CHAINTWEAK: SAMPLING FROM THE NEIGHBOURHOOD OF A PROTEIN CONFORMATION

ROHIT SINGH and BONNIE BERGER*†

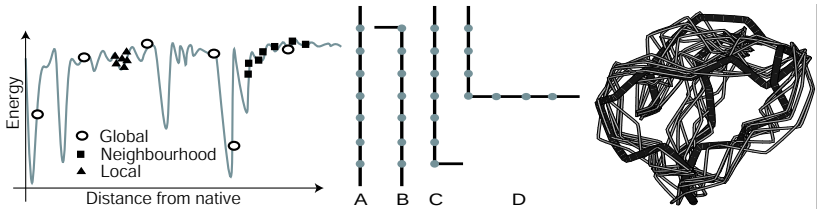
Computer Science and Artificial Intelligence Laboratory
Massachusetts Institute of Technology
Cambridge MA 02139
E-mail: {rsingh, bab}@mit.edu

When searching for an optimal protein structure, it is often necessary to generate a set of structures similar, e.g., within 4Å Root Mean Square Deviation (RMSD), to some *base* structure. Current methods to do this are designed to produce only small deviations ($<$ 0.1Å RMSD) and are inefficient for larger deviations. The method proposed in this paper, ChainTweak, can generate conformations with larger deviations from the base much more efficiently. For example, in 18 seconds it can generate 100 backbone conformations, each within 1-4Å RMSD of a given 45-residue conformation. Moreover, each conformation has correct bond lengths, angles and omega torsional angles; its phi-psi angles have energetically favorable values; and there are rarely any backbone steric clashes. The method uses the insight that loop closure techniques can be used to perform compensatory changes of dihedral angles so that only a part of the conformation is changed. It is demonstrated, using decoys from the Decoys 'R Us data-set, that ChainTweak can be used to construct good decoys. It also provides a novel and intuitive way of analyzing the energy landscape of a protein. In addition, ChainTweak can improve the accuracy and performance of the loop modeling program RAPPER by an order of magnitude (1.1 min. vs. 36 min. for an 8-residue chain).
Availability & Supp. Info.: `http://theory.csail.mit.edu/chaintweak`

1. Introduction

A fundamental axiom of molecular biology is that the function of a protein is determined by its structure. In turn, most protein structure determination problems are, essentially, search problems. In some of these, e.g., homology modeling or protein re-design, the problem specification may restrict the search to the neighbourhood of some template structure. In other cases, restricting the search to the neighbourhoods of a set of candidate structures might just be a solution strategy (e.g., in the Rosetta[4] method for *ab-initio* folding). Here, the *neighbourhood* of a structure is the set of structures similar to it. For example, the set of all structures within, say, 4Å RMSD of a base structure could be defined as its neighbourhood[a].

*Corresponding author
†Also in the MIT Dept. of Mathematics
[a]Of course, the size of the neighbourhood and, consequently, the exact choice of a RMSD threshold would depend on the problem instance and the size of the protein. Typically,



(a) (b) (c)

Figure 1: **(a)** A cartoon illustrating the space coverage differences between global, neighbourhood, and local search. Observe that local search techniques can only cover the basin on one local minima. **(b)** Cartoon illustrating that changes in dihedral angles near the terminal regions of a chain (A) result in small perturbations (B and C), while changing an angle in the middle of the chain results in a very large perturbation (D). **(c)** Example output from ChainTweak. Ten conformations from the neighbourhood of a 32-residue protein structure (PDB:1clv, chain I) were sampled and aligned with the original. The original structure is in black, the others are in gray.

Efficiently searching in the neighbourhood of a possible protein structure (conformation) is thus an important and frequently recurring problem. As the term "neighbourhood search" signifies, this search problem is different from global or local search problems (Fig 1a), even though it has usually been studied as an extreme case of these. This paper focuses on the sampling component of this search problem and presents a method, ChainTweak, for efficiently and representatively sampling from a given neighbourhood.

Many different approaches to neighbourhood sampling have been tried. High-temperature Molecular Dynamics (MD) methods have been used to generate structures with 2-4Å RMSD from the native[1]. Methods based on discrete off-lattice models[2,3] discretize the dihedral-angle space and try out different combinations. Similarly, in Monte Carlo (MC) search methods, various move-sets have been developed for making local moves. For example, fragment-swap MC in Rosetta[4] relies on using a database of polypeptide fragments to swap one fragment for another, as long as their ends match. Another set of approaches, such as in torsional dynamics[5], or the MC-based methods proposed by Ulmschneider & Jorgensen[6] and Cahill et al.[7] use geometric insights to perform such local modifications.

Our proposed neighbourhood sampling method, ChainTweak, has many advantages over existing methods. Rather than being closely tied to some search strategy (or an energy function), it is a stand-alone method that can be used by researchers as a black-box, allowing them to focus on other parts of the search problem (e.g., energy function design[8]). Moreover, ChainTweak

for a 50-residue protein, two conformations within 2Å of each other are considered almost identical. Thus, in this case, the threshold size should be $\geq$ 2Å.

enables fast generation of ensembles (sets of conformations) centered around any given base conformation. The flexibility of ChainTweak enables novel applications (e.g., energy function analysis) and enhances the performance of existing applications (see Section 5).

1.1. *Neighbourhood Sampling: The Right Representation?*

Almost all neighbourhood sampling methods work by perturbing the base conformation's structure to generate conformations in its neighbourhood. To model the structure, these methods use either an all-atoms Cartesian coordinates based model or a dihedral angles based model.

Most existing methods use the Cartesian coordinate based model. With this model, however, an energy minimization step is needed to restore correct bond lengths/angles in the perturbed structures. Efficiency and convergence issues with this step limit the size of a single perturbation step (< 0.1Å^9). Thus, only a small neighbourhood around the base can be explored. For larger deviations, successive perturb-and-minimize operations, using an MD-like approach[1], can be done. However, generating many MD trajectories, to ensure representative sampling, may become computationally expensive.

In contrast, representing the protein backbone by its dihedral angles offers distinct advantages. All conformations sampled from the neighbourhood will then have different dihedral angles but the same bond lengths/angles. Since the latter can always be set to their desired/ideal values, no minimization step is necessary. Hence, the restriction on small perturbation sizes is removed. However, modifying a dihedral angle at residue i changes the positions of all residues $i + 1$ onwards. As a result, the perturbed structure may deviate so far from the base as to not be in the neighbourhood at all, especially if residue i is in the middle of the chain (Fig 1b). This problem is the major stumbling block in using a dihedral angles based representation.

One way to solve this problem, e.g., in Torsional Dynamics (TD)[5], is compensatory modification of multiple torsional angles such that the overall structural deviation is acceptably small. However, the differential calculus-based methods used by TD algorithms work well only for small perturbations. Moreover, the sampling behavior is effected by the energy function chosen for the TD simulations. The reader might also notice the parallel here with the loop closure problem where one needs to find small chains joining two fixed ends. Indeed, our proposed algorithm, ChainTweak, exploits this parallel.

1.2. *Contributions*

ChainTweak is an algorithm for efficiently sampling from the neighbourhood of a given base conformation. It generates a set of backbone conforma-

tions such that each new conformation has the following properties: it lies in a neighbourhood of the base; it has the terminal (first and last) residues fixed in the same relative positions as the base; and it has bond lengths/angles set to their desired/ideal values. In Section 2 we describe a simple extension that allows the positions of terminal residues to vary as well.

ChainTweak iteratively perturbs the base conformation using the dihedral angle representation. A sliding window approach is used to successively move some atoms by 0–2Å while keeping all others fixed. Inside the window, loop closure methods are used to generate such perturbations. Moreover, residue-specific phi-psi angle preferences can be used to choose a perturbation.

We show that ChainTweak can explore large neighbourhoods efficiently. Given a conformation of a 45-residue protein, in 18 seconds it can generate 100 backbone conformations, each within 1-4Å RMSD of the base. Moreover, by running ChainTweak for more iterations larger neighbourhoods can be explored: for this protein, a conformation with RMSD of 12Å from the base can be found. In contrast, after 18 seconds, an MD simulation (run using TINKER[9]) produces a single conformation for the same protein (with 0.91Å RMSD). Even theoretically, ChainTweak's running time is asymptotically optimal— linear in the length of the chain and the number of samples desired.

We also describe some applications of ChainTweak (Section 4.2). It improves upon the performance of some existing applications (decoy generation and *ab-initio* loop-modeling using RAPPER) and also enables novel applications (energy function analysis in an intuitive way).

2. Algorithm

Here we present the algorithm ChainTweak that has the following input and output:
Input: A single backbone conformation $\mathcal{C}_0$ described by its bond lengths, bond angles and dihedral angles.
Output: N conformations such that the RMSDs of these conformations w.r.t. $\mathcal{C}_0$ roughly follow a desired distribution. For example, half of the output conformations are 0–2Å RMSD from the base while the rest are 2–4Å RMSD from the base. For each output conformation, the bond lengths, bond angles and the relative positions of the end-residues are the same as in $\mathcal{C}_0$.

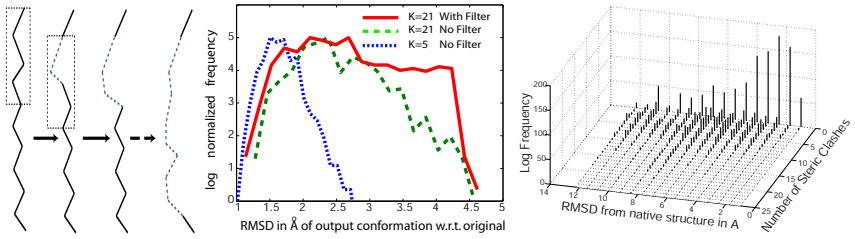
The initial restriction on preserving the relative positions of the end-residues can be adapted for flexible chain ends by pre-processing $\mathcal{C}_0$ to produce a set of conformations with randomly sampled values for dihedral angles at the end-residues. Recall that modifying dihedral angles at the ends only results in local structure changes (Fig 1b). Each of these conformations then becomes the input to a separate ChainTweak instance.

Observe that by iteratively setting each output conformation as the input of a new ChainTweak problem, more solutions can be found for the original ChainTweak problem. Also, the problem can be *recursively* solved by splitting the input chain into two sub-chains and concatenating the respective solutions. We do this until we have a chain small enough to be solved using loop-closure techniques. The pre-processing step (moving the chain ends) mentioned previously is required only at the top level of recursion, i.e., for the full-length chain.

The loop closure problem was informally discussed by Robert Diamond[14] and was formally defined by Go and Scheraga[15]. The input in such a problem is the relative position of two fixed residues (anchors) at each end and the goal is to find different possible conformations for a polypeptide chain of length m joining the fixed ends. For a problem instance with 6 unknown dihedral angles, i.e. 6 degrees of freedom (DOFs), the maximum number of possible solutions is 16. With more DOFs, the number of solutions is infinite. In the 6-DOF case, Manocha et al.[16] applied inverse kinematics techniques from robotics to numerically generate all possible 16 solutions. More recently, Wedemeyer and Scheraga[17] and Coutsias et al.[18] have also presented analytic solutions for the 6-DOF problem. ChainTweak can use any of these as a subroutine (Algorithm 3 in Supp. Info.).

ChainTweak iteratively calls the subroutine SlideWin (Algorithm 2 in Supp. Info.). Given a starting backbone conformation, SlideWin finds a new backbone conformation using a sliding window approach (Fig 2a). A window of 3 residues (9 points) is chosen. After fixing 3 points on both ends, this results in a 6-DOF loop closure problem. We use Manocha et al.'s algorithm when omega angles are unrestricted and Coutsias et al.'s algorithm when omega angles have to be restricted to particular values (say, 180°). A wrapper around these routines (LoopClsr6, Algorithm 3) suggests up to 15 alternative conformations for the conformation inside the window. Of these, we randomly select one conformation, biasing our choice towards a conformation that has phi-psi angles in favorable/acceptable regions of the Ramachandran Plot (Fig 3). Residue and secondary structure information can thus be encoded by designing appropriate phi-psi preference maps.

A single iteration of SlideWin moves each residue by about 0.5–1.5Å. ChainTweak (Algorithm 1 in Supp. Info.) iteratively applies SlideWin K times to achieve a much larger deviation from the starting conformation; the output conformations of one iteration form the input for the next. Between each iteration, some conformations may be pruned out, depending on their RMSD from the original structure. The exact pruning policy is described by



(a) (b) (c)

Figure 2: **(a)** A cartoon describing SLIDEWIN. Inside each window, LOOPCLSR6 is used to perform the tweak. Observe that the first and last positions in the window are not changed, both in LOOPCLSR6 and SLIDEWIN (see Supp. Info.). **(b)** A plot showing the frequency distribution of ChainTweak-generated conformations vs. their RMSD w.r.t the base. The parameters K and `Filter` can be used to control structural variation in the output set. A low value of K (=5) results in conformations that are similar to the the original. $K = 21$ resulted in greater structural variation. `Filter` was used to ensure that the distribution was "more even". The frequencies of each distribution have been scaled so that the maximum is same across all three. **(c)** A plot showing the frequency distribution of ChainTweak-generated conformations, classified by the number of steric clashes per conformation and its RMSD from the native. 10000 backbone conformations from the neighbourhood of a 45-residue structure (PDB 1bh9:31-75) were generated. For each conformation, the number of backbone steric clashes, using a cutoff of 2Å, were counted. Most of the conformations, even those with large RMSDs from the base, have no steric clashes. Note that the frequencies are shown on a log scale.

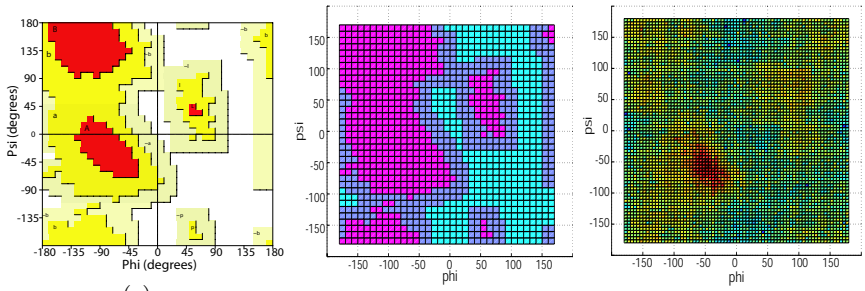
the user-specified parameter `Filter` (described below) and helps in achieving a desired structural variation in the final solution set (Fig 2b).

3. Results

3.1. *Performance Analysis*

The size of the neighbourhood explored by ChainTweak, measured in RMSD from the base, is controlled by the number of iterations, K. In our simulations, we observed that this size increases from 2.5Å, for $K = 5$, to about 4.5Å, for $K = 21$ (Fig 2b). ChainTweak can explore rather large neighbourhoods: for a 45-residue protein it can generate a conformation with 12Å RMSD from the base.

Another user-specified parameter, `Filter`, can be used to control the structural variation in ChainTweak's output by describing a pruning policy. An example pruning strategy (Fig 2b) is to remove enough structures, after every 4^{th} iteration, such that the RMSDs (w.r.t. the base) of the remaining structures are uniformly distributed. Without any pruning, the output set's composition is skewed towards structures with low RMSD (approx 1-2Å) from the base. This is understandable— having performed a tweak operation on a conformation, a second tweak operation is as likely to take it further away from the original as it is to bring it back closer to the original. Analogously,



(a) (b) (c)

Figure 3: **(a)** Using a reference Ramachandran Plot, **(b)** we implemented a simple phi-psi priority scheme: (red: favorable, allowed) > (dark blue: generously allowed) > (light blue: others). **(c)** For 10000 conformations of a 45-residue protein (PDB: 1bh9,31-75) generated by ChainTweak, the phi-psi distributions match well with the specified priorities. This protein has 2 alpha-helices which explains the higher frequency of phi-psi angles in regions of the plot corresponding to alpha-helical structure.

recall that in a 1-D random walk, the probability of being at distance d from the origin decreases exponentially with d.

Can ChainTweak representatively sample from the entire neighbourhood? Some recent theoretical work[19,20] on the folding of polygonal chains suggests that any two protein backbone conformations (with same bond lengths/angles) can be converted into each other by simply changing the dihedral angles. This suggests that ChainTweak can explore the entire neighbourhood. Also, observe that the "tweak" operation of SlideWin is essentially a random walk in this neighbourhood. This, in turn, suggests that ChainTweak's sampling is representative.

ChainTweak is efficient in both practice and theory: for a chain of length n with N output conformations, the running time of ChainTweak is $O(Nn)$. It is dominated by the approximately $KNn/3$ calls to LoopClsr6. The actual time spent per call of LoopClsr6 does not vary much (avg: 8.3 millisecs; std dev: 3.6 millisecs on a Pentium-4 2.4GHz PC). Also, observe that just writing the output (N conformations, each of size $O(n)$) would take $O(Nn)$ time. Hence, ChainTweak is an *asymptotically* optimal algorithm.

ChainTweak has high numerical accuracy. Its implementation avoids error accumulation (see Supp. Info. for details). For example, the deviation of atom positions in the terminal residues is negligible: avg error = 0.001Å.

Conformations generated by ChainTweak have very few backbone steric clashes (Fig 2c). This is probably because, in all our experiments, the base conformation did not have any steric clashes and the output conformations are similar to the base. After the addition of sidechains to the generated backbone conformations, both new sidechain and old backbone steric clashes, if

	ChainTweak + RAPPER					RAPPER only				
		Best generated (RMSD)					Best generated (RMSD)			
	Time	Backbone		C_α	C Anchor	Time	Backbone		C_α	C Anchor
Length	(min.)*	Global	Local	Local	(RMSD)†	(min.)‡	Global	Local	Local	(RMSD)
8	0.7(1.1)	1.40	0.92	0.93	0.02	36.4	1.11	0.70	0.56	0.30
9	1.1(1.5)	1.61	1.07	1.12	0.03	30.5	1.29	0.81	0.72	0.33
10	1.3(1.7)	2.02	1.24	1.31	0.01	44.15	1.67	1.11	1.00	0.41
11	1.7(2.3)	2.45	1.49	1.61	0.04	59.17	1.99	1.27	1.23	0.33
12	2.1(3.1)	2.21	1.56	1.72	0.02	100.4	2.21	1.47	1.46	0.54

Table 1: ChainTweak can improve upon the performance of the loop modeling program RAPPER. The latter's performance in generating 1000 loop conformations for loops of various lengths has been measured using the FISER dataset[10]. From the same dataset, for chain lengths between 8-12 residues, we picked 20 chains each. For each of these, a representative set (in terms of their RMSDs from the native conformation) of 10 conformations was picked from the RAPPER-generated set. Using ChainTweak, 100 conformations in the neighbourhood of each such conformation were sampled. As in ref. (10), the quality of these 1000-conformation ensembles is measured in terms of the smallest Global (only loop ends aligned) and Local (whole chain aligned) RMSD of any conformation w.r.t. to the native (averaged across all 20 chains) and the deviation of C-terminal loop ends from the desired position. [$\star$] The time in parentheses includes the estimated cost of generating 10 conformations using RAPPER. [†] Before running ChainTweak, the chosen RAPPER-generated conformations were fixed, if possible, so that their ends matched those of the native. [‡] To account for differences in processing power (2.4GHz for us vs. 900MHz in ref. (10)), these running times are one-fourth of the actual times reported in ref. (10).

any, can be relieved simultaneously. Hence, we decided against explicitly checking for steric clashes in ChainTweak. In case such checks become necessary, they can be done efficiently by taking advantage of ChainTweak's incremental modification approach and using kinetic data structures[22] or Lotan et al.'s hierarchical approach[23].

ChainTweak generates structures where most of the phi-psi angles have favorable values. A random 6-DOF chain with fixed ends has multiple alternative conformations. Residue and secondary-structure related phi-psi preferences can be used to pick the most appropriate alternative. We encoded a simple phi-psi preference map which accorded higher priority to any phi-psi combination lying in favorable and acceptable regions of the Ramachandran Plot. Even this simple map yielded impressive results (Fig 3).

3.2. *Applications*

Loop Modeling: ChainTweak can be used to supplement an *ab-initio* loop modeling program like RAPPER (DePristo et al.[10]). The latter generates loop conformations by sampling in a discretized phi-psi angle space and then using a dihedral angle-based minimizer to ensure that the position of loop ends is roughly unchanged. The method is computationally expensive

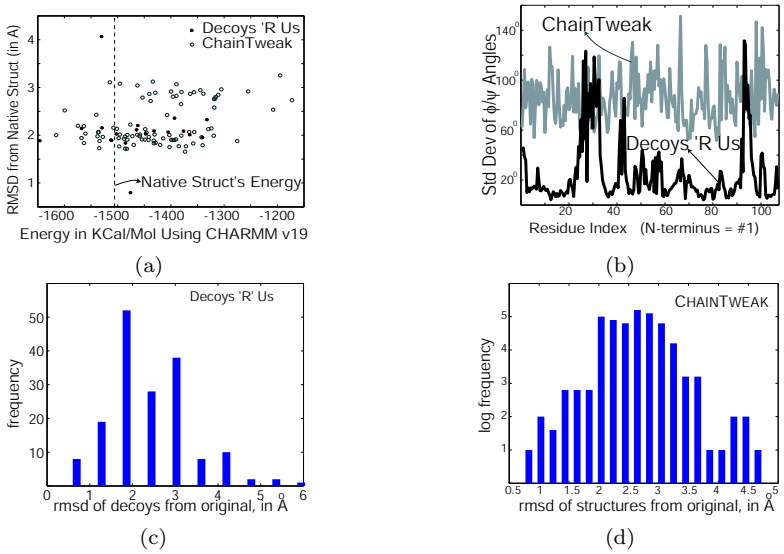
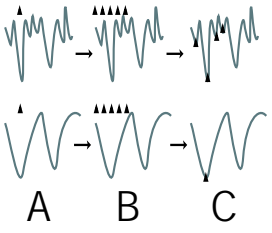


Figure 4: **(a,b)** Decoys of a protein domain (PDB 1mfa:1-111, ig_structal_hires in Decoys 'R Us) were extracted from the Decoys 'R Us Database (DB) and were also created using ChainTweak (CT). Both the sets, as well as the native, were minimized using the CHARMM v19 energy function and the TINKER package. **(a)** Post-minimization energies and C_α-RMSD from the native structure are plotted to illustrate that both DB and CT decoy-sets manage to "fool" the energy function and are structurally similar to the native structure. **(b)** To identify regions with local structural variation, we measured the standard deviation of phi-psi angles along the chain. This indicates that the CT set is more representative: its local structural variation is not limited to a few regions. **(c,d)** Comparison of decoys produced by DB (c) and CT (d) for the loop region 1vfa:158-166. The structural variation among the decoys, as measured by C_α-RMSD to the native conformation, is comparable across the two sets

because all conformations with incorrect positions of the loop ends have to be rejected. In contrast, ChainTweak only generates conformations that have the loop ends in the right positions.

ChainTweak can be used to efficiently expand a small ensemble generated by RAPPER, thus improving overall efficiency by an order of magnitude (Table 1). The ensembles generated by the two methods are of comparable quality (as measured by the RMSD to the native conformation). On one important criterion, that of fixing the positions of the loop ends, ChainTweak actually performs much better.

Decoy Generation: Decoys[2] are non-native structures that can be used to design energy functions[8] capable of distinguishing such structures from the native. We have found that ensembles generated by ChainTweak can be used to generate decoys, especially those that are globally similar to the



Energy Function	w.r.t. the native's minimum, # of minima with < 0.1Å RMSD	< 0.5Å RMSD	largest RMSD
Amber '99 + GB/SA	6 of 10	10 of 10	0.32Å
Charmm v19 + GB/SA	5 of 10	10 of 10	0.16Å
OPLS + GB/SA	3 of 10	10 of 10	0.25Å
Charmm v27 + GB/SA	3 of 10	10 of 10	0.39Å
Charmm v19	2 of 10	5 of 10	4.38Å
Charmm v27	0 of 10	4 of 10	2.33Å
Amber '99	0 of 10	4 of 10	1.72Å
OPLS	0 of 10	0 of 10	3.17Å

Figure 5: **Figure:** This figure illustrates how ChainTweak can be used for analyzing energy landscapes. (A) Given some decoy and two candidate energy functions, (B) ChainTweak can sample from the neighbourhood of the decoy. (C) The generated conformations are minimized and the distribution of local minima provides information about the energy landscape. Note that the same set of conformations is used for each energy function.
Table: ChainTweak was used to generate 10 conformations similar (within 0.003-0.13Å RMSD) to an alpha-helix (PDB 2gib:22-37). Eight different energy functions were used to minimize the ensemble conformations, resulting in 10 local minima per function (see Supp. Info. for details). For each function, the RMSDs of these local minima from the local minimum corresponding to the native structure were measured. An energy function ranked higher if it had more local minima with a very low (< 0.1Å) RMSD from the minimum corresponding to native structure. As can be seen, the addition of a solvation term (GB/SA[24]) improves the performance of these energy functions.

native structure but have significant local differences from it. As the use of homology modeling to predict structure increases, the need for such decoys will increase. We used some loop decoy-sets and some homology modeling based decoy-sets (HM) from the the Decoys 'R Us database (Samudrala and Levitt[12]) to evaluate ChainTweak-generated decoys.

ChainTweak's HM decoys are comparable to the database decoys in terms of their energy-vs-RMSD profile. CT decoys are more representative, i.e, their local structural variation is not limited to a few small regions (Fig 4a, 4b). With HM decoys, the use of homology forces biased sampling: the local structural variation across them is limited to a few regions. With ChainTweak, the user has the option to either emulate such behavior (by applying ChainTweak only on specific parts of the chain) or have equal local variation through-out the entire chain (Fig 4b).

We also compared ChainTweak-generated loop decoys against some loop decoy-sets from the database. The former performed comparably with database decoys in terms of their structural characteristics (Fig 4c, 4d). They performed significantly better on the criteria of preserving the positions of loop ends (~ 0.01Å deviation vs. ~ 0.5Å deviation).

Energy Landscape Analysis: As discussed by Keasar and Levitt[21], well-designed functions should have wide basins and few local minima so that

structurally similar conformations are minimized to the same local minima (Fig 5). Ensembles generated by ChainTweak can be used to analyze the energy landscape of any energy function f around a protein structure b: after each conformation in the ensemble is minimized, the distribution of these local minima and their proximity to the base provide direct information about the energy landscape. Observe that such analysis does not require that the native structure be known. This is an important advantage of ChainTweak: it can be used in homology modeling to pick the right energy function.

Using a ChainTweak-generated ensemble around an alpha-helix, we compare different energy functions and demonstrate, in a direct way, the value of incorporating solvation effects (Table in Fig 5).

4. Discussion

In this paper, we have presented a formulation of the neighbourhood sampling problem that is independent of any search problem or energy function. Our proposed method for this problem, ChainTweak, can be used as a tool in many different applications and also enables novel applications like analysis of the energy-landscape around a particular conformation.

ChainTweak provides significant performance improvements over existing methods. Unlike discrete off-lattice phi-psi angle models[2,3], it does not generate (and reject) infeasible solutions. Its perturbation size (and, thus, efficiency) is much larger than what is possible with MD-like methods[1,5] and, unlike these methods, it can also modify partial structures. With database-based methods, e.g., fragment-swap MC[4], a small database size restricts the number of solutions that can be found while a larger database reduces efficiency. ChainTweak, in contrast, is fast and explores all possible local perturbations at each step.

Like ChainTweak, some MC-based methods[6,7] also make local moves by compensatory modification of dihedral angles. While ChainTweak's modular design allows easy emulation of these methods' local-modification approaches, its currently chosen methods for loop-closure[16,18] allow larger local perturbations (i.e., more efficient space coverage) and the ability to get multiple possible alternative local moves at each step, at no extra cost. Thus, unlike existing methods, per-residue phi-psi preferences can be easily supported.

A goal of this paper has been to demonstrate the usefulness of a stand-alone neighbourhood sampling program. In future work, we hope to further explore the use of ChainTweak in problems where it might enable new analyses and methods. For example, ChainTweak-generated ensembles could be used to further analyze energy functions and add entropic terms to them. Using ChainTweak, conformational propensities of disordered regions

in proteins[25] and conformational variation across sets of re-engineered[28,29] or homologous[30] proteins could be studied. It could be used in conjunction with existing methods[26,27] to analyze the ligand-protein docking process. We are also considering extending the algorithm to handle sidechain rotamer preferences and covalently-modified resiudues (e.g., phosphorylation).

Acknowledgments: The authors thank Phil Bradley, Amy Keating and Michael Levitt for their suggestions; Jean-Claude Latombe for pointing out reference 18; and Nathan Palmer and Allen Bryan for their comments.

References

1. Huang ES *et al. Using a hydrophobic contact...* J Mol Biol, 257(3):716-25, 1996
2. Park B *et al. Energy functions that discriminate...* J Mol Biol, 258(2):367-92, 1996
3. Kolodny R *et al. Small libraries of protein ...* J Mol Biol, 323(2):297-307, 2002
4. Chivian D *et al. Automated prediction....* Proteins, 53:524-533, 2003
5. Guntert P *et al. Torsion angle dynamics...* J Mol Bio, 273:283-298, 1997
6. Ulmschneider JP, Jorgensen WL. *Polypeptide folding...* J Am Chem Soc 18;126(6):1849-57, 2004
7. Cahill M, Cahill S, Cahill K. *Proteins wriggle* Biophys J, 82(5):2665-70, 2002
8. Krishnamoorthy B, Tropsha A. *Development of a...* Bioinformatics 19:1540-8, 2003
9. Dudek MJ, Ponder JW. *Accurate Modeling of...* J Comp Chem, 16:791-816, 1995
10. DePristo MA *et. al. Ab initio construction...* Proteins, 1:51(1):41-55, 2003
11. Berman HM *et al. The Protein Data Bank.* Nucl Acids Res, 28:235-242, 2000
12. Samudrala R, Levitt M. *Decoys 'R Us ...* Protein Sci, 9(7):1399-401, 2000
13. Branden C, Tooze J. *Introduction to Protein Structure.* Garland Pub, NY, 1991
14. Diamond R. *Personal communication with Michael Levitt*
15. Go N, Scheraga H. *Ring closure....* Macromolecules, 3(2):178-187, 1970
16. Manocha D *et al. Conformational analysis....* Comp App of Bio Sci, 11(1):71-86, 1995
17. Wedemeyer WJ, Scheraga HA. *Exact Analytical...* J Comp Chem, 20:819-844, 1999
18. Coutsias EA *et. al. A kinematic view....* J Comp Chem 25(4):510-28, 2004
19. Aloupsis G *et al. Flat-state connectivity of linkages under dihedral motion.* In Proc 13^{th} Intl Sym on Alg and Comp, 369-380, 2002
20. Biedl T *et al. Locked and unlocked polygonal chains in 3D.* In Proc 10^{th} ACM-SIAM Symposium on Discrete Algorithms, 866-867, 1999
21. Keasar C, Levitt M. *A novel approach...* J Mol Biol, 23;329(1):159-74, 2003
22. Basch J *et al. Data structures for mobile data.* In Proc of 8^{th} ACM-SIAM Symp Discrete Algo, 747-756, 1997
23. Lotan I *et al. Efficient Maintenace....* In Proc Symp Comp Geom, 2002
24. Qiu D *et al. The GB/SA Continuum...* J Phys Chem A, 101:3005-3014, 1997
25. Dunker K *et al. Intrinsically disordered protein.* J Mol Grpa Mod, 19:26-59, 2001
26. Edelsbrunner H *et al. Anatomy of protein...* Prot Sci, 7:1884-1897, 1998
27. Apaydin MS *et al. Studying Protein-Ligand...* Bioinformatics, 18(2):18-26, 2002
28. Babbitt PC, Gerlt JA. *New Functions from...* Adv in Prot Chem, 55:1-28, 2000
29. Mooney SD *et al. Conformational Preferences of...* Biopolymers, 64(2):63-71, 2002
30. Gerstein M, Altman RB. *Using a measure...* Comp App Bio, 11(6):633-44, 1995