BIOINFORMATICS

Vol. 24 ISMB 2008, pages i196–i204
doi:10.1093/bioinformatics/btn169

Algorithm for backrub motions in protein design

Ivelin Georgiev[1], Daniel Keedy[2], Jane S. Richardson[2], David C. Richardson[2] and Bruce R. Donald[1,2,*]

[1]Department of Computer Science, Duke University and [2]Department of Biochemistry, Duke University Medical Center, Durham, NC 27708, USA

ABSTRACT

Motivation: The *Backrub* is a small but kinematically efficient side-chain-coupled local backbone motion frequently observed in atomic-resolution crystal structures of proteins. A backrub shifts the C_α–C_β orientation of a given side-chain by rigid-body dipeptide rotation plus smaller individual rotations of the two peptides, with virtually no change in the rest of the protein. Backrubs can therefore provide a biophysically realistic model of local backbone flexibility for structure-based protein design. Previously, however, backrub motions were applied via manual interactive model-building, so their incorporation into a protein design algorithm (a simultaneous search over mutation and backbone/side-chain conformation space) was infeasible.

Results: We present a combinatorial search algorithm for protein design that incorporates an automated procedure for local backbone flexibility via backrub motions. We further derive a dead-end elimination (DEE)-based criterion for pruning candidate rotamers that, in contrast to previous DEE algorithms, is provably accurate with backrub motions. Our backrub-based algorithm successfully predicts alternate side-chain conformations from four ≤0.9 Å resolution structures, confirming the suitability of the automated backrub procedure. Finally, the application of our algorithm to redesign two different proteins is shown to identify a large number of lower-energy conformations and mutation sequences that would have been ignored by a rigid-backbone model.

Availability: Contact authors for source code.

Contact: brd+ismb08@cs.duke.edu

1 INTRODUCTION

Protein design algorithms aim at identifying protein mutation sequences with desired improved or novel properties, such as: stability (Korkegian *et al.*, 2005; Malakauskas and Mayo, 1998), specificity (Havranek and Harbury, 2003; Kortemme *et al.*, 2004; Lilien *et al.*, 2005; Looger *et al.*, 2003), binding affinity (Lippow *et al.*, 2007), enzymatic function (Jiang *et al.*, 2008; Lassila *et al.*, 2006; Stevens *et al.*, 2006) or even overall fold (Kuhlman *et al.*, 2003). Typically, the *input model* for a structure-based protein design algorithm includes the following: (1) an initial (usually rigid) backbone structure, used as a *template* for the redesign; (2) a rotamer library (Dunbrack, 2002; Lovell *et al.*, 2000; Ponder and Richards, 1987) of low-energy side-chain conformations that discretizes the continuous side-chain conformation space, and thus makes the computational search feasible; and (3) a pairwise energy function (Gordon *et al.*, 1999; Kuhlman and Baker, 2000; Vizcarra and Mayo, 2005) for scoring and ranking the algorithm predictions. To further improve the accuracy of the model, extended rotamer libraries (De Maeyer *et al.*, 1997), flexible rotamers (Georgiev *et al.*, 2008; Mendes *et al.*, 1999) and different levels of backbone flexibility (Desjarlais and Handel, 1999; Fung *et al.*, 2007; Georgiev and Donald, 2007; Harbury *et al.*, 1998; Kuhlman *et al.*, 2003; Su and Mayo, 1997; Zanghellini *et al.*, 2006) have also been introduced. Incorporating additional backbone/side-chain flexibility into the computational model allows the identification of lower energy mutations/conformations that would have been ignored by a rigid model (Georgiev and Donald, 2007).

The combinatorial problem of considering all possible mutations and conformations for each of the mutatable residue positions in a protein poses a significant computational challenge for protein design algorithms. In fact, it has been shown that finding the optimal solution, the Global Minimum Energy Conformation (GMEC), for a given input model with a rigid backbone, a rotamer library and a pairwise energy function, is NP-hard (Chazelle *et al.*, 2004; Pierce and Winfree, 2002). For a protein with a rigid backbone, n mutatable residue positions and at most q rotamers per residue position, a brute-force enumeration procedure must consider $O(q^n)$ possible conformations. Hence, many heuristic techniques, such as Monte Carlo and Self-Consistent Mean Field, have been applied in protein design (Desjarlais and Handel, 1999; Hu and Kuhlman, 2006; Jin *et al.*, 2003; Street and Mayo, 1999; Voigt *et al.*, 2001). Such heuristics are generally fast since only a small subset of the possible conformations is enumerated, but they cannot guarantee the identification of the GMEC for the given input model and can make significant errors (Voigt *et al.*, 2000).

As an alternative, Dead-End Elimination (DEE; Desmet *et al.*, 1992; Gordon *et al.*, 2003) is a provably accurate deterministic algorithm that efficiently reduces the mutation/conformation search space, while enjoying provable guarantees with respect to the GMEC. DEE uses pairwise upper and lower bounds on the rotameric energy interactions to efficiently prune rotamers that are provably not part of the GMEC. Effectively, the DEE-based pruning stage reduces the base q of the enumeration exponent, typically making the subsequent enumeration of the remaining unpruned conformations computationally feasible. Depending on the types of flexibility allowed, several DEE-based algorithms have been derived, in order to guarantee the identification of the GMEC for the respective model. *Traditional DEE* (Desmet *et al.*, 1992; Goldstein, 1994; Lasters and Desmet, 1993; Looger and Hellinga, 2001; Pierce *et al.*, 2000; Yanover *et al.*, 2007) is only provably accurate for a model with a rigid backbone and rigid rotamers. The *MinDEE* pruning criterion (Georgiev *et al.*, 2006, 2008) is provably accurate for a model with a rigid backbone and flexible rotamers over a continuous voxel of side-chain conformation space. The *BD* DEE-based pruning criterion (Georgiev and Donald, 2007) is provably accurate for a model with rigid rotamers and a continuous family

[*]To whom correspondence should be addressed.

of backbone conformations. In general, unlike heuristic approaches, provably accurate algorithms can guarantee the identification of the optimal solution for a given design problem. Furthermore, with provably accurate algorithms, feedback from *in vitro* experiments can be more reliably incorporated into the model, since discrepancies between experimental results and computational predictions can be attributed *solely* to deficiencies of the model (as opposed to the algorithm) (Georgiev and Donald, 2007).

1.1 BD: DEE with backbone flexibility

BD (Georgiev and Donald, 2007) is a DEE-based algorithm that, in contrast to traditional DEE, is provably accurate with rigid rotamers and backbone flexibility. BD places restraining boxes around each residue in a protein, in order to define a continuous family of backbone conformations with small (ϕ, ψ) changes that nonetheless can cause global shifts in the backbone coordinates. Upper and lower bounds on the pairwise rotameric energy interactions are then precomputed within the defined restraining boxes and used to determine which rotamers are provably not part of the respective GMEC. The BD algorithm consists of two stages: (1) first, BD is used to prune a large portion of the candidate rotamers; (2) using A^* search (Leach and Lemon, 1998), the remaining unpruned conformations are then enumerated in order of increasing lower bounds on their energies, in order to obtain the GMEC. When tested on two different protein systems, BD was shown to generate conformations and sequences with significantly lower energies than traditional DEE (albeit at slower running times), thus confirming the potential benefit of incorporating backbone flexibility.

1.2 Backrubs

Based on stereochemical intuition, the existence of a subtle backbone motion coupled to rotamer jumps has long been suspected. Such a motion, the ‘backrub’, was recently confirmed by closely examining the electron density for side-chains modeled as alternates in very high-resolution crystal structures and inferring that the backbone must have shifted between the two conformations to maintain reasonably ideal geometry (Davis *et al.*, 2006). It is conservatively estimated that 3% of all residues undergo backrubs, with a large fraction occurring at the protein surface, most likely reacting to bombardment from solvent molecules. In addition to modeling dynamics, we show that backrubs can allow rotamer changes. Hence, by deduction, they can accommodate mutations to amino acid types for which no rotamers fit in the original backbone. Therefore, it is reasonable to assume that backrubs may play an evolutionary role. Such an assumption is of course impossible to demonstrate from single high-resolution structures and, due to coordinate error on the level of backrub shifts, is also difficult to tease out by comparing otherwise identical-in-sequence point mutant structures. However, one way to address the question is by investigating the effects of backrubs in protein design, which is essentially a guided form of evolution that contributes to our knowledge of the determinants of protein packing and folding. If backrubs enable a provable algorithm to design proteins with low energies, we can be confident that they may also contribute on an evolutionary timescale (Davis *et al.*, 2006).

An analytical description of this highly complex but local motion in backbone dihedral space was found to be intractable, but the simple model implemented by the Backrub tool (Davis *et al.*, 2006) very closely approximates the low-energy plasticity thought to actually occur *in vivo*. To a first approximation, a backrub can be represented by lever-like fanning of the C_α–C_β bond of a given side-chain, coupled to a small rotation of two adjacent peptides, with no effect on the rest of the protein. A backrub at residue *i* is defined by three rotation angles: $\theta_{1,3}$, $\theta_{1,2}$ and $\theta_{2,3}$. The *primary rotation* $\theta_{1,3}$ around the (virtual) $C_{\alpha_{i-1}}$– $C_{\alpha_{i+1}}$ axis rotates residue *i* and its two flanking peptides as a rigid body (Fig. 1, red arrow). The two *flanking rotations* $\theta_{1,2}$ and $\theta_{2,3}$ around the (virtual) $C_{\alpha_{i-1}}$– C_{α_i} and C_{α_i}– $C_{\alpha_{i+1}}$ axes, respectively, can then counter-rotate the individual peptides to (approximately) restore the initial hydrogen-bonding positions of the backbone O and H^N atoms and/or alleviate strain in τ-angles introduced by the primary rotation (Fig. 1, blue arrows). For a given initial backbone conformation, the magnitude of the two flanking rotation angles $\theta_{1,2}$ and $\theta_{2,3}$ can thus be defined as a function of the magnitude of the primary rotation angle $\theta_{1,3}$.

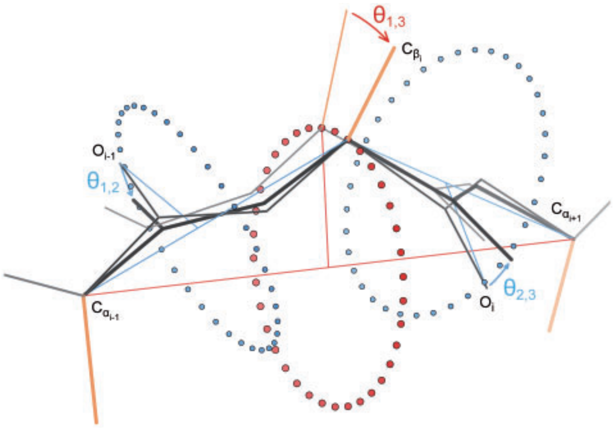

Fig. 1. A backrub schematic. The primary rotation axis $C_{\alpha_{i-1}}$– $C_{\alpha_{i+1}}$ (red) is shown along with the two flanking rotation axes $C_{\alpha_{i-1}}$– C_{α_i} and C_{α_i}– $C_{\alpha_{i+1}}$ (blue). Atom labels indicate the intermediate conformation (after the primary and before the flanking rotations). The red and blue dots trace the paths followed by C_{α_i} and O_{i-1}/O_i during the primary and flanking rotations, respectively. For illustration purposes, the rotation angles shown are larger than typically used in computational experiments.

1.3 Contributions of the article

Backbone flexibility in BD is represented by *global* backbone motions: a change in the backbone conformation of residue position *i* tends to propagate along the rest of the chain (Georgiev and Donald, 2007). In contrast, in this article, we evaluate the benefits of allowing *local* backbone flexibility via backrubs. The local backrub motions and the global BD motions represent very different types of flexibility, and should thus be viewed as complementary, rather than competing, approaches for backbone flexibility in protein design.

By using manual interactive model building, the Backrub tool (Davis *et al.*, 2006) allows a user to choose the three rotation angles (Section 1.2) and apply the corresponding backrub motion. However, no automated backrub procedure has been previously developed. In this article, we present a straightforward approach that automates the backrub computation (Section 2.2). We further apply this approach as part of a combinatorial search algorithm for protein design (Section 3). The latter captures a theme in computational protein design. Many modeling improvements, such as backrubs, can be suggested *for a single protein structure or sequence*. A design algorithm must ‘lift’ each such model to a pairwise bounding and pruning mechanism. Such a mechanism is usually a non-trivial exercise in algorithm design (viz. Sections 2–4), and a prerequisite before the new model (in this article: backrubs) can be exploited in a combinatorial search (e.g. DEE) across *all allowed protein mutations and conformations*.

We show that choosing the primary rotation angle $\theta_{1,3}$ can define the flanking rotations $\theta_{1,2}$ and $\theta_{2,3}$ via kinematics and minimization, and is hence sufficient to parameterize a backrub for a given residue, resulting in a 1 degree-of-freedom per residue design problem. Hence, defining a finite set of possible backbone conformations by sampling the single $\theta_{1,3}$ parameter for each flexible residue, should not be prone to severe undersampling. A simple approach could then apply traditional DEE separately for each of the possible backbone conformations in the finite set. However, this would require that all of the following stages be explicitly performed separately for each backbone: pairwise energy precomputation (cf. Section 3), DEE pruning, and conformation enumeration. As an alternative, we have derived a novel DEE-based algorithm that can be *simultaneously* applied for a finite set of possible backbone conformations (Section 2.1); with this new algorithm, the pairwise energy precomputation, DEE pruning, and conformation enumeration stages must be performed only once. This obtains a significant advantage in computational efficiency. In particular, we make the following contributions in this article: (1) Backrub DEE (BRDEE): a DEE-based algorithm for pruning rotamers that are provably not part of the GMEC for a finite set of backbone conformations; (2) An automated procedure for the generation and energy-based ranking of backrub motions; (3) A novel algorithm for protein design, incorporating our automated backrub procedure and BRDEE; and (4) We first apply our algorithms to predict alternate conformations from crystal structures. Next, we apply them to redesign two proteins: (a) the adenylation domain of the non-ribosomal peptide synthetase (NRPS) enzyme Gramicidin Synthetase A (GrsA-PheA) and (b) the core of the β1 domain of protein G (Gβ1). Gβ1 is a small protein that is a suitable benchmark for protein design algorithms (Georgiev and Donald, 2007). The redesign of GrsA-PheA has potential significant biomedical application to the design of novel antibiotics (Stevens *et al.*, 2006).

2 APPROACH

2.1 BRDEE

The BD algorithm (see Section 1.1) is applicable for protein design problems where backbone conformations are represented by a *bounded continuous* family of solutions. In this section, we first show how analogous ideas can be exploited to derive BRDEE, a provably accurate pruning algorithm for problems where backbone conformations are represented by a *finite* set of solutions. We then specialize BRDEE for backrub motions.

2.1.1 DEE for finite backbone sets. First, we make the following definitions. We will define the protein template t' to include the protein backbone, as well as the side-chains of all residues that are fixed and are not subject to rotamer-based modeling; let $E_{t'}(B_c)$ be the template energy of the system for a given backbone B_c. Let i_r denote rotamer identity r at residue position i. Then, we define $E(i_r|B_c)$ and $E(i_r, j_s|B_c)$ to be, respectively, the self-energy of i_r (the sum of the intra-rotamer and rotamer-to-template energies for i_r) and the pairwise energy between rotamers i_r and j_s when backbone conformation B_c is assumed.

Now, let us have a subset Q of residues that are modeled as flexible. Let Y be the discrete set of allowed backbone conformations B_c. Let $Z(i_r)$ be the set of side-chain dihedral conformations for rotamer i_r and let the Cartesian product $S(i_r) = Y \times Z(i_r)$ be the set of possible conformations of rotamer i_r and its associated backbone. Here, we will assume the use of rigid rotamers ($|Z(i_r)| = 1$), although the following derivation holds for $|Z(i_r)| > 1$ as well. The following lower and upper bound definitions can now be made:

$$E_{t'_{\ominus}} = \min_{B_a \in Y} E_{t'}(B_a); \quad E_{t'_{\oplus}} = \max_{B_a \in Y} E_{t'}(B_a); \tag{1}$$

$$E_{t'_{\oslash}} = E_{t'_{\oplus}} - E_{t'_{\ominus}}. \tag{2}$$

Here, $E_{t'_{\ominus}}$ represents a lower bound on the template energy for the given set of allowed backbone conformations. Similarly, $E_{t'_{\oplus}}$ is an upper bound on the template energy, and $E_{t'_{\oslash}}$ represents the range of possible template energies.

We then define the following rotamer-based terms:

$$E_{\ominus}(i_r) = \min_{z \in S(i_r),\, B_a \in Y} E(z|B_a); \tag{3}$$

$$E_{\ominus}(i_r, j_s) = \min_{z_1 \in S(i_r), z_2 \in S(j_s)} E(z_1, z_2). \tag{4}$$

Here, $E_{\ominus}(i_r)$ represents a lower bound on the self-energy of rotamer i_r for the given set of allowed backbone conformations, while $E_{\ominus}(i_r, j_s)$ represents a lower bound on the pairwise energy between rotamers i_r and j_s. The respective upper bounds ($E_{\oplus}(i_r)$ and $E_{\oplus}(i_r, j_s)$) and ranges of possible energies ($E_{\oslash}(i_r)$ and $E_{\oslash}(i_r, j_s)$) are defined analogously.

The BRDEE *pruning criterion* for a given rotamer i_r is then defined to be:

$$\begin{aligned} &E_{\ominus}(i_r) + \textstyle\sum_j \min_s E_{\ominus}(i_r, j_s) \\ &- E_{t'_{\oslash}} - \textstyle\sum_j \max_s E_{\oslash}(j_s) - \sum_j \sum_k \max_{s,u} E_{\oslash}(j_s, k_u) \\ &> E_{\oplus}(i_t) + \textstyle\sum_j \max_s E_{\oplus}(i_t, j_s), \end{aligned} \tag{5}$$

where $j, k \neq i, k > j$ and $\max_{s,u}$ is over the rotamer sets R_j and R_k for given residues j and k.

When Equation (5) holds, rotamer i_r can be pruned from further consideration, since it provably cannot belong to the GMEC for the allowed set Y of backbone conformations. The proof of this claim is identical to the proof of Proposition 1 in (Georgiev and Donald, 2007). The inclusion of the $E_{\oslash}(\cdot)$ terms in Equation (5) accounts for possible energy changes due to changes in the backbone conformation. Hence, unlike traditional DEE, by appropriately manipulating lower and upper energy bounds, Equation (5) *simultaneously* takes into account all possible conformations from a given finite set of backbones. Since the $E_{\oslash}(\cdot)$ terms can be precomputed, the cost of evaluating Equation (5) is $O(q^2n^2)$ for n residue positions and at most q rotamers per position, equivalent to the cost of the corresponding traditional DEE, MinDEE and BD conditions.

It should be noted that the *form* of Equation (5) is identical to the initial BD pruning condition (Georgiev and Donald, 2007). The major difference, however, is that in BD the $E_{\ominus}(\cdot)$, $E_{\oplus}(\cdot)$ and $E_{\oslash}(\cdot)$ are defined over a bounded *infinite* and *continuous* voxel of backbone conformation space; in BRDEE, these terms are defined over a *finite* set of backbone conformations. BD and BRDEE are thus applicable to significantly different protein design problems.

2.1.2 Specialization for backrubs. Since our interest is in introducing backrubs into protein design, we will now consider the case where the set Y is defined using backrubs. For each residue $i \in Q$, let Y_i be the set of allowed backbone conformations (resulting from backrub motions) for residues $i-1$, i and $i+1$. To avoid combinatorial blowup, we will require that the *backrub independence condition* (*BIC*) holds. Let A_i be the set of atoms that can change their position (3D coordinates) upon a backrub at residue $i \in Q$; similarly, we define A_j. *BIC* then ensures that $A_i \cap A_j = \emptyset$, i.e. that there is no overlap between A_i and A_j, for all pairs $i,j \in Q$. Due to the local nature of backrubs (Section 1.2), *BIC* only requires that no two residues that are adjacent in the protein sequence will be simultaneously allowed to perform backrub motions. When *BIC* holds, the set Y of possible backbone conformations can simply be defined as $Y = F \times \prod_{i \in Q} Y_i$, where F is the fixed part of the protein template. Further, when *BIC* holds, the set of possible conformations for rotamer i_r will only depend on Y_i (and not on Y_j, for all $j \neq i \in Q$), i.e. $S(i_r) = Y_i \times Z(i_r)$. Finally, when Equation (5) holds, rotamer i_r is *provably* not part of the GMEC when all possible backrubs in Y_k for all $k \in Q$ are considered.

2.2 Automated backrubs

The backrub motion is well defined (Section 1.2). Currently, however, the magnitude of the backrub primary and flanking rotations must be determined manually, through visual inspection, using the BACKRUB tool (Davis *et al.*, 2006). However, the manual application of backrub motions for each mutation/rotameric conformation in a protein design combinatorial search is infeasible. Automating the computation of backrub motions is therefore essential if these types of motions are to be used as part of a protein design algorithm. Here, we present the following straightforward computational procedure for fully automated backrubs.

(1) *Given a residue position i to be backrubbed and a primary rotation angle, determine the magnitude of the two flanking rotations.* We use a simple geometric approach to determine the magnitude of the two flanking rotations for a given primary rotation. We must compute the flanking rotations $\theta_{1,2}$ and $\theta_{2,3}$ (Section 1.2). Let $\mathbf{p}(O_{i-1})$ and $\mathbf{p}'(O_{i-1})$ be the positions of the backbone O of residue $i-1$ before and after the primary rotation, respectively. Let $f_{i-1}(\alpha, \mathbf{p})$ be the position of point $\mathbf{p}$ after a flanking rotation of α degrees for peptide $i-1$. Then, let $\alpha_o = \mathrm{argmin}_\alpha |f_{i-1}(\alpha, \mathbf{p}'(O_{i-1})) - \mathbf{p}(O_{i-1})|$ be the flanking rotation angle that moves the backbone O of peptide $i-1$ to the point closest to its original position. We can then compute $\theta_{1,2} = \varepsilon \alpha_o$, where $0 \leq \varepsilon \leq 1$ is a scaling factor used to limit the distortion in the respective τ angles (Section 3). The rotation angle $\theta_{2,3}$ is computed analogously.

(2) *Given a set Q of residues for which backrubs will be applied and a set U_i of primary (together with the corresponding flanking) rotations for each residue i in Q, find the optimal backrub combination $j_i \in U_i$, for each i.* Here, a *backrub combination* $(j_1, \ldots, j_n)$ is a particular assignment of backrub rotation angles for each of the n flexible residue positions. Backrub combinations are generated from the Cartesian product of the sets U_i. A steric filter is applied *during* the backrub enumeration, in order to prune a combinatorial number of backrub combinations from further consideration (see Section 4). Backrub combinations that pass the steric filter are evaluated and ranked using our energy function (Section 4). This step guarantees the identification of the optimal backrub combination, given the input parameters and energy function.

Hence, using as input only: (1) a set of residues for which backrubs will be applied and (2) a set of primary rotation angles, backrub conformations can be generated using the automated procedure described in this section. We therefore now have the necessary tools to use backrubs in protein design.

3 ALGORITHM

We now present our novel protein design algorithm, incorporating the automated backrub procedure described in Section 2.2 and BRDEE (Section 2.1). The algorithm consists of four main steps: (1) *Backrub set generation.* In this first step, the sets Y_i of allowed backrubs at each residue position i are generated using step 1 of Section 2.2. The input for this step is the set Q of residue positions that are modeled as flexible using backrubs and rotamers, and a set of allowed primary rotation angles. A steric filter (Section 4) is applied to prune clashing backrub/residue position combinations. Since a backrub at residue i introduces small changes into the τ angles (N–C$_\alpha$–C$'$) for residues $i-1$, i and $i+1$ (Davis *et al.*, 2006), a τ-angle filter (Section 4) further prunes backrubs that introduce large distortions in the τ angles of the affected residue positions.

(2) *Pairwise lower and upper energy bounds precomputation.* Using the sets Y_i computed in Step 1 above, compute the $E_\ominus(\cdot)$ and $E_\oplus(\cdot)$ terms (Section 2.1). Details of the method for computing the lower and upper energy bounds can be found in Section 4.

(3) BRDEE *pruning.* The precomputed $E_\ominus(\cdot)$ and $E_\oplus(\cdot)$ terms are applied to evaluate Equation (5). Analogously to the extensions for traditional DEE and BD (Georgiev and Donald, 2007), we have also derived four extensions to BRDEE for improved pruning: the simple, general, and pairs Goldstein (1994), and the conformational splitting (Pierce *et al.*, 2000) criteria. These extensions are used in combination with the initial BRDEE criterion Equation (5) and the *DACS* algorithm (Georgiev *et al.*, 2006) in repeated rotamer pruning cycles until no further pruning can be achieved. This pruning step aims at significantly reducing the number of unpruned rotamers that must be considered in the subsequent enumeration stage.

(4) *Enumeration and minimization.* In the final step of the algorithm, using the $E_\ominus(\cdot)$ terms, a version of A^* search enumerates *rotamer vectors* (an assignment of a particular rotamer identity for each flexible residue position) in order of increasing lower bounds (Section 4) on their energy. For each of the generated rotamer vectors, *backrub minimization* is then performed by applying the automated procedure from Step 2 of Section 2.2 to find the respective lowest energy backrub combination. A steric filter is applied to prune a combinatorial number of backrub/rotamer combinations (Section 4). The enumeration is halted once the lower bound on the energy of the next rotamer vector generated by A^* exceeds the best conformation energy found in the search. At that point, we are guaranteed (cf. Georgiev *et al.*, 2008) to have obtained the GMEC for the given design problem and input model.

4 METHODS

Two different sets of experiments were performed to validate our algorithms: recovery of alternate conformations from atomic-resolution crystal structures

(Section 5.1) and redesign of two proteins: the active site of GrsA-PheA and the core of Gβ1 (Section 5.2).

The structural model for GrsA-PheA (PDB id: 1amu; Conti *et al.*, 1997) is as described in (Georgiev and Donald, 2007). The residues modeled as flexible using backrubs and rotamers were seven of the active site residues: A236, W239, T278, I299, A301, A322, I330. The allowed amino acid types at each of these positions were GAVLIFYWM, as well as the wildtype identity. The Penultimate rotamer library modal values (Lovell *et al.*, 2000) were used. The ligand was also modeled using rotamers and was further allowed to rotate/translate. Five primary backrub rotation angles were allowed for each of the flexible residues: −8, −4, 0, 4 and 8 degrees, for a total of five backrubs per flexible residue. A 2-point mutation search (in a k-point mutation search, any k flexible residues are allowed to mutate simultaneously) was performed, to switch the GrsA-PheA specificity towards a non-cognate substrate, Leu. The structural model for Gβ1 is as described in Georgiev and Donald (2007). The same set of five primary backrub angles was allowed for each of the 12 residues (3, 5, 7, 9, 20, 26, 30, 34, 39, 41, 52, 54) in the core of Gβ1 that were modeled as flexible using backrubs and rotamers. For the alternate conformation experiments, four atomic-resolution structures were used: deamidated bovine pancreatic ribonuclease (PDB id: 1dy5; Esposito *et al.*, 2000), *Micrococcus lysodeikticus* catalase (1gwe; Murshudov *et al.*, 2002), xylose isomerase (1muw; Fenn *et al.*, 2004) and extended-spectrum SHV-2 β-lactamase (1n9b; Nukaga *et al.*, 2003). Hetero atoms and water were not included. The allowed primary rotation angles were from −10° to 10° at steps of 1°, for a total of 21 backrubs per flexible residue.

The energy function consists of the Amber electrostatic and vdW terms (Cornell *et al.*, 1995; Weiner *et al.*, 1984) and the EEF1 pairwise solvation energy term (Lazaridis and Karplus, 1999). The following parameters were used: a distance-dependent dielectric of 6.0, a solvation-energy scaling factor of 0.05 and a vdW radii scaling factor of 0.95. A lower bound on the energy of a conformation (used by the A^* enumeration) is computed as a sum of lower bounds on pairwise interactions (Georgiev *et al.*, 2006). The lower bound $E_{\ominus}(i_r, j_s)$ for a given rotamer pair is computed as the minimum energy between i_r and j_s of a sterically allowed conformation, over the set of backrub combinations for residues i and j. Similarly, the upper bound $E_{\oplus}(i_r, j_s)$ is computed as the maximum energy of a sterically allowed conformation for the given set of backrubs. The energies involving the template are computed analogously.

All of traditional DEE, BD and BrDEE are GMEC-based algorithms: typically, the goal is to identify only the single lowest-energy conformation. These algorithms (e.g. Equation 5) can be modified to guarantee the identification of all sequences/conformations within E_w from the respective GMEC energy (Georgiev *et al.*, 2008). In our alternate conformation experiments, $E_w = 20$ kcal/mol, except for 1n9b, where $E_w = 100$ kcal/mol. In our redesign experiments, $E_w = 5$ kcal/mol for BrDEE and traditional DEE; BD used a cutoff of 5 kcal/mol relative to the best BrDEE conformational energy for the given redesign. Hence, the BD and traditional DEE running times are longer than (Georgiev and Donald, 2007), where $E_w = 0$.

For the BD experiments, the restraining boxes around each residue in the protein were defined using the following two bounding criteria: (1) a maximum C$_\alpha$ displacement of 1.5 Å from the original PDB coordinates and (2) a maximum change of ±3° (from the initial values in the PDB structure) for the (ϕ, ψ) angles of flexible residues (Georgiev and Donald, 2007).

A value of 0.7 was used for ε (Section 2.2), to limit distortion in the τ angles. In the *Backrub set generation* step (Section 3), the τ-angle filter prunes backrubs causing large distortions in the τ angles. For a backrub at residue i, the τ angles at residues $i-1$, i and $i+1$ are checked. If all of these angles are within $\pm\delta$ degrees from an ideal value λ, the backrub is allowed; otherwise, if the post-backrub τ angles are closer to λ than the initial τ angles, the backrub is allowed; otherwise, the backrub for the given residue position is pruned. In all of the described experiments, $\lambda = 111.0$ and $\delta = 5.5$.

All conformations for which at least one pair of atoms has a steric overlap of more than η Å are pruned. For the *Backrub set generation* step (Section 3), the steric filter reduces the set of backrubs allowed at each flexible residue position; here, steric checks are performed only against the fixed part of the molecule. For the *Pairwise lower and upper energy bounds precomputation* step (Section 3), the steric filter prunes backrub combinations for the given rotamers; here, steric checks also include the side-chains of the given rotamers. For the *Enumeration and Minimization* step (Section 3), the steric filter prunes entire subtrees of the conformation search tree (cf. Lilien *et al.*, 2005). The alternate conformation experiments used $\eta = 0.6$. For the protein redesign experiments, hydrogens were not used in steric checks; hence, a stricter cutoff of $\eta = 0.4$ was used. Finally, for the *Pairwise lower and upper energy bounds precomputation*, initial rotamer conformations (before backrub application) with a steric overlap of more than 1.75 Å (alternate conformation experiments) and 1.5 Å (redesign) were pruned. Rotamers with a self-energy lower bound and rotamer pairs with a pairwise energy lower bound of more than 30 kcal/mol were also pruned.

5 RESULTS AND DISCUSSION

5.1 Alternate conformation recovery

A first-level test of BrDEE was whether it could reproduce well-characterized backrubs in high-resolution (≤0.9 Å) crystal structures (Table 1). The residues chosen (Table 1) are at least partially buried, represent both hydrophobic and polar amino acids, and have two alternate conformations related by a backrub (Davis *et al.*, 2006), so they serve as excellent starting points for this analysis. Here, we define side-chain plus backbone conformations that are not present in either of the two alternate conformations as *decoys*. The initial backbone conformation and C$_\beta$ position was used as a starting point for each alternate; ideal geometry rotamers were subsequently introduced by BrDEE. If the side-chain rotamers and backbone conformations represented by alternates A and B scored better (i.e. had lower energy) than other decoy conformations, we could be confident that our automated backrub procedure produces physically reasonable conformations. The results were as follows.

5.1.1 1muw. In the deposited model, the alternate side-chain and backbone conformations for Val168 have a relatively large C$_\beta$ displacement (0.66 Å). This residue is found on the buried side of a helix with minimal exposure to solvent. The swap between *m* and *t* rotamers (Lovell *et al.*, 2000) is enabled by a backrub in a manner that is commonly observed for valines (Fig. 2). When starting from A, BrDEE finds the lowest conformational energy to be the A-like rotamer (that is, the rotamer whose side-chain dihedrals are on average closest to those of the deposited alternate A) with a backrub in the A direction and the second lowest energy (about 9 kcal/mol worse) to be the B-like rotamer with a backrub in the B direction. When starting from B, the order of returned conformations is reversed, but the calculated energy difference (about 0.3 kcal/mol) is negligible, suggesting that the initial C$_\beta$ position biases the comparison between these putatively equivalent alternates but that BrDEE approximates the correct relationship either way. From either starting conformation, the third possible rotamer (*p*), definitively not observed in the experimental density, is sterically allowed, but it scores 5–13 kcal/mol worse than its closest competitor.

5.1.2 1gwe. Asp163 is a helix N-cap (Richardson and Richardson, 1988) that alternates between the two common hydrogen bonds at such a position, satisfying either the NH of residue $i+2$ (A) or $i+3$ (B). Although the structure was deposited with a single backbone and the C$_\beta$ displacement between alternate side-chains is relatively low (0.2 Å), close examination reveals

Table 1. Alternate conformation results

PDB[a]	Res[b]	Starting from A[c]			Starting from B[c]		
		Conf[d]	$\theta_{1,3}$	E[e]	Conf[d]	$\theta_{1,3}$	E[e]
1muw	V168	A-like	0	−188.6	B-like	+1	−186.0
		B-like	+10	−179.4	A-like	− 5	−185.8
		decoy	+9	−174.5	decoy	0	−172.9
1gwe	D163	A-like	0	−280.1	A-like	−1	−280.1
		B-like	+4	−270.8	B-like	+3	−268.8
1n9b	I47	A-like	0	−254.1	A-like	0	−226.8
		B-like	+10	−251.4	B-like	−10	−222.6
1dy5	Mb29	B-like	+1	−254.6	B-like	+3	−254.6
		A-like	−2	−254.0	A-like	−1	−253.4
		decoy	−2	−253.1	decoy	−1	−252.8
		decoy	−3	−253.1	decoy	−1	−252.4
		decoy	+10	−240.0			

The [a]PDB id for each structure is shown along with the [b]residue for which alternate conformation recovery was performed. [c]A and B refer to the alternate conformations labeled as A and B in the PDB files. The [d]conformation predicted by the algorithm is similar to A (A-like), to B (B-like), or to neither (a decoy); $\theta_{1,3}$ is the primary backrub angle (in degrees); [e]the computed energy (in kcal/mol).

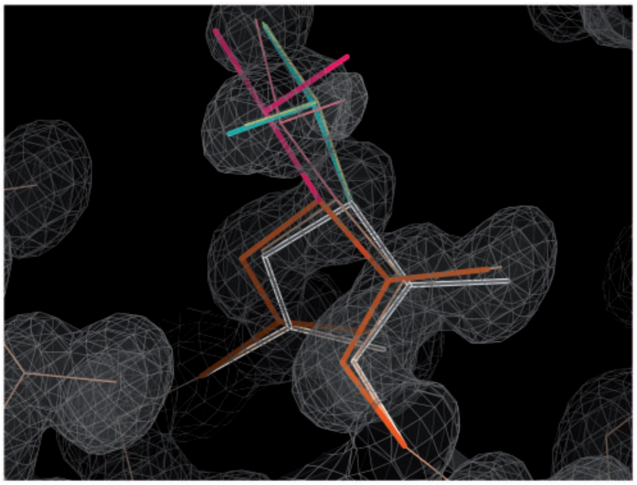

Fig. 2. Alternate conformation recovery for 1muw. Val168 backbones and side-chains from the PDB model (thinner lines) and predicted by Brdee (thicker lines) are shown. The side-chains are colored as follows: model A (pale yellow), A-like from Brdee (cyan), model B (light pink), B-like from Brdee (dark pink). $2F_o - F_c$ electron density is shown at 1.2σ. The model A and A-like from Brdee conformations are almost perfectly superposed.

that the electron density can be equally well satisfied when this displacement is modeled by a backrub; this assertion is supported by observations in the Richardson lab that backrubs may play an important role at N-caps (data not shown). Both alternates are very close to modal rotamers in dihedral space, so Brdee is able to recapitulate them very accurately (when starting from either A or B) with approximately the same 10 kcal/mol energy difference between A-like and B-like conformations. No decoys were identified by Brdee.

5.1.3 1n9b. Ile47 is in an antiparallel β-sheet. The adjacent Ala59 ends a three-residue alternate conformation and is deposited with separate backbones, and although Ile47 is not, it should be: conformation A is non-ideal with a 0.38 Å C_β deviation (Lovell *et al.*, 2003) and 0.59 Å between alternate C_β's. It has been shown that a backrub better relates the two alternate rotamers, while allowing near-ideal geometry (Davis *et al.*, 2006). To demonstrate this in the context of our new algorithm, we first used the original, distorted backbone and C_β conformations. When starting from the A form on both strands, the only conformation returned reproduces the A-like rotamer and backrub of Ile47. When starting from both B forms, however, the A-like rotamer and backrub at Ile47 is returned first, followed by a decoy conformation and then the B-like rotamer and backrub. Notably, all three of these B-derived conformations scored at least 40 kcal/mol worse than the A-derived one, indicating distortion in the original B alternate backbone and C_β. To improve the geometry of the model, we split Ile47's backbone with a manual backrub and idealized the C_β's of both its alternates as a pre-processing step. As a result, the only conformations returned from either starting point (A or B on both strands) represent the original A- and B-like rotamers and backrubs, in that order. Moreover, the energy differences between A-like and B-like conformations at Ile47 are only about 4 kcal/mol or less, indicating the success of our backrub modeling.

Occupancies of the A and B conformations are nearly equal for both Ile47 and Ala59, and therefore it seems possible that they were paired incorrectly in the deposited coordinates. To investigate this question, we tried starting from the A conformation at 47 and the B conformation at 57–59 and vice versa, to determine if a better strand–strand coupling might be obtained from Brdee. In all cases, only the same A-like and B-like Ile47 conformations are returned, but the A conformation for 57–59 is preferred by about 30 kcal/mol, regardless of the returned conformation for Ile47. This indicates that from a purely energetic perspective, there is no strand–strand coupling and the deposited alternate conformation designations are satisfactory.

To ensure that a B-like conformation was returned in the tests just described, it was necessary to augment the rotamer library with the deposited B side-chain's dihedrals. Before splitting the backbone and adding this side-chain to the library, Brdee scored the B-like modal rotamer for Ile47 worse than a decoy when starting from B and forwent it entirely when starting from A. This can likely be attributed to local packing constraints—the deposited B side-chain has χ_1 over 20° away from the modal rotamer, which therefore does not match well. More generally, this shows that denser coverage of side-chain and backbone conformational space can be combined to achieve more realistic modeling.

5.1.4 1dy5. Met b29 (from chain b) was chosen because its alternate side-chains are better rotamers, have more closely matched occupancies, and fit the experimental electron density more clearly than those of Met a29 (from chain a). This residue lies on a helix approximately half-exposed to solvent. It is also deposited with a single backbone, but the displacements of the side-chain atoms as seen in the electron density and that density's anisotropy perpendicular to the chain's direction reveal that a backrub better models this residue. Starting from either A or B, a B-like rotamer and backrub are ranked highest by Brdee, followed by an A-like rotamer and backrub, and two decoys differing from alternate A only at C_ε. Starting from A, a third, substantially worse (13 kcal/mol greater than any other conformation) decoy is also returned. This decoy is therefore clearly energetically discriminated from more realistic conformations, but it was identified by the algorithm, since the backrub allowed the side-chain to displace itself laterally to mostly escape a steric clash on one side. The first two decoys most likely would have been pruned had the nearby acetate from the PDB structure been included in the structural model. The reason the B-like conformation scores slightly better than the A-like is likely due to the fact that the deposited alternate A has a modest steric clash with Tyr b25 of the preceding helical turn. This explanation is further supported by the observation that Ser b32 on the following turn of the helix from Met b29 is modeled with alternate side-chains, indicating the presence of helical turn-to-turn interactions. Interestingly, all generated conformations from each starting point (except for the third decoy from A) were within 2.2 kcal/mol of each other, implying that despite dealing with small energetic differences, BRDEE correctly discriminated physically realistic conformations from decoys.

After proper remodeling of 1n9b's backbone, A- and B-like conformations in terms of rotamer and backrub direction and approximate magnitude were recovered in every case, whether starting from the A or B backbone and C_β. Moreover, if generated at all, decoys always scored worse than the crystallographically observed conformations. This brings up an interesting point that arose from energy minimization of the side-chain dihedrals for 1muw Val168's BRDEE conformations starting from A. After energy minimization, the difference between the real A-like and B-like conformations generated by BRDEE decreased by over 5 kcal/mol whereas the difference between the best scoring conformation (A-like) and the decoy changed by <1 kcal/mol. If generally applicable, this simple test reveals the potential improvement in accuracy/decoy discrimination that BRDEE might gain by further exploring conformational space around rotamers, e.g. with an extended rotamer library or continuous side-chains (Georgiev *et al.*, 2008).

5.2 Protein redesign

5.2.1 Redesign of GrsA-PheA for Leu. For comparison, each of traditional DEE, BRDEE, and BD was applied in separate redesigns of GrsA-PheA (Table 2). As Table 2 shows, the lowest conformation energy identified by BRDEE is >1 kcal/mol lower than the lowest conformation energy identified by traditional DEE. Moreover, traditional DEE identified only 88 rotameric conformations (each conformation represents a unique rotamer vector), representing 7 unique sequences, with energies within 5 kcal/mol from the lowest BRDEE conformation energy. In contrast, BRDEE identified more than 600 conformations, representing 39 unique sequences, with energies within 5 kcal/mol from the lowest BRDEE energy. This confirms that BRDEE is capable of generating a significantly larger number of low-energy sequences and conformations, as compared to traditional DEE. It is interesting to note that the set of low-energy conformations/sequences identified by BD (Table 2) is much larger than BRDEE. This finding can be explained by the fact that backbone flexibility in BD is represented by global motions, whereas backrubs represent much smaller scale local motions. Moreover, BD is defined over a *continuous* family of solutions, whereas BRDEE is defined over a *discrete* backbone set. Hence, since BD and BRDEE represent very different (and complementary) types of backbone motions, interesting future work would involve the derivation of an algorithm for protein design that simultaneously allows both types of backbone flexibility.

Table 2. DEE comparison for GrsA-PheA and Gβ1 redesign

Redesign	DEE[a]	Best energy[b]	Sequences[c]	Confs[d]
GrsA-PheA	traditional DEE	−241.41	7	88
	BRDEE	−242.58	39	605
	BD	−251.19	422	6805
Gβ1	traditional DEE	−371.19	67	169
	BRDEE	−372.86	164	599
	BD[e]	−375.13	> 950	> 3500

The [b]lowest conformation energy (in kcal/mol) identified by each of the three [a]DEE algorithms (traditional DEE, BRDEE, and BD), which is shown along with the [c]number of sequences and [d]number of conformations with energy better (lower) than −237.58 kcal/mol (GrsA-PheA) and −367.86 (Gβ1) (so that all sequences and conformations within 5 kcal/mol from the corresponding lowest BRDEE conformation energy are included). [e]The BD experiments for Gβ1 were halted after not completing in more than 4 weeks.

Furthermore, as Figure 3 shows, the distribution and frequencies of the Leu-binding mutations for each of the seven GrsA-PheA mutatable residue positions is significantly different for traditional DEE, BRDEE and BD. Specifically, by identifying additional low-energy conformations and sequences, BRDEE expanded the computationally predicted sequence space for residues 239 and 301. While traditional DEE predicted only a single amino acid type at each of these two residue positions, BRDEE identified 3 (for 239) and 6 (for 301) amino acid types. Thus, as expected, the incorporation of backrubs into the protein design algorithm leads to the identification of lower energy sequences and conformations that would have been ignored by a rigid-backbone model.

As an adjunct to comparisons of sequence diversity, it can also be insightful to examine the conformational diversity. One example that illustrates the complementary capabilities of the two algorithms with backbone flexibility is the sequence predicted most often by BRDEE, 236L/239L (represented by 74 low-energy conformations in BRDEE and 68 in BD). The rotamers identified at all flexible positions are similar, overall between BD and BRDEE, but BRDEE allows an additional rotamer for Ile299. A closer examination of the structures generated by both algorithms reveals why this is the case. Ile299 is situated on a β-strand, and the new rotamer (*tt*) has a different χ_1 from the other rotamers (*mm*, *mp*, *mt*) (Lovell *et al.*, 2000) identified by the algorithms. This would normally cause the C_{γ_2} atom to clash with Thr278 on an adjacent β-strand. However, BRDEE enables these two strands to move away from each other locally, whereas in the BD conformations, the strands were observed to move together in the same direction, as part of a coordinated shift of the entire active site's backbone. This implies that BRDEE can facilitate small-scale, anti-correlated backbone motions, whereas the strength of BD is large-scale, correlated backbone motions.

To quantify the effect of incorporating backrubs on the conformational energies, we further analyzed the set of 605 low-energy conformations (Table 2) returned by BRDEE. For this analysis only, the ligand was removed from the complex, since its rotation/translation could also impact the conformational energies. The computed energy decrease resulting from backrub minimization (Step 4 from Section 3) varied significantly between conformations, ranging from no effect to an improvement of almost 80 kcal/mol (data not shown). However, in 462 of the 605 conformations (76%),

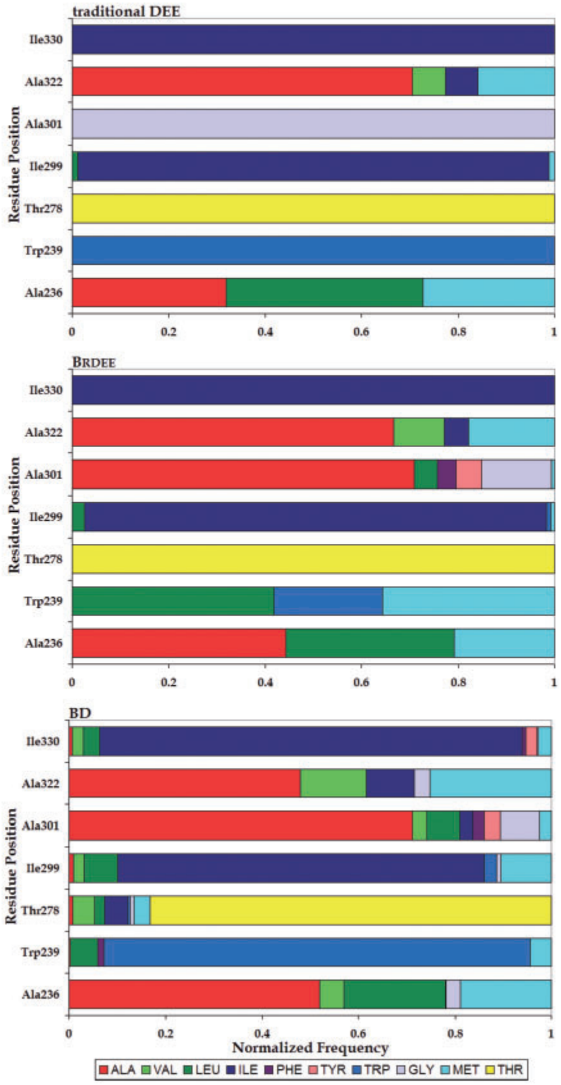

Fig. 3. Distribution of Mutations: GrsA-PheA redesign for the non-cognate substrate Leu. The distribution of mutations for all conformations with an energy within 5 kcal/mol from the lowest BRDEE energy (Table 2) is shown for: traditional DEE (top), BRDEE (middle), and BD (bottom).

allowing backrubs resulted in a decrease of the conformational energy, and in 261 of these conformations, the decrease in energy was >1 kcal/mol, thus confirming the potential benefit of including backrubs in protein design.

For the GrsA-PheA redesign, the application of the steric and τ filters from Step 1 of Section 3 reduced the possible backrub combinations by a factor of 20, to a total of 3888. The BRDEE pruning stage reduced the number of possible rotameric conformations from 1.99×10^{12} to 4.78×10^{9}. Including the pairwise energy precomputation, the entire redesign took almost a day on a cluster of 10 processors. In contrast, traditional DEE (for a single rigid backbone) completed in ~45 min on 10 processors. Thus, the incorporation of backbone flexibility and the provable algorithmic guarantees significantly reduce the computational efficiency of BRDEE when compared to traditional DEE for a *single* rigid backbone. However, as a simple comparison, performing traditional DEE separately for *each* of the 3888 backbones (assuming similar CPU times for each backbone), would require ~120 days on the same number of processors. This implies BRDEE is approximately two orders of magnitude faster, although further benchmarks would be necessary to determine its precise computational benefits.

5.2.2 Redesign of Gβ1. The results from the Gβ1 redesign (Table 2) show a similar trend to the GrsA-PheA results. BRDEE identified a significantly larger number of low-energy sequences and conformations than traditional DEE; similarly, BD identified a larger number of low-energy sequences and conformations than BRDEE. The application of the steric and τ filters (Step 1 of Section 3) reduced the possible backbone combinations by a factor of more than 3000. The BRDEE pruning stage reduced the number of possible rotameric conformations by a factor of more than 10^7. Including the pairwise energy precomputation, the BRDEE redesign required 2 weeks on a cluster of 16 processors, as compared to ~3 h for traditional DEE for a single rigid backbone. However, without the pruning filters of our backrub algorithm, a mutation search over backrub/rotamer space would be computationally infeasible.

To determine whether BRDEE introduces unreasonable bond strain, we compared the τ-angle distributions for the residues affected by backrubs in the top 605 (GrsA-PheA) and 599 (Gβ1) BRDEE-generated structures against all residues in the respective crystal structures (in which τ-angles are based on direct experimental evidence and can be assumed to be energetically tolerable). The mean τ-values for the sets of BRDEE conformations are less than one standard deviation and not more than 0.3° from those in their respective crystal structures (data not shown). In addition, all τ-values are very close to the ideal value of 111°. Therefore, we can conclude that BRDEE introduces only small τ changes that do not cause significant strain.

6 CONCLUSION

In this article, we presented an algorithm for protein design that incorporates local backbone flexibility via backrub motions. As confirmed by the redesigns of the GrsA-PheA active site and the core of Gβ1, the additional flexibility provided by the backrub algorithm allows the identification of a large number of low-energy sequences and conformations that would have otherwise been ignored by a rigid-backbone model. Such an expansion in the predicted sequence/conformation space for rational protein design likely implies that backrubs may play an important evolutionary role as well; further computational and experimental validation will be necessary to confirm this hypothesis.

BRDEE only requires that the three most expensive steps of the design algorithm be performed once, simultaneously for all backbones (Section 1.3, paragraph 3). Preliminary benchmark tests (Section 5.2) indicate that BRDEE obtains significant computational benefits, as compared to applying traditional DEE separately for each backbone. However, further experiments on more proteins will be necessary to determine the precise benefits of BRDEE.

Backrubs represent local motions, whereas in the BD algorithm, backbone flexibility is represented by global motions. BRDEE and BD are thus complementary in nature, so combining these two backbone flexibility approaches within a single protein design algorithm presents interesting future work. The main challenge for such an algorithm will be to overcome the combinatorial explosion resulting from the consideration of all possible backbone conformations. Preliminary evidence from our alternate conformation recovery experiments suggests that BRDEE may be sensitive to the coarseness of the rotamer library. Hence, expanding the accessible side-chain conformation space by incorporating an extended rotamer library or by allowing continuous flexible rotamers (Georgiev *et al.*, 2008) may prove important for further improving the algorithm's predictions. Using the tools introduced in this article, both extensions are expected to similarly yield provable algorithms.

ACKNOWLEDGEMENTS

We thank J. Groh and all members of the Donald and Richardson labs for helpful discussions and comments.

Funding: This work was funded by National Institutes of Health (R01 GM-65982 and R01 GM-78031 to B.R.D., R01 GM-073930 to D.C.R.).

Conflict of Interest: none declared.

REFERENCES

Chazelle,B. *et al.* (2004) A semidefinite programming approach to side-chain positioning with new rounding strategies. *INFORMS J. Comput. Comput. Biol. Special Issue*, **16**, 380–392.

Conti,E. *et al.* (1997) Structural basis for the activation of phenylalanine in the non-ribosomal biosynthesis of Gramicidin S. *EMBO J.*, **16**, 4174–4183.

Cornell,W. *et al.* (1995) A second generation force field for the simulation of proteins, nucleic acids and organic molecules. *JACS*, **117**, 5179–5197.

Davis,I.W. *et al.* (2006) The backrub motion: how protein backbone shrugs when a sidechain dances. *Structure*, **14**, 265–274.

De Maeyer,M. *et al.* (1997) All in one: a highly detailed rotamer library improves both accuracy and speed in the modelling of sidechains by dead-end elimination. *Fold. Design*, **2**, 53–66.

Desjarlais,J.R. and Handel,T.M. (1999) Side-chain and backbone flexibility in protein core design. *J. Mol. Biol.*, **289**, 305–318.

Desmet,J. *et al.* (1992) The dead-end elimination theorem and its use in protein side-chain positioning. *Nature*, **356**, 539–542.

Dunbrack,R.L. (2002) Rotamer libraries in the 21st century. *Curr. Opin. Struct. Biol.*, **12**, 431–440.

Esposito,L. *et al.* (2000) The ultrahigh resolution crystal structure of ribonuclease A containing an isoaspartyl residue: hydration and sterochemical analysis. *J. Mol. Biol.*, **297**, 713–732.

Fenn,T.D. *et al.* (2004) Xylose isomerase in substrate and inhibitor michaelis states: atomic resolution studies of a metal-mediated hydride shift. *Biochemistry*, **43**, 6464-6474.

Fung,H.K. *et al.* (2007) Novel formulations for the sequence selection problem in de novo protein design with flexible templates. *Optim. Method. Softw.*, **22**, 51–71.

Georgiev,I. and Donald,B.R. (2007) Dead-end elimination with backbone flexibility. *Bioinformatics*, **23**, i185–i194.

Georgiev,I. *et al.* (2006) Improved pruning algorithms and divide-and-conquer strategies for dead-end elimination, with application to protein design. *Bioinformatics*, **22**, e174–e183; In *Proceedings of the International Conference on Intelligent Systems for Molecular Biology* (ISMB), Fortaleza, Brazil.

Georgiev,I. *et al.* (2008) The minimized dead-end elimination criterion and its application to protein redesign in a hybrid scoring and search algorithm for computing partition functions over molecular ensembles. *J. Comput. Chem.*, Feb 21 [Epub ahead of print]. PMID: 18293294.

Goldstein,R. (1994) Efficient rotamer elimination applied to protein side-chains and related spin glasses. *Biophys. J.*, **66**, 1335–1340.

Gordon,D.B. *et al.* (2003) Exact rotamer optimization for protein design. *J. Comput. Chem.*, **24**, 232–243.

Gordon,D.B. *et al.* (1999) Energy functions for protein design. *Curr. Opin. Struct. Bio.*, **9**, 509–513.

Harbury,P.B. *et al.* (1998) High-resolution protein design with backbone freedom. *Science*, **282**, 1462–1467.

Havranek,J.J. and Harbury,P.B. (2003) Automated design of specificity in molecular recognition. *Nat. Struct. Biol.*, **10**, 45–52.

Hu,X. and Kuhlman,B. (2006) Protein design simulations suggest that side-chain conformational entropy is not a strong determinant of amino acid environmental preferences. *Proteins*, **62**, 739–748.

Jiang,L. *et al.* (2008) De Novo computational design of retro-aldol enzymes. *Science*, **319**, 1387–1391.

Jin,W. *et al.* (2003) De novo design of foldable proteins with smooth folding funnel: Automated negative design and experimental verification. *Structure*, **11**, 581–591.

Korkegian,A. *et al.* (2005) Computational thermostabilization of an enzyme. *Science*, **308**, 857–860.

Kortemme,T. *et al.* (2004) Computational redesign of protein-protein interaction specificity. *Nat. Struct. Mol. Biol.*, **11**, 371–379.

Kuhlman,B. and Baker,D. (2000) Native protein sequences are close to optimal for their structures. *Proc. Natl Acad. Sci.*, **97**, 10383–10388.

Kuhlman,B. *et al.* (2003) Design of a novel globular protein fold with atomic-level accuracy. *Science*, **302**, 1364–1368.

Lassila,J.K. *et al.* (2006) Combinatorial methods for small-molecule placement in computational enzyme design. *Proc. Natl Acad. Sci. USA*, **103**, 16710–16715.

Lasters,I. and Desmet,J. (1993) The fuzzy-end elimination theorem: correctly implementing the side chain placement algorithm based on the dead-end elimination theorem. *Protein Eng.*, **6**, 717–722.

Lazaridis,T. and Karplus,M. (1999) Effective energy function for proteins in solution. *Prot. Struct. Funct. Genet.*, **35**, 133–152.

Leach,A. and Lemon,A. (1998) Exploring the conformational space of protein side chains using dead-end elimination and the A* algorithm. *Proteins*, **33**, 227–239.

Lilien,R. *et al.* (2005) A novel ensemble-based scoring and search algorithm for protein redesign, and its application to modify the substrate specificity of the Gramicidin Synthetase A phenylalanine adenylation enzyme. *J. Comput. Biol.*, **12**, 740–761.

Lippow,S.M. *et al.* (2007) Computational design of antibody-affinity improvement beyond in vivo maturation. *Nat. Biotechnol.*, **25**, 1171–1176.

Looger,L. and Hellinga,H. (2001) Generalized dead-end elimination algorithms make large-scale protein side-chain structure prediction tractable: implications for protein design and structural genomics. *J. Mol. Biol.*, **307**, 429–445.

Looger,L. *et al.* (2003) Computational design of receptor and sensor proteins with novel functions. *Nature*, **423**, 185–190.

Lovell,S.C. *et al.* (2000) The penultimate rotamer library. *Proteins*, **40**, 389–408.

Lovell,S.C. *et al.* (2003) Structure validation by C_α geometry: ϕ, ψ and C_β deviation. *Proteins*, **50**, 437–450.

Malakauskas,S.M. and Mayo,S.L. (1998) Design, structure and stability of a hyperthermophilic protein variant. *Nat. Struct. Biol.*, **5**, 470–475.

Mendes,J. *et al.* (1999) Improved modeling of side-chains in proteins with rotamer-based methods: a flexible rotamer model. *Proteins*, **37**, 530–543.

Murshudov,G.N. *et al.* (2002) The structures of Micrococcus lysodeikticus catalase, its ferryl intermediate (compound II) and NADPH complex. *Acta. Crystallog. D. Biol. Crystallogr.*, **58**(Pt 12), 1972–1982.

Nukaga,M. *et al.* (2003) Ultrahigh resolution structure of a class A beta-lactamase: on the mechanism and specificity of the extended-spectrum SHV-2 enzyme. *J. Mol. Biol.*, **328**, 289–301.

Pierce,N. and Winfree,E. (2002) Protein design is *NP*-hard. *Prot. Eng.*, **15**, 779–782.

Pierce,N. *et al.* (2000) Conformational splitting: a more powerful criterion for dead-end elimination. *J. Comput. Chem.*, **21**, 999–1009.

Ponder,J. and Richards,F. (1987) Tertiary templates for proteins: use of packing criteria in the enumeration of allowed sequences for different structural classes. *J. Mol. Biol.*, **193**, 775–791.

Richardson,J.S. and Richardson,D.C. (1988) Amino acid preferences for specific locations at the ends of alpha helices. *Science*, **240**, 1648–1652.

Stevens,B. *et al.* (2006) Redesigning the PheA domain of Gramicidin Synthetase leads to a new understanding of the enzyme's mechanism and selectivity. *Biochemistry*, **45**, 15495–15504.

Street,A. and Mayo,S. (1999) Computational protein design. *Structure*, **7**, R105–R109.

Su,A. and Mayo,S.L. (1997) Coupling backbone flexibility and amino acid sequence selection in protein design. *Prot. Sci.*, **6**, 1701–1707.

Vizcarra,C.L. and Mayo,S.L. (2005) Electrostatics in computational protein design. *Curr. Opin. Chem. Biol.*, **9**, 622–626.

Voigt,C.A. *et al.* (2000). Trading accuracy for speed: a quantitative comparison of search algorithms in protein sequence design. *J. Mol. Biol.*, **299**, 789–803.

Voigt,C.A. *et al.* (2001). Computational method to reduce the search space for directed protein evolution. *Proc. Natl Acad. Sci. USA*, **98**, 3778–3783.

Weiner,S. *et al.* (1984) A new force field for molecular mechanical simulation of nucleic acids and proteins. *J. Am. Chem. Soc.*, **106**, 765–784.

Yanover,C. *et al.* (2007) Dead-end elimination for multistate protein design. *J. Comput. Chem.*, **28**, 2122–2129.

Zanghellini,A. *et al.* (2006) New algorithms and an in silico benchmark for computational enzyme design. *Prot. Sci.*, **15**, 2785–94.