessment of Predicted Interactions (CAPRI) experiments since 2001.[5–7] Not surprisingly, for the test cases in which significant backbone conformational changes are observed upon formation of the complex, no methods are able to consistently generate models close to the correct docking conformation. Such results clearly indicate the necessity for incorporating protein backbone flexibility in docking methods.

Protein interfaces exhibit considerable plasticity, and various types of backbone conformational changes have been observed upon the binding of two proteins, including loop reconfigurations, hinge movements and other more complex motions.[8] Several promising approaches have been explored to treat backbone flexibility explicitly in protein docking. HADDOCK performs rigid-body docking followed by a molecular dynamics (MD) simulated annealing refinement on backbone and side-chain degrees of freedom, and the added flexibility improves the docking results.[9] Smith *et al.* used a rigid-body docking method, 3D-DOCK, to cross-dock an ensemble of starting structures generated by MD and showed that it sometimes improves the rankings of near-native models.[10] Bastard *et al.*

**Corresponding author.* E-mail address: dabaker@u.washington.edu.

Abbreviations used: CAPRI, Critical Assessment of Predicted Interactions; ICM, Internal Coordinates Modeling; MC, Monte Carlo; MCM, Monte Carlo minimization; MD, molecular dynamics; RF2, release factor 2.

0022-2836/$ - see front matter. Published by Elsevier Ltd.

recently developed a new docking method to account for interface loop movements by including multiple loop copies during the docking search and showed that this can produce models much closer to the crystal complex in comparison with rigid-body docking.[11] A multibody docking approach has been implemented in FlexDock to deal with hinge motions associated with complex formation given the knowledge of hinge regions prior to the docking and the method was able to correctly model large conformational changes occurring in the binding of calmodulin and a target peptide.[12]

Previously, we developed a docking program, RosettaDock, to predict protein–protein interactions.[13] RosettaDock employs a full atomic representation for protein components and allows side-chain conformations of interface residues to change in the course of rigid-body displacement. An enhanced version of RosettaDock with improved side-chain modeling[14] was able to produce models with atomic accuracy for the targets exhibiting limited backbone conformational changes upon binding in CAPRI rounds 4 and 5.[15] However, it failed on the test cases requiring the explicit modeling of backbone flexibility. In RosettaDock, an internal rigid-body coordinate system is used to describe the orientation between the two docking partners, and during the course of sampling the rigid-body space, the proteins have backbone torsion angles fixed while the side chains are free to rotate and sample alternative rotamer conformations.

Recently, a "fold-tree" representation was implemented in Rosetta to improve prediction of β-sheet protein structures.[16] The fold tree allows simultaneous optimization of rigid-body, backbone and side-chain torsional degrees of freedom. The concept of representing a biomolecular system by a "treelike" graph has been implemented in several previous studies. In a pioneering study by Go and colleagues, a fast analytical algorithm to calculate energy function derivatives was derived based on a tree representation of a single polypeptide molecule in which only dihedral torsion angles are considered as variables.[17,18] The program Undertaker developed by the Karplus group implements a similar tree representation for protein structure prediction.[19] The Internal Coordinates Modeling (ICM) suite[20] developed by Abagyan *et al.* uses an "ICM-tree" model (formerly known as "BKS-tree" model) to describe systems in which bond lengths, bond angles and torsion angles can all be treated as independent variables and the spatial orientation between any two rigid-body parts can be encoded by six internal coordinates.[21,22] ICM has been used for protein–protein docking with side-chain flexibility[23] and protein–ligand docking with backbone flexibility.[24] "Treelike" topologies have also been implemented in X-ray and NMR refinement packages such as CNS[25] and XPLOR-NIH,[26] which perform molecular dynamics in internal coordinates to refine protein and complex structures.[27,28]

In this paper we describe the use of the fold-tree representation to enable a wide range of flexible backbone protein–protein docking applications. Within the general kinematic framework of the fold-tree system, the traditional docking rigid-body coordinate frame and internal protein backbone torsional space are seamlessly integrated and all rigid-body and torsional degrees of freedom can be optimized simultaneously. In the Results section, we first provide an overview of the fold tree framework and illustrate how, by combining different fold trees with different sampling strategies, it can be readily applied to a broad range of docking problems with backbone flexibility. We then present results obtained by local-perturbation docking studies using the fold-tree-based method for different types of flexible-backbone docking problems. For docking complexes involving small-scale backbone motions, we show that the flexible-backbone treatment can create more native-like models and improve their energetic discrimination. To tackle docking problems in which large loop conformational changes occur upon complex formation, we incorporate an improved loop modeling algorithm into the fold-tree-based docking method and show that for several protein complexes exhibiting such large motions the explicit treatment of backbone flexibility in loop regions improves the prediction of the structures of complexes over the traditional rigid-body procedure. Finally, we describe the successful modeling of a large loop conformational change in a CAPRI blind prediction challenge.

## Results

### Fold-tree representation

The molecular system (single chain or complex) is represented by a fold tree directed, acyclic, connected graph composed of peptide segments together with long-range connections. This tree is constructed from a simple linear graph in which each residue (vertex) $i$ is connected to residues $i–1$ and $i+1$ via peptide-bond edges within one protein chain, and the first residue of a new chain is connected to the last residue of the previous chain by a pseudo bond edge if there are multiple chains. A new edge is added to the graph for each long-range connection ("jump") that bridges two residues ($j$ and $k$). These edges can represent rigid connections such as those between "stub" residues (fixed template residues next to the terminal residues of each variable loop region) for loop modeling ($j$ and $k$ in the same chain) or fully flexible linkages such as the rigid-body transformation between two docking partners ($j$ and $k$ in different chains). These connections determine how conformational change propagates through the structure. To avoid overconstraining the graph, one peptide-bond edge must be deleted for each new edge added. If the long-range edge is established across two chains, the pseudo bond edge between them is deleted; otherwise, an artificial chain break point is

introduced randomly. An ordering of the graph is defined by selecting a root vertex, e.g., the N terminus of one of the proteins. This graph provides a rule for generating three-dimensional coordinates from backbone torsion angles and a set of rigid-body transformations (one for each long-range connection): starting with an arbitrary location and orientation for the root vertex, traverse the edges of the graph in order, using torsion angles and/or rigid body transformations to build the terminal vertex of each edge given the coordinates of the initial vertex for that edge.

### Examples of fold trees for different structure modeling tasks

The general fold-tree representation integrates torsional freedom and rigid-body freedom together so that they can be optimized simultaneously. It can be defined flexibly to handle a wide variety of molecular structure modeling problems. Most relevant to this paper are docking problems in which backbone motion must be modeled to assemble the complex structure correctly. The following are examples of fold trees for different structure modeling tasks (Fig. 1): (1) Protein structure prediction (Fig. 1a): the fold tree contains only peptide-bond edges all of which are flexible. More complex fold trees can be very useful in predicting structures of β-sheet-containing proteins.[16] (2) Domain assembly[29] with a flexible hinge (Fig. 1b): the fold tree is the same as in (1), but only a subset of peptide-bond edges are allowed to vary. (3) Loop modeling (Fig. 1c): for each local variable region (loop), a rigid long-range edge is added between the loop stub residues and a chain break point ("x" in Fig. 1c) is randomly selected within the loop region to allow folding the loop from both directions. Only the peptide-bond edges (backbone torsions) in the loops are flexible. (4) Rigid-backbone docking (Fig. 1d): a flexible long-range edge is established between the two residues that are closest to the geometrical centers of the docking partners. All the local peptide-bond edges are held rigid. (5) Docking with backbone minimization (Fig. 1e): the fold tree is the same as in (4) except that all the peptide-bond edges in the system are considered flexible. This allows the rigid-body orientation to be optimized while allowing the internal backbone freedom of each individual partner to relax simultaneously. (6) Docking with hinge motion (Fig. 1f): the fold tree is the same as in (4) except that the peptide-bond edges within the defined hinge regions are variable. This allows domains connected by the hinge regions to move relative to one another while the rigid-body orientation of the two partners is optimized. (7) Docking with loop reconstruction or refinement (Fig. 1g): the fold tree is a combination of the docking fold tree and the loop fold tree, and contains both rigid and flexible long-range edges. This allows simultaneous refinement of the docking rigid-body orientation and the loop conformations of the interacting partners.

### Sampling of variable degrees of freedom

The degrees of freedom allowed to vary in the fold tree can be sampled at several different levels in the Rosetta modeling process (Fig. 2). Conformational sampling in Rosetta is first carried out at a low-resolution stage in which fragment insertion and rigid-body Monte Carlo (MC) search are used to rapidly survey the backbone torsional and rigid-

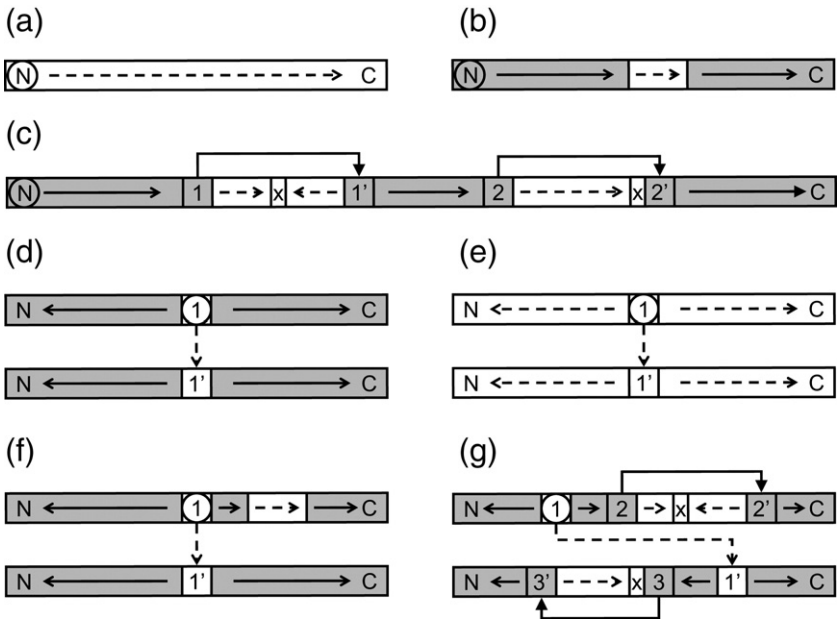


**Fig. 1.** Examples of fold trees for different structure modeling tasks. (a) Protein folding, (b) domain assembly, (c) loop modeling, (d) rigid-backbone docking, (e) docking with backbone minimization or folding and docking, (f) docking with large or small hinge motion, (g) Docking with loop rebuilding or loop refinement. Each individual polypeptide chain is represented by a horizontal box and has its N terminus and C terminus labeled with "N" and "C" respectively. The two residues bridging each long-range jump edge are labeled with the index number of the edge and the residue at the "downstream" end of the edge is labeled with an additional apostrophe. The arrows indicate the directions along which conformational changes are propagated through the structures. Flexible and rigid peptide segments are gray and white, respectively. Flexible and rigid edges of the fold tree are indicated by dashed lines and solid arrows, respectively. "x" indicates the location of an artificial chain break point and "O" indicates the location of the root vertex of the fold tree.

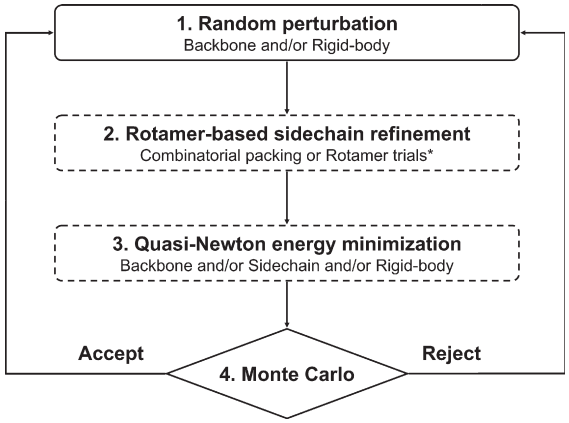


**Fig. 2.** Schematic of Rosetta MCM move. A high-resolution Rosetta MCM move consists of four steps as indicated. Monte Carlo indicates the acceptance or rejection of a move based on the standard Metropolis criterion.[31] Low-resolution Rosetta MC moves include only steps 1 and 4 (as steps 2 and 3 are bordered by dashed lines). *Rotamer Trials[14] can include off-rotamer sampling by minimization.[14]

body space, respectively, to generate starting points for high-resolution all-atom refinement. This step can be skipped if the backbone conformation and/or rigid-body orientation can be approximately determined based on information from homologous structures or experimental constraints. The high-resolution refinement protocol consists of 50–300 Monte Carlo minimization (MCM)[30] moves. Each MCM move (Fig. 2) consists of (1) a random perturbation to one or more degrees of freedom; (2) discrete global optimization of the side-chain degrees of freedom using a rotamer representation; (3) quasi-Newton minimization of the energy with respect to a specified subset of degrees of freedom and (4) acceptance or rejection of the composite move according to the standard Metropolis criterion.[31] Degrees of freedom in which large sampling ranges are desired are explicitly perturbed in step (1) as well as optimized in step (3), while degrees of freedom in which only small "fine tuning" is desired are kept fixed in step (1) and only allowed to vary in the minimization in step (3), which generally introduces only relatively small changes. Thus, within each of the broad clusters of fold trees included in Fig. 1, there are numerous variations that can be chosen based on the problem at hand (Table 1). For example, in Fig. 1f and Table 1, if relatively large hinge variation is expected, perturbation to the backbone torsion angles in the hinge region can be included in step (1) in addition to rigid body perturbation; whereas if only small changes in hinge angles are expected, the random MC perturbation can be restricted to the rigid-body degrees of freedom and the hinge degrees of freedom are only allowed to vary in the subsequent minimization step.

## Applications of the new methodology to protein–protein docking

The ultimate goal of computational protein–protein docking is to assemble the complex structure purely from the structures of unbound partners, taking into account internal conformational changes at both backbone and side-chain levels. The completely general flexible-backbone docking problem is quite formidable because of the very large number of degrees of freedom. While we treat this problem in this paper, in practice the number of degrees of freedom can be reduced based on information on the system under study. In the following sections, we illustrate how fold trees can be readily used for a number of commonly occurring docking scenarios.

**Table 1.** Fold-tree-based sampling strategies for a range of structure modeling problems

| | | Low-resolution MC | High-resolution MCM | |
|---|---|---|---|---|
| Modeling task | Fold tree | Perturbation | Perturbation | Minimization |
| Protein folding<br>Domain assembly<br>Loop modeling | 1a<br>1b<br>1c | Backbone (fragment insertion) | Backbone (small, shear, wobble) | Backbone, side chain |
| Fixed-backbone docking | 1d | Rigid body | Rigid body | Rigid body |
| *Composite type* | | | | |
| Docking with backbone relaxation | 1e | Rigid body | Rigid body | Backbone, side chain, rigid body |
| Folding and docking | 1e | Backbone, rigid body | Backbone, rigid body | Backbone, side chain, rigid body |
| Docking with small hinge motion | 1f | Rigid body | Rigid body | Backbone, side chain, rigid body |
| Docking with large hinge motion | 1f | Backbone, rigid body | Backbone, rigid body | Backbone, side chain, rigid body |
| Docking with loop refinement | 1g | Rigid body | Rigid body | Backbone, side chain, rigid body |
| Docking with loop rebuilding | 1g | Backbone, rigid body | Backbone, rigid body | Backbone, side chain, rigid body |

The flexible regions in the fold trees in Fig. 1 can be varied in several different ways as indicated in Fig. 2. The table indicates the combinations of fold trees with sampling strategies appropriate for different modeling tasks.

### *Rigid-backbone docking*

In the fold tree (Fig. 1d), a flexible long-range edge (jump) is established across two protein partners to represent their relative rigid-body orientation, which is functionally equivalent to the internal rigid-body coordinate frame implemented in the original RosettaDock method,[13] and the backbone of each monomer is held rigid. The residues connected by this edge can be chosen arbitrarily within each partner, but in this study they were selected to be those closest to the partners' geometrical centers. The rigid-body orientation encoded by this jump can be randomly moved, locally perturbed and energetically optimized while side-chain freedom is also allowed to be sampled at both on-rotamer[32] and off-rotamer[14] levels. Local-perturbation docking studies were performed on a set of 25 protein complexes (see Materials and Methods) starting from either backbone conformations from the complex (bound) or backbone conformations from independently solved structures (unbound). As summarized in Table 2, rigid-backbone docking with the bound structures (column 2) is significantly more successful than with the unbound structures (column 3); improving performance with the unbound structures requires the incorporation of backbone flexibility during the docking process to sample the structural changes accompanying binding. In Table 2, we also report, for each complex, the lowest rmsd model obtained in rigid-backbone global docking calculations previously (column 4); most are less than 5 Å, and hence these models could be used as starting structures for flexible-backbone local docking refinement (described below) if no other experimental information is available.

### *Docking with backbone minimization*

The most general way to model flexibility is to allow backbone movement over all residues in both protein structures. Even modest changes can dramatically alter the energy landscape around the native binding mode when all atoms are modeled explicitly. The fold tree is illustrated in Fig. 1e: in addition to the flexible long-range edge between monomers, all the local peptide-bond edges are allowed to vary. If the movements in either or both of the partners are large, then large perturbations must be allowed, and the sampling problem becomes formidable for all but the smallest systems. More tractable is the frequently occurring case in which relatively small changes are expected in the internal structure of the monomers. In such cases, the random perturbation (Fig. 2, step 1) can be restricted to the rigid-body degrees of freedom and then in the energy minimization (Fig. 2, step 3) both the rigid-body and internal torsional degrees of freedom can be simultaneously optimized.

Local-perturbation docking studies were carried out for the same set of 25 complexes using this flexible-backbone approach. Overall, 12 complexes show energy funnels as compared to 9 from rigid-backbone docking (Table 2) and improvements were observed for several cases. For 1BRC, 1GLA, 1TGS, 2SNI, the flexible-backbone protocol recovers the energy funnel lost in the unbound rigid-backbone docking and the near-native models have more favorable interface energies compared to the rigid-backbone control (Table 2 and Fig. 3). For 1DFJ, the energy funnel surrounding the native structure is more evident with more models being pushed into the funnel tip (Table 2). For other complexes in the set, the performance of the flexible-backbone docking method is less good, with either energy funnel being lost, such as 1CSE, or decreased quality for low-energy models, such as 2SIC (Table 2). There are two possible explanations: (1) The unbound backbones are already accurate enough for rigid-body docking and the "prerelaxing" step introduces unnecessary errors (see the example of 1UGH below). (2) The significantly increased number of degrees of freedom complicates the energy landscape considerably and thus it is more difficult for the optimization process to find the global energy minimum without becoming trapped.

Similar overall performance was observed when complexes were ranked by the total interaction energy across the interface ("Interface energy," column 6 in Table 2) or the more physically correct "binding energy" (the total system energy minus the energy of the isolated repacked monomers, column 7 in Table 2). As we will return to in the Discussion section, this highlights a subtle problem with allowing optimization of a large number of degrees of freedom that is associated with noise due to the stochastic nature of the optimization process. Only considering interactions the interface greatly reduces this noise, but is not physically correct and can fail to detect clashes formed within the monomers. The equivalent performance at the current stage of development of the interface energy and the binding energy suggests that the gain in physical accuracy of the binding energy is largely neutralized by the increasing noise far from the interface.

Because the unbound structures were prerelaxed (see Materials and Methods) prior to the docking with backbone minimization, we also conducted traditional rigid-body docking using the same set of prerelaxed structures as an intermediate control in order to examine the effect of the prerelaxing step (column 5 in Table 2). An example of the negative effect is shown by the result of 1UGH: the energy funnel was completely lost, indicating that some backbone errors were introduced during the prerelaxing step. Nevertheless, the loss of the energy funnel was rescued by the additional backbone minimization in the flexible-backbone docking protocol. In contrast, 1DFJ, 1BRC and 1TGS are examples where rigid-body docking using the prerelaxed structures is able to produce a distinct funnel. Interestingly, Smith *et al.* observed similar improvement for 1DFJ when they carried out rigid-body docking starting from an ensemble of MD-

**Table 2.** Results for the fixed-backbone docking and the docking with backbone minimization

| | Bound, PPK, rigid, interface energy | | | | Unbound, PPK, rigid, interface energy | | | | Global rigid ABL | Unbound, RLX, rigid, interface energy | | | | Unbound, RLX, BBmin, interface energy | | | | Unbound, RLX, BBmin, binding energy | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| PDB | N3 | BC | BL | BI | N3 | BC | BL | BI | | N3 | BC | BL | BI | N3 | BC | BL | BI | N3 | BC | BL | BI |
| 1ACB | 3 | 0.846 | 1.08 | 0.24 | 2 | 0.513 | 4.94 | 1.25 | 3.58 | 3 | 0.615 | 2.41 | 1.48 | 2 | 0.487 | 7.79 | 1.64 | 3 | 0.538 | 2.91 | 1.36 |
| 1AHW | 3 | 0.778 | 0.70 | 0.34 | 0 | 0.244 | 6.12 | 1.96 | 4.70 | 2 | 0.422 | 4.54 | 1.81 | 0 | 0.289 | 7.15 | 2.24 | 1 | 0.422 | 6.34 | 1.65 |
| 1AVW | 3 | 0.872 | 0.54 | 0.09 | 3 | 0.489 | 3.69 | 1.17 | 5.02 | 3 | 0.532 | 5.15 | 1.18 | 3 | 0.660 | 4.87 | 0.99 | 2 | 0.447 | 4.80 | 1.29 |
| 1AVZ | 3 | 0.800 | 0.55 | 0.34 | 0 | 0.000 | 11.49 | 5.61 | — | 0 | 0.000 | 13.80 | 6.37 | 0 | 0.048 | 14.95 | 6.29 | 0 | 0.238 | 12.85 | 6.25 |
| 1BRC | 3 | 0.970 | 0.95 | 0.19 | 0 | 0.121 | 9.95 | 3.57 | 3.14 | 3 | 0.576 | 3.87 | 1.31 | 2 | 0.667 | 3.75 | 1.26 | 3 | 0.667 | 3.83 | 1.20 |
| 1BRS | 3 | 0.944 | 0.26 | 0.13 | 3 | 0.667 | 2.95 | 1.18 | 2.82 | 3 | 0.694 | 3.55 | 1.46 | 3 | 0.750 | 3.00 | 1.30 | 3 | 0.750 | 3.00 | 1.30 |
| 1BVK | 1 | 0.759 | 0.62 | 0.24 | 0 | 0.138 | 9.31 | 3.16 | — | 0 | 0.138 | 9.86 | 3.42 | 0 | 0.000 | 11.59 | 6.06 | 0 | 0.000 | 9.85 | 5.91 |
| 1CHO | 3 | 0.800 | 0.62 | 0.25 | 3 | 0.575 | 2.01 | 0.62 | 1.67 | 3 | 0.575 | 1.92 | 0.81 | 3 | 0.625 | 1.88 | 0.74 | 3 | 0.375 | 2.25 | 1.03 |
| 1CSE | 3 | 0.907 | 0.46 | 0.17 | 3 | 0.698 | 2.81 | 0.74 | 2.40 | 0 | 0.000 | 11.87 | 6.68 | 0 | 0.000 | 14.33 | 7.24 | 0 | 0.116 | 5.40 | 3.07 |
| 1DFJ | 3 | 0.651 | 2.33 | 0.88 | 1 | 0.465 | 3.36 | 1.42 | 5.55 | 3 | 0.442 | 3.75 | 1.21 | 3 | 0.488 | 3.66 | 1.53 | 2 | 0.395 | 3.95 | 1.24 |
| 1DQJ | 3 | 0.776 | 0.82 | 0.33 | 0 | 0.020 | 12.25 | 6.27 | — | 0 | 0.000 | 18.91 | 11.56 | 0 | 0.000 | 21.49 | 9.54 | 0 | 0.000 | 18.63 | 11.78 |
| 1FSS | 3 | 0.867 | 0.57 | 0.32 | 0 | 0.244 | 6.70 | 2.40 | 2.35 | 1 | 0.333 | 2.69 | 1.35 | 0 | 0.044 | 12.02 | 4.49 | 0 | 0.178 | 7.61 | 2.87 |
| 1GLA | 1 | 0.714 | 1.28 | 0.42 | 0 | 0.000 | 16.38 | 5.59 | — | 0 | 0.095 | 16.06 | 6.69 | 2 | 0.619 | 2.94 | 1.04 | 0 | 0.190 | 16.23 | 4.97 |
| 1MAH | 3 | 0.744 | 0.60 | 0.25 | 0 | 0.103 | 8.30 | 3.92 | 1.52 | 1 | 0.410 | 3.64 | 1.70 | 0 | 0.308 | 8.61 | 4.14 | 0 | 0.231 | 4.68 | 2.16 |
| 1MDA | 0 | 0.083 | 11.01 | 3.53 | 0 | 0.250 | 5.75 | 1.76 | — | 0 | 0.250 | 8.51 | 2.25 | 0 | 0.000 | 9.72 | 4.21 | 0 | 0.000 | 17.16 | 7.97 |
| 1MLC | 3 | 0.939 | 0.46 | 0.16 | 0 | 0.212 | 10.28 | 3.62 | — | 0 | 0.000 | 22.30 | 9.06 | 0 | 0.152 | 7.81 | 3.20 | 0 | 0.061 | 29.25 | 8.36 |
| 1TGS | 3 | 0.896 | 0.49 | 0.16 | 0 | 0.375 | 5.89 | 2.50 | 2.84 | 3 | 0.438 | 2.62 | 1.49 | 3 | 0.563 | 2.94 | 1.47 | 2 | 0.417 | 3.18 | 1.64 |
| 1UGH | 3 | 0.838 | 0.34 | 0.11 | 3 | 0.486 | 1.91 | 1.11 | 1.70 | 0 | 0.054 | 11.94 | 6.72 | 3 | 0.459 | 3.50 | 1.64 | 0 | 0.189 | 14.76 | 7.19 |
| 1WEJ | 0 | 0.313 | 7.17 | 2.59 | 0 | 0.250 | 10.62 | 3.59 | — | 0 | 0.063 | 10.62 | 5.41 | 0 | 0.094 | 8.55 | 4.54 | 0 | 0.406 | 5.46 | 2.44 |
| 1WQ1 | 3 | 0.818 | 1.40 | 0.48 | 0 | 0.156 | 6.40 | 3.41 | 6.11 | 0 | 0.244 | 6.03 | 2.38 | 0 | 0.222 | 6.18 | 3.21 | 2 | 0.333 | 4.10 | 1.85 |
| 2KAI | 3 | 0.881 | 0.20 | 0.09 | 0 | 0.000 | 16.12 | 10.47 | — | 0 | 0.000 | 27.32 | 12.43 | 0 | 0.000 | 22.37 | 10.77 | 0 | 0.000 | 19.93 | 8.27 |
| 2PCC | 0 | 0.474 | 5.30 | 2.65 | 0 | 0.278 | 10.56 | 3.96 | — | 0 | 0.389 | 5.66 | 2.29 | 0 | 0.222 | 5.44 | 3.28 | 0 | 0.222 | 5.47 | 3.18 |
| 2PTC | 3 | 0.923 | 0.48 | 0.10 | 3 | 0.487 | 3.86 | 1.01 | 3.19 | 3 | 0.513 | 3.97 | 1.02 | 3 | 0.513 | 3.96 | 0.98 | 3 | 0.538 | 3.75 | 0.98 |
| 2SIC | 3 | 0.864 | 1.66 | 0.22 | 3 | 0.773 | 2.17 | 0.41 | 2.91 | 0 | 0.136 | 18.45 | 5.03 | 2 | 0.409 | 6.41 | 1.34 | 1 | 0.386 | 4.61 | 1.29 |
| 2SNI | 3 | 0.786 | 0.53 | 0.18 | 0 | 0.333 | 8.01 | 2.24 | 2.66 | 0 | 0.333 | 9.20 | 2.29 | 3 | 0.738 | 4.26 | 1.26 | 2 | 0.738 | 5.05 | 1.43 |
| Total | 22 | 24 | 22 | 22 | 9 | 11 | 9 | 11 | | 11 | 13 | 10 | 11 | 12 | 13 | 10 | 12 | 12 | 13 | 11 | 12 |

Results for the rigid-backbone docking and the docking with backbone minimization. For each local-perturbation docking run, backbone conformations of the starting structures are taken from either the complex form (bound) or the independently solved structures (unbound); the staring structures were prepared by either the prepacking (PPK) or the prerelaxing (RLX) procedure; the models were generated using either the rigid-backbone (rigid) or the flexible-backbone (BBmin) protocol and afterwards the 5% lowest energy models were ranked based on either "interface energy" (interaction energy across the interface) or "binding energy" (the total energy of the complex model minus the energy when it is pulled apart and fully repacked). N3: the number of models among the top 3 ranking models with at least "medium" accuracy (see Materials and Methods for accuracy classification). BC: the best fraction of native contacts of the top 3 ranking models. BL: the best ligand $C^{\alpha}$ rmsd of the top 3 ranking models. BI: the best interface $C^{\alpha}$ rmsd of the top 3 ranking models. The total number of cases with N3 $>0$, BC $\geq 0.3$, BL $\leq 5.0$ and BI $\leq 2.0$ is counted and N3 $>0$ is used to measure whether an energy funnel exists or not. In addition, for each complex, when data are available, the absolute best ligand rmsd (ABL, i.e., no clustering or ranking) value among the l% lowest energy models from a previously conducted unbound rigid-backbone global docking run is shown.

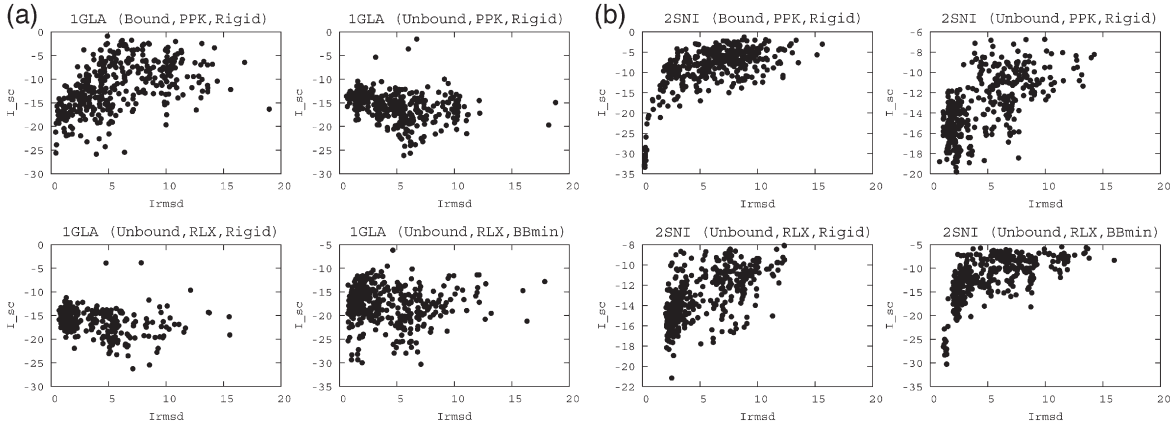


**Fig. 3.** Docking with backbone minimization. Energy *versus* rmsd plots for (a) 1GLA and (b) 2SNI. Upper left: rigid-backbone docking of the bound structures; upper right: rigid-backbone docking of the unbound structures; bottom left: rigid-backbone docking of the prerelaxed structures (intermediate control); bottom right: flexible-backbone docking of the prerelaxed structures.

derived conformations instead of the unbound X-ray structures.[10]

### *Docking with loop minimization*

Backbone changes in protein docking are often focused in interface loop regions, such as complementarity determining regions in antibodies. Even with small loop variations, the native binding mode can have high energy when unbound structures are docked rigidly because of steric clashes. This type of problem is exemplified by two docking complexes: 1T6G and 1MLC (Fig. 4a and b). Both show distinct energy funnels in local-perturbation docking studies using the bound backbones, however, when the unbound backbones are docked, the native binding modes are no longer favorable. For 1T6G, the impact is much more dramatic because the native binding mode has such high energy that the MCM protocol does not produce native-like models.

Fold trees, as illustrated in Fig. 1g, were constructed using predefined loop regions (see Materials and Methods). As in the previous section, the random perturbations were confined to the rigid-body degrees of freedom and the backbone degrees of freedom (in the loops) were varied only in the minimization. In contrast to the docking with full backbone minimization using fold tree 1e, the variation of the loop backbone degrees of freedom with fold tree 1g produces only very local perturbation of the protein backbone coordinates because of the rigid long-range edge between the loop stub residues. Docking with simultaneous optimization of the backbone and side-chain torsion angles in the loop regions using fold tree 1g rescued in both cases the energy funnels (Table 3, Fig. 4a and b). In the 1TG6 case, the lowest energy model has the correct docking arrangement and the backbone optimization yields a loop conformation much more similar to the one in the complex than the one from the unbound structure (Fig. 4c). Examining the structural details of the docking interface reveals that the backbone movement helps relieve the atomic clashes between Leu292 and Pro119 that would otherwise prevent the correct docking arrangement being sampled due to steric clashes in the rigid-backbone docking (Fig. 4d). In the case of 1MLC, the treatment of loop flexibility improves the energetic discrimination of the near-native models from others (Fig. 4b) due to the formation of more favorable interactions across the protein–protein interface (data not shown).

### *Docking with large-scale loop movement*

Docking protein complexes with large-scale backbone movement upon binding presents a significant challenge to traditional rigid-body docking methods because the basic principle of searching for shape complementarity is no longer valid due to dramatic changes in steric properties between unbound and complex structures. This is illustrated by prediction results from the CAPRI blind docking experiment[6,7] and from results on a docking benchmark set.[13] In this section, we first present an improved protocol for modeling loops in monomeric structures using the fold-tree representation. We then describe the integration of this new loop modeling method into a fold-tree-based flexible-backbone docking protocol and show that it improves results for three benchmark test cases that exhibit significant backbone conformational changes between the unbound and bound structures. Finally, we describe a successful blind docking prediction using this approach for the CAPRI target 20, which is a rather challenging case because of the large-scale loop movement upon binding.

*Loop modeling.* We provide an overview of the method here; a more detailed description is provided in the Materials and Methods section. The new method uses the fold-tree representation illustrated in Fig. 1c to reformulate and improve the method developed earlier in our group[33] in which an initial low-resolution sampling stage using fragment inser-

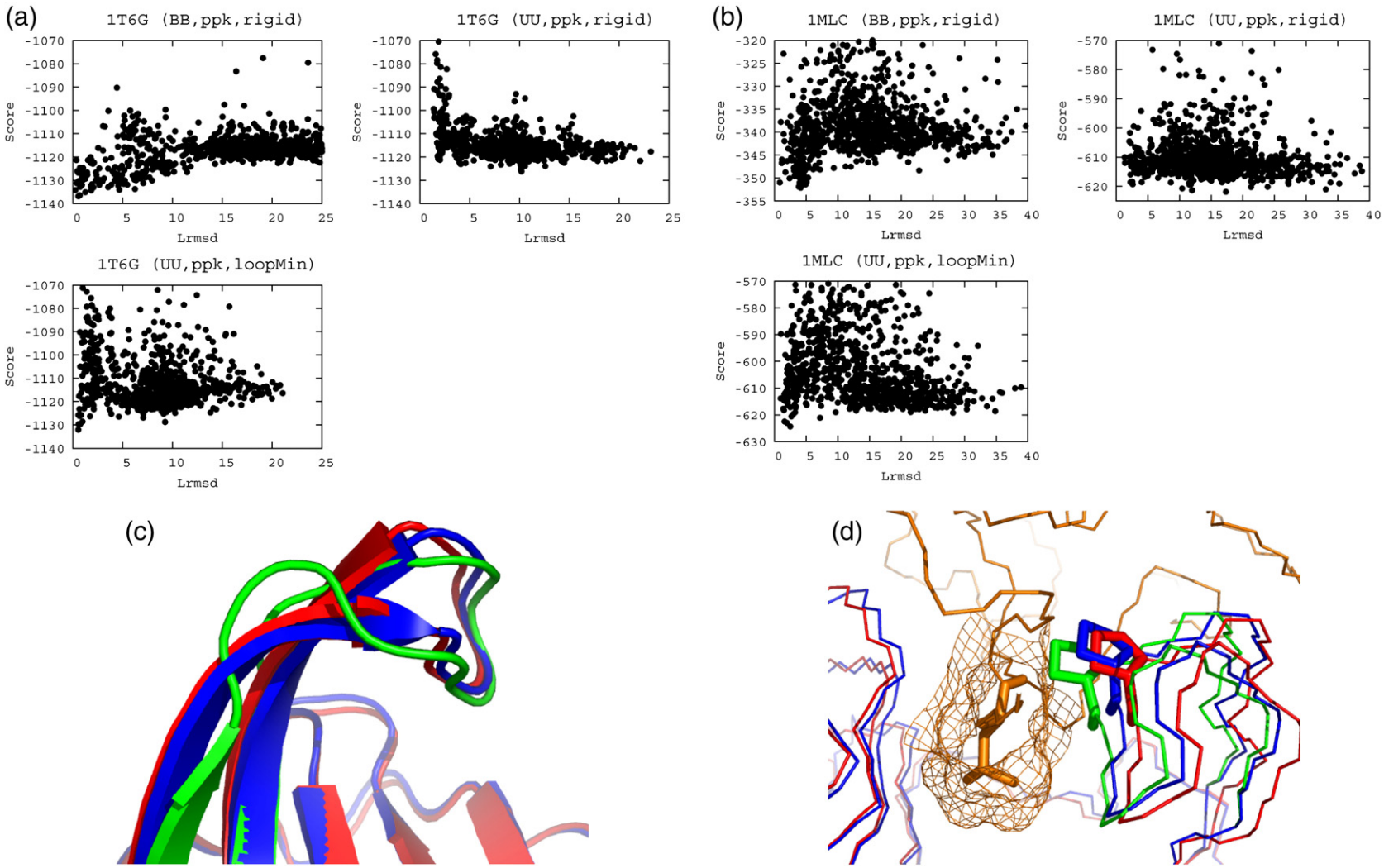


**Fig. 4.** Docking with loop refinement. Energy *versus* rmsd plots for (a) 1T6G and (b) 1MLC. Upper left: rigid-backbone docking of the bound structures; upper right: rigid-backbone docking of the unbound structures; bottom left: flexible-backbone docking with loop minimization of the prepacked structures. (c) Superimposition of the 1T6G acceptor in the native complex, the unbound structure and the Rosetta model. The backbones are drawn in cartoons. (d) Zoom-in view of the interface of 1T6G. Pro119 of the acceptor is shown in sticks and the Leu292 of the ligand is shown in mesh. The backbones are drawn in lines. The native complex is in red and orange. The unbound partner is green. The model is blue.

**Table 3.** Results for docking with loop minimization and remodeling

| Category | Complex | Partner I | Partner II | Bound rigid | | | | Unbound rigid | | | | Unbound flexible control | | | | Unbound flexible | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | N3 | BC | BL | BI | N3 | BC | BL | BI | N3 | BC | BL | BI | N3 | BC | BL | BI |
| Docking with loop minimization | 1MLC | 1MLB (20:34:B, 49:58:B) | 1LZA (68:75:E) | 3 | 0.455 | 3.82 | 0.71 | 0 | 0.000 | 11.97 | 7.96 | — | — | — | — | 2 | 0.394 | 1.58 | 1.15 |
| | 1T6G | 1UKR (113:128:C) | 1T6E | 3 | 0.833 | 0.56 | 0.15 | 0 | 0.000 | 14.75 | 10.12 | — | — | — | — | 3 | 0.646 | 0.80 | 0.55 |
| Docking with loop remodeling | 1BTH | 2HNT (34:41:C, 60:63:C) | 6PTI | 3 | 0.797 | 0.78 | 0.24 | 0 | 0.075 | 7.20 | 4.59 | 2 | 0.453 | 3.77 | 2.65 | 3 | 0.642 | 1.48 | 2.39 |
| | 1FQ1 | 1B39 (145:165:A) | 1FPZ(F) | 3 | 0.812 | 0.61 | 0.17 | 0 | 0.125 | 15.79 | 5.88 | 1 | 0.375 | 3.15 | 3.75 | 2 | 0.438 | 1.48 | 1.97 |
| | 3HHR | 1HGU (42:69:A) | 3HHR(B) | 3 | 0.647 | 0.60 | 0.19 | 0 | 0.176 | 9.74 | 3.18 | 0 | 0.294 | 1.82 | 2.78 | 1 | 0.314 | 3.88 | 2.63 |

For each local-perturbation docking run, backbone conformations of the starting structures are taken from either the complex form (bound) or the independently solved structures (unbound); the models were generated using either the rigid-backbone (rigid) or the flexible-backbone protocol with loop minimization or loop remodeling (flexible). For docking with loop remodeling, a docking run followed by a loop modeling run was performed as an intermediate control (flexible control). Models are ranked by the total energy as described in Materials and Methods. N3, BC, BL and BI are as defined in Table 2.

tions is followed by full-atom refinement. The protocol utilizes MCM moves similar to those outlined in Fig. 2 except that an explicit loop closure step using the cyclic coordinate descent (CCD) algorithm[34] is inserted after the backbone perturbation (step 1) and before the energy minimization (step 3).

We used the benchmark set tested by Rohl *et al.* to evaluate the performance of the new loop modeling method.[33] It contains 40 cases for each of the 8-residue and 12-residue loop prediction categories. For each of these test cases, 1000 models were generated and the global loop rmsd value (see Materials and Methods) of the lowest energy model was examined. As shown in Fig. 5, for both 8-residue and 12-residue loops, the new method yields lower rmsd predictions in more test cases than the original protocol; the improvement is most dramatic for the 12-residue loops. Several aspects of the new method may have contributed to the improved loop modeling results. First, the explicit CCD loop closure procedure removes the need for a large-chain discontinuity penalty term in the energy function (see Materials and Methods) so that the optimization is guided by a more physically realistic energy function. Second, implementation of the fold tree provides more freedom to choose directions for chain building and cut points flexibly, and this can help overcome limitations imposed on the conformational space accessible to the previous method. The new protocol is also simpler than the earlier protocol in that the initial loop conformations are built up on the template using standard Rosetta nine-residue and three-residue fragments and there is no need to generate specialized fragment libraries for each loop modeled.

*Docking with loop remodeling.* In order to treat docking problems with large-scale loop conformational changes, we take advantage of the fold-tree representation to develop an automated flexible-backbone docking method that combines rigid-body docking and loop modeling. In this approach, flexible loops are removed first and then rebuilt after the proteins are docked. The resulting rigid-body orientation and loop conformations are further optimized by alternating between rigid-body docking and loop refinement.

Three complexes classified as "difficult" in the docking benchmark set,[35] 1BTH, 1FQ1 and 3HHR, were tested using this approach. Because of the large-scale backbone conformational changes observed between the unbound and bound structures near the native interface, rigid-backbone local-perturbation docking studies using the bound and unbound backbones yield dramatically different results (Fig. 6a–c). With the new treatment of backbone flexibility, the docking performance is significantly improved (Table 3, Fig. 6). By removing the occluding loops, the rigid-body space in the vicinity of the native binding arrangement can be accessed and subsequent steps of loop modeling and docking/loop refinement help strengthen interactions across the interface so that these near-native models can be identified as lower energy states (Fig. 6). As

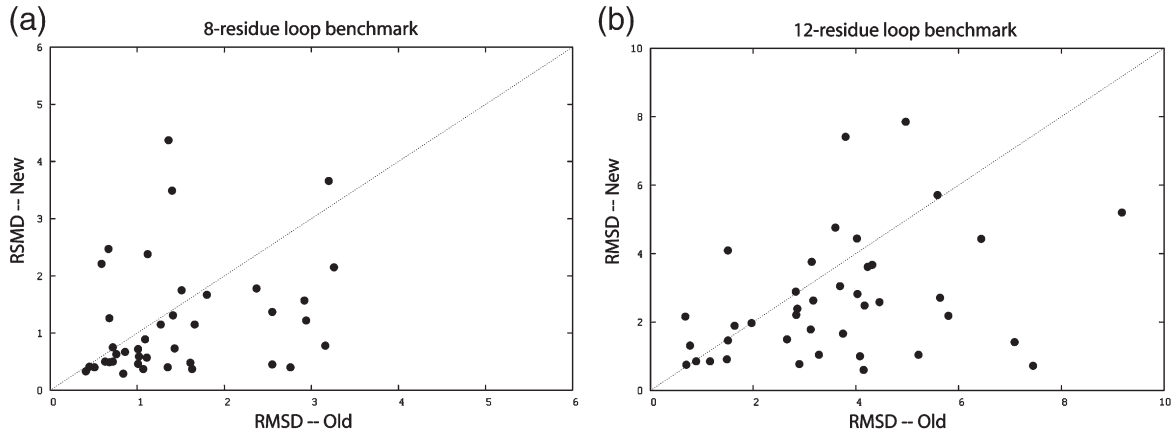


**Fig. 5.** Benchmark results for improved loop modeling method. (a) 8-residue loop benchmark; (b) 12-residue loop benchmark. Each point corresponds to one protein in the benchmark. The $x$ coordinate is the global loop rmsd value of the lowest energy model by the previous method.[33] The $y$ coordinate is the global loop rmsd value of the lowest energy model by the method described in this paper. The line $y=x$ indicates equal performance, and points in the lower triangle represent improved predictions.

an intermediate control, we carried out local-perturbation studies with a hybrid protocol that docks templates without loops using the standard rigid-backbone RosettaDock method and then rebuilds loops with the Rosetta loop modeling method in a sequential manner. Although the hybrid protocol improves sampling the space round the native rigid-body orientation, near-native rigid-body arrangements did not have favorable energies (Table 3, Fig. 6a–c). This result suggests that in the absence of some of the interface loops, the remaining templates can constitute a reasonable interface to be identified by docking, but substantial refinement of the rigid-body orientation and loop conformations is necessary for optimization of the interface, which improves energy-based ranking of model structures.

*Blind docking prediction in CAPRI.* CAPRI is a community-wide double-blind docking experiment aimed at assessing the performance of current protein–protein docking methods.[5] Using the flexible-backbone approach of docking with loop modeling, we were able to make a successful blind docking prediction for CAPRI target 20, a challenging target with significant backbone conformational changes (the performance of our fold-tree-based docking approach in recent rounds of CAPRI is described in detail in a separate manuscript[36]). The docking task was to predict the complex structure of HemK and release factor 2 (RF2).[37] The HemK structure was solved independently by X-ray crystallography,[38] but for RF2, only the structure of its close homologue RF1 was available.[39] HemK was known to methylate the Gln235 in the "Q-loop" of RF2 *in vivo*;[38] however, in the experimentally solved structure of RF1, the equivalent Q-loop region was completely disordered and a computationally remodeled loop conformation was provided in the starting structure.[39] We decided to exclude the flexible Q-loop first and carried out rigid-backbone docking with the remaining template. We then rebuilt the missing Q-loop onto the lowest energy complex models. Based on the "methylation" constraint, the list of docking solutions was filtered and the surviving full-length models underwent a second round of rigid-body docking refinement (loop degrees of freedom were not optimized explicitly, as the new methodology described in this paper was not yet fully developed). Upon release of the native "HemK–RF2" complex structure, our blind docking approach with loop remodeling was found to be successful. The model (with "acceptable" accuracy according to the CAPRI evaluation) has an interface backbone $C^{\alpha}$ rmsd of 2.34 Å with respect to the native complex (Fig. 7a), and after superimposing the aligned template regions of RF2, the backbone $C^{\alpha}$ rmsd for the Q-loop in our model is 4.8 Å in contrast to 11.8 Å for that in the starting structure (Fig. 7b). As illustrated in Fig. 7a, the dramatic conformational change of the Q-loop in RF2 makes it impossible for rigid-backbone docking methods to identify (or even sample) the correct binding mode because it would otherwise clash badly with HemK.

## Discussion

Protein molecules are dynamic and protein–protein association is often accompanied by conformational changes within the monomers. High-resolution prediction of the structures of protein complexes requires modeling such changes explicitly. We have shown previously that when the conformational changes are mainly restricted to side chains, RosettaDock is able to generate atomic-accuracy models due to the explicit treatment of side-chain flexibility, but the method had little success when backbone rearrangements occur, because of the inherent rigid-backbone representation. To address this methodological limitation, we have taken advantage of a fold-tree-based representation to reformulate the RosettaDock method to

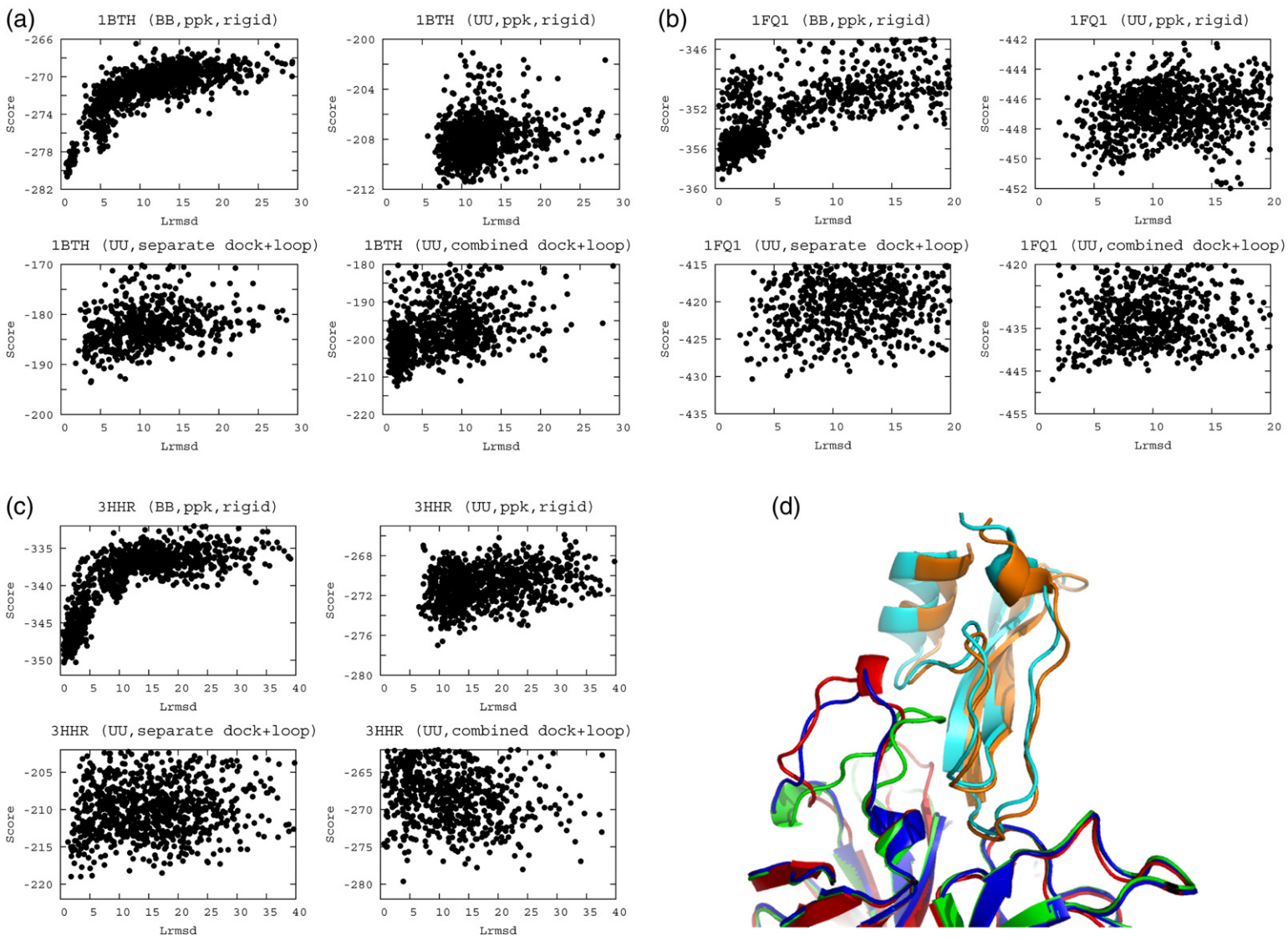


**Fig. 6.** Docking with large-scale loop movement. Energy *versus* rmsd plots for (a) 1BTH, (b) 1FQ1 and (c) 3HHR. Upper left: rigid-backbone docking of the bound structures; upper right: rigid-backbone docking of the unbound structures; bottom left: independent sequential rigid-backbone docking and loop modeling (intermediate control); bottom right: flexible-backbone docking alternating between rigid-backbone docking and loop modeling. (d) Superimposition of the native complex and the Rosetta model for 1BTH. The backbone coordinates are represented as cartoons. The native complex is in red and orange. The unbound partner is in green and cyan. The model is in blue.

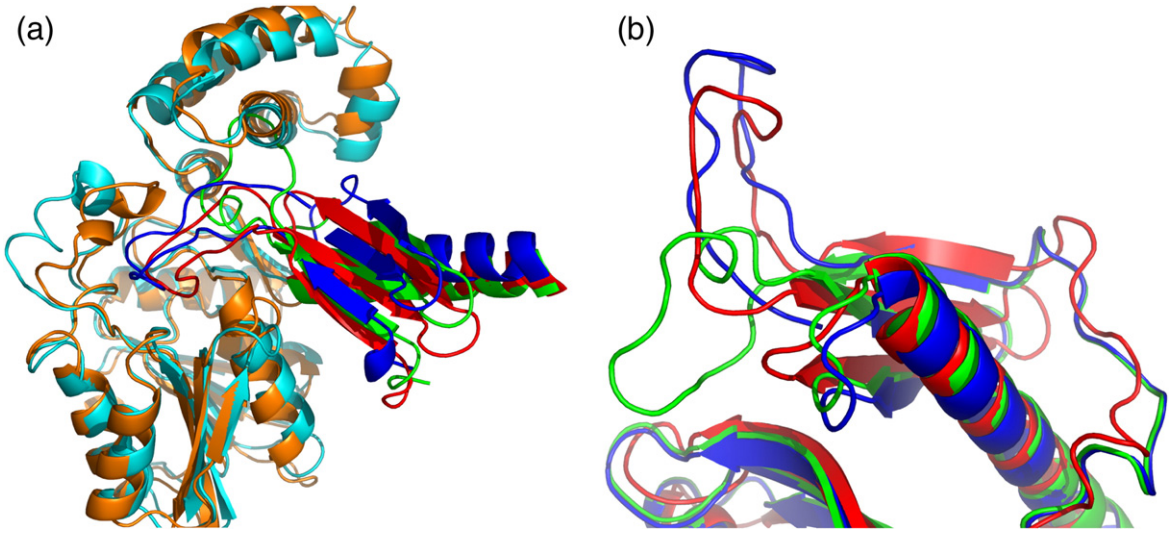


**Fig. 7.** Blind prediction of CAPRI target 20. (a) Superimposition of the native complex and the Rosetta model based on residues at the protein–protein interface. (b) Superimposition of the native and predicted RF1 structures. The drawing and coloring schemes are the same as in Fig. 6.

incorporate backbone flexibility into protein–protein docking. The fold-tree-based RosettaDock method preserves the same functionalities as the original rigid-backbone RosettaDock protocol but is technically superior in that it allows simultaneous treatment and optimization of backbone/side-chain torsional and rigid-body freedom. The examples described in this paper of protein–protein docking with either complete backbone flexibility or flexibility confined to loop regions demonstrate the power and generality of the new approach. The most notable of these examples is the CAPRI Target 20 blind docking challenge in which we were able to successfully model the significant backbone rearrangement upon the complex formation.

The fold-tree concept of representing protein molecules by a directed, acyclic and connected treelike graph is sufficiently general to be applied to a wide spectrum of protein structure modeling tasks, such as *de novo* and homologue-based structure prediction, rigid-backbone and flexible-backbone docking, or even multibody protein assembly. This framework is of particular interest for the flexible-backbone docking problem for several reasons. First, the completely unrestricted flexible-backbone docking problem is over a search space that is of formidable size (all internal degrees of freedom of both partners plus the rigid-body degrees of freedom), and so when possible it is useful to focus sampling on the degrees of freedom (loops, hinges, etc.) most likely to be changing during docking given the available information on the system. Given a specific type of backbone movement, the fold tree can be tailored to direct comprehensive and efficient sampling through the relevant part of conformational space. Second, handling backbone movement in protein–protein docking requires frequent utilization of modeling techniques from monomeric protein structure prediction, such as fragment-based protein folding and loop modeling. With the rigid-backbone representation in the previous RosettaDock method, these steps needed to be conducted independently. In the context of the fold-tree system, protein folding and protein–protein docking tasks can be easily coupled and become interchangeable by dynamically adjusting the fold tree, and this greatly enhances the method efficiency and automation. Last, the fold-tree system integrates both backbone/side-chain torsional and rigid-body freedom and allows simultaneous, gradient-based optimization of all degrees of freedom subject to the energy function. This has direct application in high-resolution refinement of docking models where the energy landscape is extremely complicated and simultaneous adjustment of backbone, side-chain and rigid-body conformations is desirable to overcome local energy barriers to better access the global minimum. As described in the Introduction, tree-based approaches have been used for refinement in internal coordinates[27,28] and protein–ligand docking;[24] the work described in this paper goes beyond these studies by applying tree-based methods to model backbone flexibility during protein–protein docking. A further advance is the great versatility of the problems that can be readily treated by the combination of different sampling strategies with different fold trees, as illustrated by Table 1, Fig. 1, and the wide range of flexible docking problems described in this paper.

Information on the degrees of freedom on which sampling should be focused can come from a variety of sources. Experimental sources include high *B*-factor and disordered regions in X-ray structures, structural variation in NMR ensembles or information obtained via other experimental techniques such as Fourier transform infrared spectroscopy and fluorescence quenching.[40] Computational sources

include molecular dynamic simulations in combination with principal components analysis,[41] analysis of variations in homologous structures in a protein family, flexibility predictions from graph theory[42] and normal mode analysis.[43] In the case of our successful prediction for CAPRI target 20, the flexibility of the Q-loop is evident from the X-ray structure and excluding it in the docking simplifies the sampling space dramatically. The complexity of the resulting loop modeling problem was also reduced significantly given the strong functional constraint.

Our completely flexible docking calculations met with two sampling challenges, the first expected, the second not. The expected challenge was the difficulty in capturing large conformational change, which is a direct reflection of the very large size of the conformational space: in the most general formulation of this problem it is equivalent to two *ab initio* structure prediction problems plus a docking problem simultaneously. The second, more subtle problem is the noise in the energy of the entire macromolecular system (total energy) produced by structural variation far from the binding site. This problem arose in cases where the conformational changes were relatively modest; because of the very large number of degrees of freedom and interatomic interactions within the two interacting proteins, different MCM trajectories can optimize different regions of two partners to different extents, producing considerable variation in the total energy independent of the protein–protein interface. This problem can be alleviated to some extent by focusing on the interface energy rather than the total energy, but this is physically incorrect and can lead to artifacts (e.g., contorted loops that optimize interactions across but not within the interface). Ranking models based on the more physically correct binding energy did not produce better results than ranking solely on the interactions across the interface (Table 2). With more complete optimization, ranking based on the free energy of binding, including entropic effects, should yield more accurate predictions.

In this paper we have presented a general approach for incorporating backbone flexibility in protein–protein docking and shown that the method yields improved results when conformational changes occur upon docking. Despite this progress, our work also highlights the challenges associated with fully accounting for conformational changes in protein–protein docking, primarily the great difficulty of global optimization in very high-dimensional spaces.

## Materials and Methods

### Data set

The monomeric protein test set for loop modeling was originally compiled by Fiser *et al.*[44] It contains 40 proteins for each of the 8- and 12-residue loop subsets. The docking test cases were selected from the benchmark set constructed by Chen *et al.*[35] except 1T6G, which is CAPRI target 18.[45]

### Evaluation of model accuracy

To evaluate model accuracy in the loop modeling test, an rmsd value is calculated over all backbone heavy atoms in the loop region between the model and the native structure after template backbones are superimposed. To evaluate accuracy of the docking models generated in various flexible docking tests, several metrics are used, including fraction of native contacts (Fnat), ligand $C^{\alpha}$ rmsd (Lrmsd) and interface $C^{\alpha}$ rmsd (Irmsd), all of which have been implemented as standard evaluation criteria in the CAPRI experiment.[6] In detail, two residues are considered to be in contact if any of their heavy atoms are found within the cutoff distance of 4 Å. An interface residue is defined if its side-chain centroid is found within 8 Å of any of the side-chain centroids of the other docking partner. Additionally, a CAPRI-style measure[6] is used to combine all the three metrics to determine prediction accuracy of docking models, namely, "high accuracy" for models with Fnat $\geq$50% and Lrmsd $\leq$1.0 Å or Irmsd $\leq$1.0 Å, "medium accuracy" for models with Fnat $\geq$30% and Lrmsd $\leq$5.0 Å or Irmsd $\leq$2.0 Å, "acceptable accuracy" for models with Fnat $\geq$10% and Lrmsd $\leq$10.0 Å or Irmsd $\leq$5.0 Å) and "incorrect" for models with Fnat <10%. We count the numbers of high-accuracy and medium-accuracy models among the top three ranking models in each docking run as a performance measure.

### Loop modeling

In the fold tree, a long-range edge is established for the pair of "stub" residues in the template for each of the predefined loops, and the rigid-body transforms between these residues as well as the backbone torsion angles within the template are fixed throughout. The break points within each loop are randomly selected at the start of each simulation, which has the advantage over the previous protocol[33] that each loop can be built from either the N-terminal stub residue or the C-terminal stub residue or both. For each individual loop, a random starting conformation is constructed by arbitrarily inserting fragments in the loop region. The simulation is performed first in the low-resolution stage in which side chains are represented by centroids and then followed by the full-atom refinement stage in which all atoms including hydrogen atoms are explicitly represented. Within each stage, a series of MC CCD minimization cycles are conducted each of which consists of a step of perturbing the loop conformation, a step of closing the loop chain break by the CCD algorithm[34] and a step of energy minimization of the torsional degrees of freedom in the loop. In the low-resolution stage, the perturbation is done by inserting nine-residue (for loops longer than 15 residues) and/or three-residue (for loops longer than 6 residues) and/or one-residue fragments into the loop region,[33] and a line minimization[46] along the gradient is performed following the CCD loop closure. In the high-resolution stage, the perturbation consists of small random changes to one or more backbone torsion angles ("small/shear" moves[47]) and the Davidon–Fletcher–Powell method[46] is used to find the nearest local minimum on the energy surface following the CCD loop closure. Additional side-chain refinement by repacking all the loop residues and their neighbors is conducted after every 20 cycles as well as at the end of the overall protocol. If multiple loops are to be modeled, they are built onto the template in a

sequential order during the low-resolution stage. In the full-atom MCM stage, a single loop is randomly selected for perturbation and CCD closure, and all the loops are minimized simultaneously. The energy functions used in the loop modeling method are quite similar to those described by Rohl *et al*. except that the chain discontinuity (gap) penalty term is implemented differently due to the inclusion of the explicit CCD loop closure step.[33] Instead of ramping up the gap penalty weight to force loop closure as in the previous protocol, we drop this component from the energy function during the low-resolution stage and only keep a modest weight on its contribution in the full-atom refinement to prevent loops from being broken during energy minimization.

### Local-perturbation docking calculations

With backbone freedom incorporated, the search space is enlarged significantly and a global docking simulation with all-atom refinement of backbone and side-chain degrees of freedom becomes computationally demanding. For each docking test, we carried out local-perturbation docking rather than global docking to reduce the computational cost of the range of docking problems treated in this paper. In these studies, the unbound structures were first superimposed on the native complex (for unbound docking only) and each trajectory starts with a random rigid-body perturbation (maximum 8 Å translation and 8° rotation, compared to complete randomization for global docking). Multiple independent trajectories are carried out to generate an ensemble of docking models in the vicinity of the native complex. The results of these docking calculations were typically evaluated by "energy *versus* rmsd" plots in which total energy, "score" or interface energy, "I_sc" is plotted *versus* ligand $C^{\alpha}$ rmsd, "Lrmsd" or interface $C^{\alpha}$ rmsd, "Irmsd" and the effectiveness of each docking method can be judged by the "funnel-like" character[48] of the plot.

The justification for carrying out perturbation studies is that when local-perturbation docking is successful, global docking is generally also successful given sufficient sampling.[13] In practice, experimental data can often be used as constraints to locate the approximate binding interface and therefore narrow the rigid-body space to be sampled. Furthermore, when backbone conformational changes are not significant, candidate models from rigid-backbone global docking (such as those low-rmsd ones shown in Table 2) can be used as the starting structures to carry out flexible-backbone local refinement.

### Rigid-backbone docking

The rigid-backbone docking protocol implemented in this paper was developed in the context of the fold-tree representation and preserves the same functionalities as the traditional RosettaDock method.[13,14] Briefly, the protocol starts with 500 steps of low-resolution rigid-body MC search followed by 50 cycles of high-resolution MCM[30] refinement during which the conformations of interface side chains are sampled in both on-rotamer and off-rotamer space.

### Docking with backbone minimization

Prerelaxed (see the next section) unbound structures were used to perform the local-perturbation docking studies with backbone minimization and at least 8000 models were generated in each run. In contrast to the rigid-backbone docking protocol, torsional freedom is enabled in the fold tree so that backbone and side-chain torsion angles are minimized simultaneously with the rigid-body freedom during the high-resolution MCM stage. Interface $C^{\alpha}$ rmsd (Irmsd) was used to measure model accuracy instead of ligand $C^{\alpha}$ rmsd (Lrmsd) because Lrmsd can be affected by both the ligand's rigid-body orientation and the internal backbone conformation. In the final ranking, all the models were first ranked by the total energy, and the 5% lowest energy population were re-ranked by the interface energy. The goal of this procedure is to avoid noise introduced into the total energy from noninterface regions while still excluding models that may have overoptimized the complex interface at the expense of intramolecular interactions within each partner. All the test cases are unbound–unbound docking targets except the receptor in 1GLA, which is from the complex conformation. As an intermediate control, those prerelaxed structures were also docked using the rigid-backbone docking method to help analyze the impact of prerelaxing on the docking performance. With the significantly increased number of degrees of freedom to be optimized, the docking protocol with backbone minimization is expected to be more computationally intensive. For example, on a single 2.80-GHz Intel Xeon CPU, it currently takes on average 2.83 and 7.26 min to produce one full-atom docking model for 2SNI (a 300-residue protein complex) using the rigid-backbone and flexible-backbone docking protocol, respectively.

### Prepacking and prerelaxing

As previously described by Gray *et al*.,[13] the starting structures for a rigid-backbone docking run are prepared in a "prepacking" step in which the two docking partners are separated and their side chains are refined by a rotamer packing protocol.[32] Similarly, prior to the docking with backbone minimization, the unbound structures were prepared in a prerelaxing step in which they are individually optimized by an all-atom structure-refinement protocol.[49] The purpose of the prepacking and the prerelaxing is to relieve any existing atomic clashes internal to the partners (side chain only for the rigid-backbone docking and backbone/side chain for flexible-backbone docking) so that they will not be carried into the docking stage and bias model discrimination. Briefly, the protocol consists of multiple MCM cycles in each of which the backbone conformation is perturbed by small torsional angle movements and then all backbone and side-chain torsions are minimized based on a full-atom energy function. The resulting conformation is either accepted or rejected according to the Metropolis criterion. Periodically, side chains are optimized by combinatorial packing[32] or rotamer trials with minimization.[14] Backbone bond lengths and angles are kept fixed during the simulation. In practice, the refinement protocol does not dramatically change the overall protein conformation but rather improves local interactions in the protein. Due to the stochastic nature of this refinement step, we generated 20 models starting from each unbound docking partner. The backbone $C^{\alpha}$ rmsd value of the models ranges from 0.08 to 1.50 Å depending on the size of the protein. We selected the lowest energy model as the starting structure for docking.

### Docking with loop minimization

Information on the loop regions was obtained by visually comparing the bound and unbound structures

and was then used to direct the construction of fold trees. In the fold tree, the long-range edge between the two partners is flexible while the remaining rigid-body connections (each of which connects the two stub residues of each loop) are kept fixed. Only the backbone and side-chain torsion angles of the loop residues as well as the side chains of the neighboring residues around the loops are flexible. During the high-resolution MCM stage, backbone torsions in the loop regions are optimized simultaneously with the rigid-body and side-chain freedom. Since backbone conformational changes are restricted to the loop regions, the starting structures for docking were prepared by prepacking[13] rather than prerelaxing. The total energy was used to rank models since the noise introduced from the noninterface region is much reduced in comparison to docking with backbone minimization.

### Docking with loop rebuiding

As described in the previous section, loop regions were defined in advance and the fold tree contains multiple long-range edges and resembles the combination of simple fold trees for docking and loop modeling. The fold tree is modified in each stage of the protocol to yield desired structural movements and optimizations. At the beginning of the protocol, flexible peptide-bond edges (corresponding to the loops) are removed from the fold tree and the rigid-body orientation between the remaining templates is sampled using a standard docking MC search protocol. The docking long-range edge is then held fixed and each of the removed peptide-bond edges is added back sequentially to the fold tree, as we rebuild the corresponding loops onto the docked structure obtained in the previous stage. The docking and loop modeling in these stages are done with a low-resolution protein representation. Once the loops are rebuilt, the side chains are added back on the current backbone position and those at the interface are repacked. The resulting all-atom model is refined by a protocol in which rigid-backbone MCM docking alternates with loop modeling by MC CCD minimization. The two distinct optimization tasks can be easily interchanged by freezing either the flexible peptide-bond edges (docking) or the docking long-range edge (loop refinement). This would have been a significant technical challenge if the fold-tree representation were not implemented to generalize torsional and rigid-body freedom. In the full-atom refinement, the alternation between docking and loop refinement is repeated five times and the weight on the Lennard–Jones repulsive energy is linearly ramped up to full strength as the simulation progresses. This prevents the two docking partners from flying apart during the initial rigid-body optimization if the loop has been placed imperfectly in the low-resolution stage. The protocol is concluded by an extra cycle of docking MCM with the full repulsive potential.

### Docking energy functions

The energy functions used in docking have similar forms to those described by Gray *et al.*,[13] but a uniform set of weights is applied that has been implemented as standard for Rosetta structure prediction, side-chain packing and sequence redesign. When backbone flexibility is introduced, a secondary structure-dependent backbone torsion potential[47] is added to ensure that backbone torsion angle changes do not step over the disallowed regions in the Ramachandran plot. When loops are modeled or minimized, a chain discontinuity penalty term is included. Models with chain discontinuity penalty larger than 1.0 are considered to have unphysical chain discontinuity and are therefore discarded. In the tests of docking with loop modeling, two types of energy functions are alternated: one is specific for docking and the other is specific for loop modeling (see the previous section on loop modeling).

### Interface energy

The interface for a docking model is calculated as the difference between the energy for the complex and the energy of the separated partners without allowing any structural relaxation. It reports intermolecular interactions across the model interface. Models with positive interface energy typically show unresolved atomic clashes and are excluded from further analysis.

### Binding energy

Binding energy for a docking model is calculated as the difference between the energy of the complex and the energy of the isolated partners after structural relaxation of the side chains. Ten independent repacking calculations were carried out and that with lowest energy was chosen as the reference state to calculate the binding energy.

### BOINC and Rosetta@Home

Rosetta@Home†, a distributed computing project running the Rosetta software on personal computers of volunteers from all over the world using the Berkley Open Infrastructure for Network Computing (BOINC) technology, was critical to the development of the novel methodology described in this paper. This substantial computing resource allowed us to rapidly test and improve the new methodology at a level not possible with only in-house computing resources.

### Plots and figures

Unless specified, Gnuplot‡ was used to make plots. PYMOL§ was used to produce figures for protein models.

### Software availability

The software described in this paper is available free for academic use at http://depts.washington.edu/ventures/UW_Technology/Express_Licenses/Rosetta as part of the Rosetta software suite.

## Acknowledgements

We thank many scientists who have participated in the development of the suite of computational tools used in the Baker laboratory for computations

† www.boinc.bakerlab.org
‡ http://www.gnuplot.info
§ http://www.pymol.org

on the structure of proteins. In particular, Ora Schueler-Furman contributed to making predictions for CAPRI Target 20. Jeffrey Gray laid the groundwork for RosettaDock. David Kim built and maintained the Rosetta@Home project. Keith Laidig and Chance Reschke maintained reliable, state-of-the-art computing resources. We thank all the Rosetta@-Home users worldwide for generously donating their computer time for our scientific research and particularly, users "Libor B" (Libor Böhm from Studenka, Czech Republic), "zoaken" (Kenzo Akamatsu from Boise, ID, USA), "highflyr" (Michel Ward from Wheelers Hill, Australia), "Blackfly" (Steven Davis, from Pasadena, TX, USA) and "Silver Streak" for generating low-energy near-native docking models on their computers. We thank Peter Brzovic for helpful discussions.

## References

1. Uetz, P., Giot, L., Cagney, G., Mansfield, T. A., Judson, R. S., Knight, J. R. *et al.* (2000). A comprehensive analysis of protein–protein interactions in *Saccharomyces cerevisiae*. *Nature*, **403**, 623–627.
2. Gavin, A. C., Bosche, M., Krause, R., Grandi, P., Marzioch, M., Bauer, A. *et al.* (2002). Functional organization of the yeast proteome by systematic analysis of protein complexes. *Nature*, **415**, 141–147.
3. Berman, H. M., Bhat, T. N., Bourne, P. E., Feng, Z., Gilliland, G., Weissig, H. & Westbrook, J. (2000). The Protein Data Bank and the challenge of structural genomics. *Nat. Struct. Biol.* **7**, 957–959.
4. Vajda, S. & Camacho, C. J. (2004). Protein–protein docking: is the glass half-full or half-empty? *Trends Biotechnol.* **22**, 110–116.
5. Janin, J., Henrick, K., Moult, J., Eyck, L. T., Sternberg, M. J., Vajda, S. *et al.* (2003). CAPRI: a Critical Assessment of PRedicted Interactions. *Proteins*, **52**, 2–9.
6. Mendez, R., Leplae, R., De Maria, L. & Wodak, S. J. (2003). Assessment of blind predictions of protein–protein interactions: current status of docking methods. *Proteins*, **52**, 51–67.
7. Mendez, R., Leplae, R., Lensink, M. F. & Wodak, S. J. (2005). Assessment of CAPRI predictions in rounds 3–5 shows progress in docking procedures. *Proteins*, **60**, 150–169.
8. Goh, C. S., Milburn, D. & Gerstein, M. (2004). Conformational changes associated with protein–protein interactions. *Curr. Opin. Struct. Biol.* **14**, 104–109.
9. Dominguez, C., Boelens, R. & Bonvin, A. M. (2003). HADDOCK: a protein–protein docking approach based on biochemical or biophysical information. *J. Am. Chem. Soc.* **125**, 1731–1737.
10. Smith, G. R., Sternberg, M. J. & Bates, P. A. (2005). The relationship between the flexibility of proteins and their conformational states on forming protein–protein complexes with an application to protein–protein docking. *J. Mol. Biol.* **347**, 1077–1101.
11. Bastard, K., Prevost, C. & Zacharias, M. (2006). Accounting for loop flexibility during protein–protein docking. *Proteins*, **62**, 956–969.
12. Schneidman-Duhovny, D., Inbar, Y., Nussinov, R. & Wolfson, H. J. (2005). Geometry-based flexible and symmetric protein docking. *Proteins*, **60**, 224–231.
13. Gray, J. J., Moughon, S., Wang, C., Schueler-Furman, O., Kuhlman, B., Rohl, C. A. & Baker, D. (2003). Protein–protein docking with simultaneous optimization of rigid-body displacement and side-chain conformations. *J. Mol. Biol.* **331**, 281–299.
14. Wang, C., Schueler-Furman, O. & Baker, D. (2005). Improved side-chain modeling for protein–protein docking. *Protein Sci.* **14**, 1328–1339.
15. Schueler-Furman, O., Wang, C. & Baker, D. (2005). Progress in protein–protein docking: atomic resolution predictions in the CAPRI experiment using RosettaDock with an improved treatment of side-chain flexibility. *Proteins*, **60**, 187–194.
16. Bradley, P. & Baker, D. (2006). Improved beta-protein structure prediction by multilevel optimization of nonlocal strand pairings and local backbone conformation. *Proteins*, **65**, 922–929.
17. Noguti, T. & Go, N. (1983). A method of rapid calculation of a 2nd derivative matrix of conformational energy for large molecules. *J. Phys. Soc. Jpn.* **52**, 3685–3690.
18. Abe, H., Braun, W., Noguti, T. & Go, N. (1984). Rapid calculation of 1st and 2nd derivatives of conformational energy with respect to dihedral angles for proteins—general recurrent equations. *Comput. Chem.* **8**, 239–247.
19. Karplus, K., Karchin, R., Draper, J., Casper, J., Mandel-Gutfreund, Y., Diekhans, M. & Hughey, R. (2003). Combining local-structure, fold-recognition, and new fold methods for protein structure prediction. *Proteins*, **53**, 491–496.
20. Abagyan, R., Totrov, M. & Kuznetsov, D. (1994). ICM—a new method for protein modeling and design-applications to docking and structure prediction from the distorted native conformation. *J. Comput. Chem.* **15**, 488–506.
21. Mazur, A. K. & Abagyan, R. A. (1989). New methodology for computer-aided modeling of biomolecular structure and dynamics. 1. Non-cyclic structures. *J. Biomol. Struct. Dyn.* **6**, 815–832.
22. Abagyan, R. A. & Mazur, A. K. (1989). New methodology for computer-aided modeling of biomolecular structure and dynamics. 2. Local deformations and cycles. *J. Biomol. Struct. Dyn.* **6**, 833–845.
23. Fernandez-Recio, J., Totrov, M. & Abagyan, R. (2002). Soft protein–protein docking in internal coordinates. *Protein Sci.* **11**, 280–291.
24. Cavasotto, C. N. & Abagyan, R. A. (2004). Protein flexibility in ligand docking and virtual screening to protein kinases. *J. Mol. Biol.* **337**, 209–225.
25. Brunger, A. T., Adams, P. D., Clore, G. M., DeLano, W. L., Gros, P., Grosse-Kunstleve, R. W. *et al.* (1998). Crystallography & NMR system: a new software suite for macromolecular structure determination. *Acta Crystallogr. D*, **54**, 905–921.
26. Schwieters, C. D., Kuszewski, J. J., Tjandra, N. & Clore, G. M. (2003). The Xplor-NIH NMR molecular structure determination package. *J. Magn. Reson.* **160**, 65–73.
27. Rice, L. M. & Brunger, A. T. (1994). Torsion angle dynamics: reduced variable conformational sampling enhances crystallographic structure refinement. *Proteins*, **19**, 277–290.
28. Schwieters, C. D. & Clore, G. M. (2001). Internal coordinates for molecular dynamics and minimization in structure determination and refinement. *J. Magn. Reson.* **152**, 288–302.

29. Wollacott, A. M., Zanghellini, A., Murphy, P. & Baker, D. (2007). Prediction of structures of multidomain proteins from structures of the individual domains. *Protein Sci.* **16**, 165–175.
30. Li, Z. & Scheraga, H. A. (1987). Monte Carlo-minimization approach to the multiple-minima problem in protein folding. *Proc. Natl Acad. Sci. USA*, **84**, 6611–6615.
31. Metropolis, N., Rosenbluth, A., Rosentbluth, N., Teller, A. & Teller, E. (1953). Equation of state calculations by fast computing machines. *J. Chem. Phys.* **21**, 1087–1092.
32. Kuhlman, B. & Baker, D. (2000). Native protein sequences are close to optimal for their structures. *Proc. Natl Acad. Sci. USA*, **97**, 10383–10388.
33. Rohl, C. A., Strauss, C. E., Chivian, D. & Baker, D. (2004). Modeling structurally variable regions in homologous proteins with Rosetta. *Proteins*, **55**, 656–677.
34. Canutescu, A. A. & Dunbrack, R. L., Jr (2003). Cyclic coordinate descent: a robotics algorithm for protein loop closure. *Protein Sci.* **12**, 963–972.
35. Chen, R., Mintseris, J., Janin, J. & Weng, Z. (2003). A protein–protein docking benchmark. *Proteins*, **52**, 88–91.
36. Wang, C., Schueler-Furman, O., Andrea, I., London, N., Fleishman, S., Bradley, P., *et al*. (2007). RosettaDock in CAPRI rounds 6–12. *Proteins*, doi:10.1002/prot.21684.
37. Graille, M., Heurgue-Hamard, V., Champ, S., Mora, L., Scrima, N., Ulryck, N. *et al.* (2005). Molecular basis for bacterial class I release factor methylation by PrmC. *Mol. Cell*, **20**, 917–927.
38. Yang, Z., Shipman, L., Zhang, M., Anton, B. P., Roberts, R. J. & Cheng, X. (2004). Structural characterization and comparative phylogenetic analysis of *Escherichia coli* HemK, a protein (N5)-glutamine methyltransferase. *J. Mol. Biol.* **340**, 695–706.
39. Vestergaard, B., Van, L. B., Andersen, G. R., Nyborg, J., Buckingham, R. H. & Kjeldgaard, M. (2001). Bacterial polypeptide release factor RF2 is structurally distinct from eukaryotic eRF1. *Mol. Cell*, **8**, 1375–1382.
40. Vigano, C., Manciu, L. & Ruysschaert, J. M. (2005). Structure, orientation, and conformational changes in transmembrane domains of multidrug transporters. *Acc. Chem. Res.* **38**, 117–126.
41. Bonvin, A. M. (2006). Flexible protein–protein docking. *Curr. Opin. Struct. Biol.* **16**, 194–200.
42. Jacobs, D. J., Rader, A. J., Kuhn, L. A. & Thorpe, M. F. (2001). Protein flexibility predictions using graph theory. *Proteins*, **44**, 150–165.
43. Bahar, I. & Rader, A. J. (2005). Coarse-grained normal mode analysis in structural biology. *Curr. Opin. Struct. Biol.* **15**, 586–592.
44. Fiser, A., Do, R. K. & Sali, A. (2000). Modeling of loops in protein structures. *Protein Sci.* **9**, 1753–1773.
45. Janin, J. (2005). The targets of CAPRI rounds 3–5. *Proteins*, **60**, 170–175.
46. Press, W. H., Teukolsky, Saul A., Vetterling, William T. & Flannery, Brian P. (1992). *Numerical Recipes in FORTRAN: The Art of Scientific Computing.* Cambridge University Press, New York, NY, USA.
47. Rohl, C. A., Strauss, C. E., Misura, K. M. & Baker, D. (2004). Protein structure prediction using Rosetta. *Methods Enzymol.* **383**, 66–93.
48. Schueler-Furman, O., Wang, C., Bradley, P., Misura, K. & Baker, D. (2005). Progress in modeling of protein structures and interactions. *Science*, **310**, 638–642.
49. Misura, K. M. & Baker, D. (2005). Progress and challenges in high-resolution refinement of protein structure models. *Proteins*, **59**, 15–29.