PROTEINS
STRUCTURE ■ FUNCTION ■ BIOINFORMATICS

Fast *de novo* discovery of low-energy protein loop conformations

Samuel W. K. Wong,[1]* Jun S. Liu,[2] and S. C. Kou [2]*

[1] Department of Statistics, University of Florida, Gainesville, Florida 32611

[2] Department of Statistics, Harvard University, Cambridge, Massachusetts 02138

ABSTRACT

In the prediction of protein structure from amino acid sequence, loops are challenging regions for computational methods. Since loops are often located on the protein surface, they can have significant roles in determining protein functions and binding properties. Loop prediction without the aid of a structural template requires extensive conformational sampling and energy minimization, which are computationally difficult. In this article we present a new *de novo* loop sampling method, the Parallely filtered Energy Targeted All-atom Loop Sampler (PETALS) to rapidly locate low energy conformations. PETALS explores both backbone and side-chain positions of the loop region simultaneously according to the energy function selected by the user, and constructs a nonredundant ensemble of low energy loop conformations using filtering criteria. The method is illustrated with the DFIRE potential and DiSGro energy function for loops, and shown to be highly effective at discovering conformations with near-native (or better) energy. Using the same energy function as the DiSGro algorithm, PETALS samples conformations with both lower RMSDs and lower energies. PETALS is also useful for assessing the accuracy of different energy functions. PETALS runs rapidly, requiring an average time cost of 10 minutes for a length 12 loop on a single 3.2 GHz processor core, comparable to the fastest existing *de novo* methods for generating an ensemble of conformations.

Proteins 2017; 85:1402–1412.

Key words: protein structure prediction; loop sampling methods; particle filtering.

INTRODUCTION

Development of computational methods for protein structure prediction from amino acid sequence has received widespread attention since the 1970's.[1] Significant progress has been made in homology modeling, which uses experimentally determined structures as templates for building the prediction [for example Refs. 2, 3]. Such template-based methods have been quite successful at identifying the overall fold of a protein and its secondary structure elements. Loops are the regions that connect regular secondary structure elements. As they often occur on the protein surface, loops have an important role in protein function. They are more challenging to model correctly as loops often have low sequence identity with structural templates. The accuracy of homology models thus tend to be lowest in the loop regions, and methods to improve the prediction of loops without the aid of overall structural templates are necessary. Some recent methods thus make use of local templates or fragment databases built specifically from loop regions to aid prediction.[4,5] The effectiveness of these methods has improved as the size of the Protein Data Bank (PDB) has increased over time. However, as loop regions have highly variable 3D structures, a large portion of the potentially viable conformational space cannot be evaluated by template-based searches. For example, as noted by the authors of the LoopIng template-based loop prediction method,[5] *de novo* methods have the potential to perform better than template-based methods when the conformational space can be adequately explored.

Additional Supporting Information may be found in the online version of this article.

Grant sponsor: NSF grant; Grant number: DMS-1510446; Grant sponsor: NIH grant; Grant number: R01 GM113242-01.

*Correspondence to: S. W. K. Wong; Department of Statistics, University of Florida, Gainesville, FL 32611. E-mail: swkwong@stat.ufl.edu or S. C. Kou; Department of Statistics, Harvard University, Cambridge, MA 02138. E-mail: kou@stat.harvard.edu

Received 29 January 2017; Revised 19 March 2017; Accepted 27 March 2017
Published online 5 April 2017 in Wiley Online Library (wileyonlinelibrary.com).
DOI: 10.1002/prot.25300

This article focuses on *de novo* loop prediction. *De novo* prediction does not assume any structural template, and instead focuses on a more comprehensive exploration of the conformational space of the loop region with the guidance of an energy function to identify low-energy candidates [for example Refs. 6–11]. The success of this approach requires two main ingredients: a sampling algorithm to generate loop candidates, and an energy (or scoring) function to rank the loop conformations. These two are inextricably linked: low energy conformations must be found among the samples, for the sampling to be considered successful; it is also necessary for the energy function to be accurate, such that loop conformations scoring favorably according to that specific energy function correspond to accurate structure predictions. While this *de novo* approach is conceptually promising, it is generally acknowledged that sampling is a critical bottleneck.[6,11,12] The method proposed in this article addresses this need for a more effective loop sampling procedure.

A good energy function should successfully discriminate native-like loop conformations from decoys. Various energy functions have been proposed and used for loop modeling, with some having been specifically designed for that purpose.[13–15] However, because of the extremely large size of the space, it is challenging to locate low-energy conformations especially for longer loops ($\geq$ 12 residues long). It is therefore necessary to simultaneously achieve the twin goals of low-energy loop samples and computational efficiency, and our method offers progress toward these goals. Previous methods that are able to achieve highly accurate predictions have depended on a sequence of sampling and minimization steps on coarse and fine-grained energy functions that typically require many hours of CPU time to model one loop region, for example, 320 hours for the KIC method[16] and 4–7 hours for the LEAP method[17] for length 12 loops; the PLOP method[7] also includes further constraints of known crystal packing that will be unavailable for *de novo* loop prediction applications. Methods that complete in less computational time have suffered from insufficient sampling, for example, see Table IV in Ref. 12, in the sense that the energy of the loop in the native structure is typically lower than all sampled conformations, when tested on loop reconstruction in known structures. Our aim is to generate loop conformations with near-native energies according to any energy function supplied, while having a low computational cost.

The new method for loop sampling presented in this article is named the Parallely filtered Energy Targeted All-atom Loop Sampler (PETALS). It draws some inspiration from chain growth strategies. In the context of loops, a sampling algorithm must generate properly closed (connected) conformations from the starting to the ending residue of the loop region. Chain growth is a general technique to construct closed loop conformations: amino acids are sequentially added, and constraints can be incorporated to favor the eventual closure of the loop.[18–22] Among these, the recent method DiSGro[13] uses a distance and energy guided Monte Carlo sampling technique to sequentially grow the loop conformation one amino acid at a time, which performed better than many existing methods with a lower computational cost. That technique has also been successfully extended to the sampling of multiple interacting loops in the same protein.[23] In contrast, the SWA protocol[24] also builds the loop one residue at a time using a stepwise enumeration procedure to achieve high accuracy, although with a significant computational cost ($\sim$5000 CPU hours). Other studies have also shown that accounting for an energy function within the sampling algorithm enables more near-native conformations to be generated in loop reconstruction test sets.[9,12,25] As these studies indicate that energy and sampling have a tight connection, we also want to leverage efficiency gains from energy-guided sampling.

We briefly describe the salient features of our method that help increase sampling efficiency. First, the positions of both backbone and side-chain atoms are explored jointly during each step of sequential growth. A filtered list of low-energy side-chain positions is maintained to ensure that a low energy side-chain state is available for the completed loop conformation. Second, we construct an entire ensemble of loop conformations together for a loop region of interest rather than building them one at a time, and the partially grown loop conformations are probabilistically filtered after adding each amino acid to retain the most promising candidates. This strategy is designed to target high coverage of the low-energy regions of the conformational space, while at the same time avoiding the sampling of substantially identical conformations. At two residues remaining, we leverage the CSJD analytical closure method[26] to complete the backbone.

We note that PETALS can be paired with any energy function selected by the user, to efficiently generate low-energy loop conformations according to the chosen energy function. As an important application, we can use PETALS to assess how well different energy functions perform in the context of loop modeling in an independent and fair manner.

The efficacy of these innovations is demonstrated on benchmark loop datasets. We show that our method can rapidly discover conformations with low, near-native energies when tested with the commonly used DFIRE potential[27] and the DiSGro loop energy function,[13] both of which are publicly available and have been used successfully previously for loop modeling. The typical time cost on a single 3.2 GHz Xeon CPU core for a length 12 loop is 10 min. Thus, PETALS helps alleviate the sampling bottleneck and will be useful for loop

modeling problems when paired with a suitable energy function. The low-energy loop conformations sampled by PETALS would be effective for downstream applications that subsequently perform energy minimization on the sampled structures.

We adopt usual loop modeling conventions throughout: we assume that positions of the backbone atoms from the C of the starting residue to the C_α of the ending residue are unknown, and positions of all side-chain atoms from the starting to ending residues are unknown. The global RMSD of the backbone N, C_α, C, and O atoms of the residues composing the loop region is calculated when comparing sampled loop conformations to the native conformation on loop reconstruction datasets.

MATERIALS AND METHODS

PETALS samples an entire ensemble of conformations together for a given loop region of interest, by alternately building one amino acid to each of its ends (N and C terminus). For example, if residues 85–96 of a protein compose the loop region, the order of construction in PETALS is $85, 96, 86, 95, \ldots$. Specifically, building position i from the N-terminus end places the coordinates of atoms: C, O, and side-chain of residue i; N and C_α of residue $i+1$. Building position j from the C-terminus end places the coordinates of atoms: C_α, N, and side-chain of residue j; C and O of residue $j-1$. We let the backbone dihedral angles (ϕ, ψ, ω) and the side-chain dihedral angles be the geometric degrees of freedom that determine these coordinates, with all bond lengths and angles fixed at standard values since they do not exhibit much variation in high-resolution crystal structures.

As loop construction proceeds sequentially, we use the term "seeds" to represent the current ensemble of partial loop conformations. Growing the next residue involves two steps. First, each seed goes through an exploration step, which determines plausible choices of backbone dihedral angles for the next residue. Then, all "seeds" (that is, partial loop conformations) together go through a filtering step that evaluates their energies, selects the most promising seeds to continue growing, and eliminates ones that are substantially identical. The algorithm is initialized with a set of 100 empty seeds. The two steps are detailed below, and an illustration of the two steps for the construction of one residue is shown in Figure 1.

Exploration step

For each seed, this step explores the environment of the next residue to be built. The dihedral angles (ϕ, ψ) are discretized into 5° by 5° bins for this purpose, while ω is sampled from a Gaussian distribution with mean 180° and SD 2.75°. The exception is when the neighboring residue is Proline, when the mean of the Gaussian used for ω is 180° with probability 0.9 and 0° with probability 0.1. A (ϕ, ψ) bin is deemed to be feasible if the following five criteria are satisfied.

1. *Feasible on Ramachandran plot*: For loops, the relevant dihedral angles for (ϕ, ψ) are based on the Ramachandran density plot,[28] for each of the 20 residue types with secondary structure type "coil" according to DSSP,[29] since loop regions are of primary interest. To create a probability mass function over the 5° by 5° bins, we tabulated the (ϕ, ψ) frequencies in each bin over "coil" regions in proteins on the CulledPDB list by PISCES[30] on March 14, 2015 with these settings: no $>20\%$ sequence similarity, resolution 2.0 Å, *R*-factor cutoff 0.25. We excluded PDBs that also appear in the test sets. Dividing by the total count, empirical probabilities for each bin are obtained. A bin is considered feasible if its empirical probability is > 0.00002. Proline has 602 such bins; glycine has 2450. All other residue types have $\sim$1600 bins.
2. *Distance feasibility*: We ensure that distances to the current endpoint of the loop are feasible, for the number of residues remaining to be constructed. This check increases the probability that we can eventually form a properly closed loop conformation. Consider building residue i from the N-terminus end (that is, the C, O, and side-chain of residue i; N and C_α of residue $i+1$), when there are l remaining residues to the C-terminus end of the partially constructed loop conformation. Let d_0 be the distance from the i-th C_α to the $(l+i)$-th C_α. For a given (ϕ, ψ) bin along with the sampled value of ω, the backbone coordinates of the i-th C atom, and the $(i+1)$-th N and C_α atoms are determined. Let d_1 be the distance from the i-th C to the $(l+i)$-th C_α, and d_2 be the distance from the $(i+1)$-th C_α to the $(l+i)$-th C_α. Then, each (ϕ, ψ) bin maps to a distance pair (d_1, d_2). With these definitions, distances should be sensible in two aspects: the overall remaining distance (d_1, d_2), and the distance increment toward the ending C_α: $(d_1 - d_0, d_2 - d_0)$ given d_0. Using the same database of proteins, we found the empirical 0.01% and 99.99% quantiles for d_1 and d_2 (for each l), and $d_1 - d_0$ and $d_2 - d_0$ (for each combination of l and d_0, with d_0 rounded to the nearest Angstrom). We then divided the range of these respective 0.01% and 99.99% quantiles into 16 equally spaced intervals, and tabulated the bivariate frequencies of (d_1, d_2) and $(d_1 - d_0, d_2 - d_0)$ in the database in each. Thus, we say a (ϕ, ψ) bin is feasible if its distance pairs (d_1, d_2) and $(d_1 - d_0, d_2 - d_0)$ fall in corresponding table cells with nonzero frequencies. Checking feasible distances for residues built from the C-terminus end is analogous.
3. *Steric feasibility of backbone atoms*: The backbone of the residue built consists of the atoms N, C_α, C, O, and C_β (except Glycine). A (ϕ, ψ) bin is deemed infeasible if one of these atoms has a steric clash with

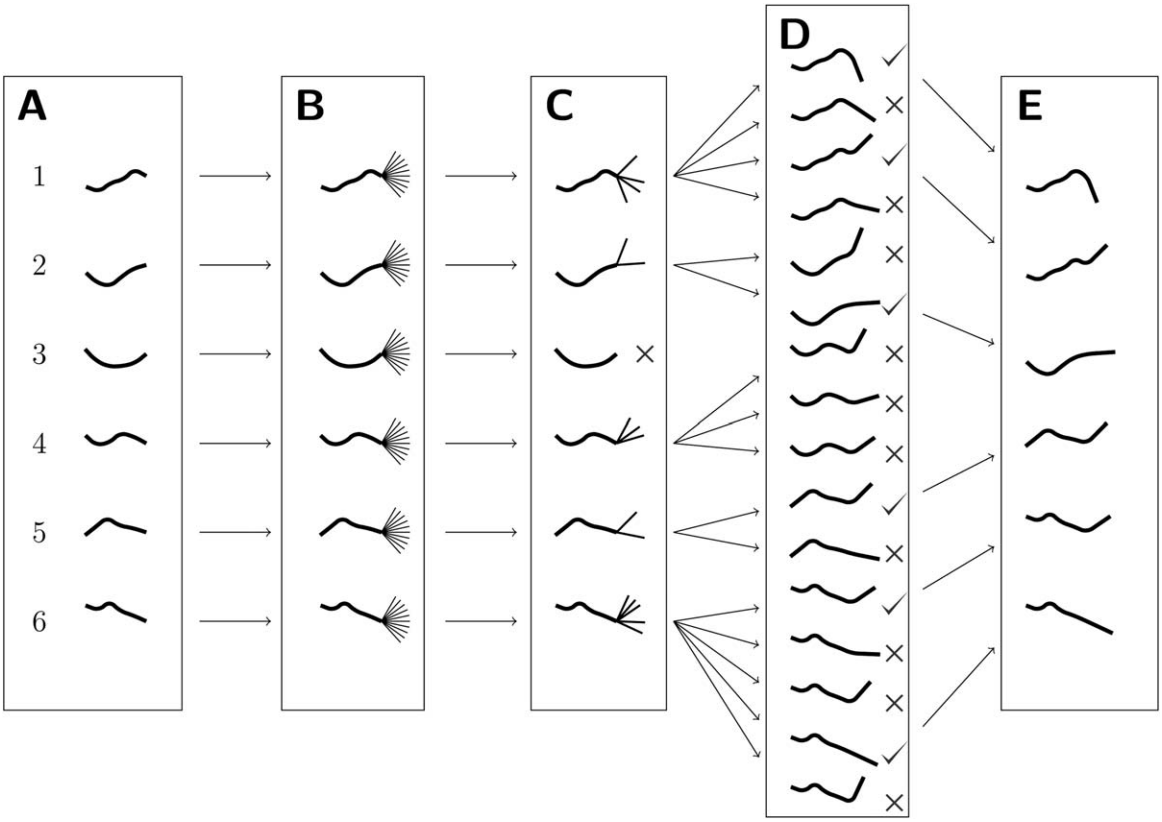

Figure 1

Illustration of the exploration and filtering steps to construct the next residue. (**A**) *Starting seeds*. These are the *N* partial loop conformations (six shown to illustrate) for the residues constructed so far. (**B**) *Beginning of exploration step*. For each seed in panel A, all (ϕ, ψ) bins for the placement of the next residue are evaluated for feasibility. (**C**) *Result of exploration step*. Feasible bins (up to 100 per seed) are identified and chosen; for example, seed 1 has four feasible bins, while seed 3 has no feasible bins and is a dead end, as indicated by the "X". (**D**) *Beginning of filtering step*. Each feasible bin of each seed gives one partial loop conformation. Together, these make up the pool of conformations to be filtered; for example, seed 1 contributes four conformations, while seed 5 contributes two. The number of partial loop conformations in this pool will be much larger than *N*; the filtering step selects the *N* most promising conformations, as indicated by the checkmarks. The remaining conformations are discarded, as indicated by the "X"s. (**E**) *Result of filtering step*. The *N* partial loop conformations selected by filtering become the starting seeds for the construction of the following residue; for example, the original seeds 1 and 6 each contribute two conformations after filtering, seeds 2 and 5 each contribute one, while seed 4 contributes none.

a backbone atom in the currently constructed loop conformation, or any backbone or side-chain atom from outside the loop region. A steric clash is defined to be a Lennard–Jones interaction between two atoms that exceeds 10.0 kcal/mol, detected using a distance cutoff. For feasible bins, we evaluate the total energy of these backbone atoms (N, C_α, C, O, and C_β) according to the provided energy function; this energy value will be used later in the filtering step. Atoms that are >12 Angstroms away from the current C_α are excluded from the energy evaluation. Further, interacting atoms between 8 and 12 Angstroms away are evaluated on a coarser 10° grid and interpolated. These approximations have little impact on the accuracy of the computed energy, but yield large computational savings.

4. *Steric feasibility of side-chain atoms*: For computational efficiency, possible side-chain positions are represented by rotamers. Energies of all rotameric positions in the library are evaluated, using the rotamer definitions in Ref. 31. For the bin to be feasible, at least one rotamer must be free of steric clashes (that is, no Lennard-Jones interaction $\geq$10.0 kcal/mol) with the rest of the protein and the backbone of the loop region. To reduce the occurrences of steric clashes because of rigid rotamers, we allow the χ_1 dihedral angle to be sampled from a Gaussian distribution centered at the rotamer definition with SD 10°.
5. *Possible placements of next residue*: Lastly, the dihedral space for the next residue conditional on the current (ϕ, ψ) is scanned on a coarse 30° grid to check that there is least one potential backbone placement free of steric clashes within a 6 Angstrom radius. Using this strategy of looking one residue ahead, upcoming dead-ends are better foreseen and eliminated. When there

are exactly two residues remaining to be built in the loop, this check is replaced by applying the CSJD analytical closure method.[26] This ensures that the loop conformation can close correctly; in this case the energy of the closure backbone atoms are then evaluated to check for clashes.

Among feasible (ϕ, ψ) bins, a maximum of 100 candidates are randomly selected. Each seed of length l_0 residues thus generates up to 100 partially built loop conformations of length l_0+1 for the filtering step. Some seeds may turn out to have no feasible bins at all; these are dead ends and are discarded.

Filtering step

The total number of partial loop conformations in the ensemble is greatly expanded by the exploration step, since each seed generates a maximum of 100 candidates; for example, beginning with 100 empty seeds the ensemble size can grow up to 10,000, 1 million, and 100 million after one, two, and three exploration steps respectively, if no conformations are discarded. Let N denote the maximum number of seeds to be used during the computation (N is set to 10,000 in our examples). The goal of filtering is to select a total of N candidates from the ensemble to use as seeds for the next exploration step. Filtering is critical to enable further exploration to target the promising low-energy regions of the conformational space, and to eliminate essentially duplicated loop conformations in the ensemble. We consider conformations as *essential duplicates* if the RMSDs between them are very small, that is, < 0.25 when the loop to be modeled is of length ≥ 6, and < 0.05 when the loop to be modeled is of length 4 to 5.

For each partial loop conformation, side chains can be rotated while keeping the backbone fixed so it is premature to finalize side-chain positions before the backbone is complete. At the same time, we wish to leverage the information gained by having evaluated the energy impact of side-chain rotamers with the rest of the protein during exploration. Our solution is to maintain a small set of viable rotamer combinations for each seed throughout growth. Specifically, for each partial loop conformation the interactions of a maximum of n_{rot} rotamers for the current residue and n_{sc} total rotamer combinations for the l_0 previously built residues are considered. Following energy evaluation, the lowest n_{sc} combinations out of the total $n_{rot} \times n_{sc}$ are retained. If n_{rot} and n_{sc} are too small, a point may be reached where there are no longer any rotamer combinations free of steric clashes, especially for long loops. Larger values of n_{rot} and n_{sc} increase the likelihood that there are low energy side-chain conformations throughout loop growth, at the cost of more computation time. For loops tested up to length 13, we found that setting $n_{rot} = 20$ and $n_{sc} = 25$ are sufficient; increasing these values further does not improve the minimum energy of the final sampled loop conformations. These interactions define the total energy for each partial loop conformation: its backbone energy plus the minimum energy of its retained side-chain rotamer combinations. For details see Algorithm 1 in the SI.

If the total number of partial loop conformations is less than N, then we proceed directly to the exploration step of the next residue. Otherwise, the list of partial loop conformations must be filtered to select N seeds, according to the sorted list of their total energies. First, partial loop conformations originating from the same seed are subject to a RMSD cutoff criterion: only the lowest energy loop conformations that are not *essential duplicates* are kept. Second, if the number of partial loop conformations is still far in excess of N (we use a cutoff of 10 N), the number of representatives kept from each seed is further reduced to a maximum of the n_{rep} lowest energies. This avoids the problem of having one seed being over-represented as sampling proceeds. Third, a selection of N seeds is made with the composition of two groups: (1) the pN loop conformations with the lowest energies, (2) $(1-p)N$ loop conformations uniformly selected at random from all remaining ones that are free of steric clashes, where $0 \leq p \leq 1$. Intuitively, low energy partial loop conformations should be retained as they appear the most promising to form eventual low energy complete loop conformations. However, the energy evaluated on only a partial loop conformation cannot be completely indicative of future success. Therefore, the pool of seeds is enriched with selections from the remaining partial loop conformations, which may themselves become more promising as growth continues. This strategy also ensures that a larger portion of the conformational space is explored. The effect of the choice of n_{rep} and the fraction p was tested on a randomly chosen set of 1000 loops of lengths 8, 10, and 12 from our list based on CulledPDB (see step 1 of Exploration section); based on those tests we suggest setting $n_{rep} = 20$ and $p = 0.90$. Fourth, if there are partial loop conformations that are *essential duplicates* in this set, only the lowest energy representative is kept (this pairwise RMSD screening is fast: <2 s for lengths up to 13 when $N = 10,000$). Any discarded loop conformations are replaced by a random selection from the remaining pool that are free of steric clashes.

Output of final loop conformations

After the last filtering step, we will have an ensemble of loop conformations and corresponding energy values ranked from smallest to largest. Two final tasks are applied to prepare a loop conformation for output in standard PDB format. First, the side-chain positions (except for the three closure residues) are set to the

Table I
Energy Sampling Comparison Between the DiSGro Algorithm and PETALS on the LoopBuilder Dataset for Different Loop Lengths

Len.	Targets	PETALS	DiSGro	Native	PETALS <DiSGro
8	63	−458	−398	−453	61 (out of 63)
9	56	−517	−432	−530	55 (out of 56)
10	40	−571	−438	−595	40 (out of 40)
11	54	−565	−422	−582	54 (out of 54)
12	40	−628	−445	−663	40 (out of 40)
13	40	−683	−443	−727	40 (out of 40)

Columns 3–6 represent respectively: average minimum DiSGro energy found by PETALS, average minimum DiSGro energy found by the DiSGro algorithm, average DiSGro energy of the native loop conformations, number of cases where the minimum DiSGro energy found by PETALS is lower than that found by the DiSGro algorithm.

minimum energy side-chain rotamer combination found during growth. The side chains of the three closure residues will be missing (as we used analytical closure for these last residues; see step 5 of Exploration section), so these are then added sequentially, choosing the lowest energy rotamer for each when evaluated against the whole protein. Second, to introduce flexibility into the rotamers to mimic real proteins and stabilize their energy, we pass through the loop residues one at a time and run 15 Levenberg-Marquardt iterations (http://www.ics.forth.gr/~lourakis/levmar/) to locally minimize the energy of the individual side-chain. We find that two such passes through the loop are sufficient; additional iterations yield negligible further improvements on the energy. If a loop conformation still has steric clashes in its side chains after this procedure, it is discarded.

PETALS allows the user to specify the number of loop conformations to output. We perform these two final tasks on the loop conformations in our ensemble, one at a time in order of energy values, until the specified number of loop conformations is reached.

Method availability

PETALS is freely available for Linux systems. It can be downloaded from either of these links: http://www.stat.ufl.edu/~swkwong/downloads/petals.tar.gz, or http://www.people.fas.harvard.edu/~skou/papers/petals.tar.gz.

RESULTS

We tested PETALS on a number of loop reconstruction datasets that have been used by other authors in previous studies: (a) the "Soto" set is the fifty-three 8-residue, seventeen 11-residue, and ten 12-residue loops considered for loop sampling in Ref. 12; (b) the "Canu" set is the ten length 8 and ten length 12 loops described in Ref. 32; (c) the "LoopBuilder" set is the length 8–13 loops in Ref. 12; (d) the "Fiser" set is the length 4–12 loops introduced originally in Ref. 33, with some low quality structures removed as detailed in Ref. 34. The number of final conformations we output is the same as other studies: 1000 for the Fiser set, and 5000 for the other sets.

PETALS can be paired with any energy function. For this study we have implemented four energy functions within PETALS: the first two are (1) DFIRE,[27] (2) DiSGro's loop-specific energy function.[13] Both have been previously used for loop modeling studies effectively and are publicly available. Since these are entirely atom distance-based functions that do not consider whether the dihedral angles (ϕ, ψ) of the backbone conformation are realistic, we created a simple backbone torsion (BBT) energy value for a loop residue using empirical log-probabilities of the (ϕ, ψ) bins (see step 1 of Exploration section). Incorporating these as an additive term, we constructed simple composite energy functions: (3) DFIRE + BBT, (4) DiSGro + BBT. Use of these functions will encourage the higher probability regions on the Ramachandran plot to be selected more frequently.

Low-energy sampling and filtering performance

To test the ability of PETALS to discover low energy loop conformations and the effectiveness of the proposed filtering criteria, we applied the method to the LoopBuilder set.

We compare PETALS to the DiSGro algorithm, which has a similar time cost and also employs a chain-growth strategy incorporating energy evaluation in the sampler to find low energy conformations. The DiSGro sampling algorithm uses the DiSGro energy function. Hence, we applied PETALS using the same DiSGro energy function for each loop target in the test to find the conformation with the minimum energy. The DiSGro algorithm has a separate step for selecting realistic backbone (ϕ, ψ) dihedral angles during sampling, so to mimic this behavior we also used the energy function DiSGro + BBT for sampling. Thus we applied PETALS twice, once with the DiSGro energy function, and a second time with the DiSGro + BBT composite energy function. To obtain the best possible energy results from the DiSGro algorithm, we ran their program with a much larger sample size than the default 5000, instead generating 100,000 closed loop conformations to be scored. The results of using PETALS with DiSGro + BBT are shown in Table I, which

Table II
Comparison of the RMSD of the Lowest DiSGro Energy Conformation Sampled by the DiSGro Algorithm and PETALS on the LoopBuilder Dataset

Length	8	9	10	11	12	13
PETALS	1.30	1.91	2.16	2.63	2.75	3.56
DiSGro	1.78	2.22	2.60	3.37	3.84	5.38

Averages for each loop length are shown. (The detailed list of these conformations is given in Supporting Information Table S2.).

Table III
Average RMSDs of the Partial Loop Conformations Selected and Discarded by Each Filtering Step, for Length 9 Loops

Filtering step	1	2	3	4	5	6
Avg RMSD selected	1.34	1.52	2.10	2.06	2.24	3.02
Avg RMSD discarded	1.38	1.62	2.38	2.36	2.63	3.13

compares the minimum DiSGro energies found by PETALS and the DiSGro algorithm averaged for each loop length as well as the average DiSGro energies of the native structures on the LoopBuilder set. We see that PETALS is able to discover lower energy conformations in the vast majority of cases than the DiSGro algorithm for all loop lengths. The results of running PETALS with the DiSGro energy function are shown in Supporting Information Table S1 in the SI, where we obtain even lower final DiSGro energy values. However, we believe the DiSGro + BBT composite energy is a better and more realistic energy function (see Assessment of energy functions Section below). As shown in Table I, the energy differences are the most pronounced at the longest loop lengths. In addition, the average energy gap between the native and the best PETALS sampled conformation is relatively small for all loop lengths, which demonstrates the ability of PETALS to discover conformations with near-native energies. The full data table that lists the sampling results on each target of the LoopBuilder set is provided in Supporting Information Table S2 in the SI, where PETALS uses the DiSGro + BBT composite energy for sampling.

To assess the geometric accuracy of these conformations (as listed in Supporting Information Table S2), we compared the RMSD of the lowest DiSGro energy conformation sampled by the DiSGro algorithm and PETALS for each loop target of the LoopBuilder set. The averages for each loop length are shown in Table II, which shows that PETALS' lowest energy conformations have lower RMSDs to the native structure compared to the DiSGro algorithm for each loop length. In Supporting Information Figure S1 in SI we show this visually, plotting for each loop length the (DiSGro energy, RMSD) pairs of the lowest DiSGro energy conformation sampled by the DiSGro algorithm and PETALS for each loop target of the LoopBuilder set; the plots show that PETALS' lowest energy conformations have both lower DiSGro energy and lower RMSDs on average. We want to emphasize that the RMSD accuracy of conformations with near-native energies is necessarily dependent on the accuracy of the energy function used. The fact that PETALS (and other methods) finds conformations with low energy but high RMSD shows that energy functions have their inaccuracies. However, this could provide useful data (decoys) for future training of improved energy functions.

The effectiveness of the PETALS filtering step can be assessed by examining the RMSDs of the partial loop conformations as each residue is grown. Table III shows this for length 9 loops, which have six exploration and filtering steps involving partially grown conformations. We see that the average RMSDs of the conformations selected as seeds for further growth at each filtering step are lower than those discarded, thus having a cumulative effect on the quality of sampled loops. We find a similar pattern of results for all other loop lengths.

As a result of filtering, we expect loop conformations generated by PETALS to have good coverage of the low RMSD regions in the conformational space. To assess this criterion, we group the sampled conformations according to RMSD, and count the number of substantively distinct conformations in each RMSD range. Here, we consider conformations to be substantively distinct if no pair is within 0.5 RMSD of each other. For both PETALS and the DiSGro algorithm, we generated 5000 conformations for each loop target in the LoopBuilder dataset. The results are summarized in Table IV, which shows the average number of substantively distinct conformations in each RMSD range. PETALS samples significantly more distinct conformations than the DiSGro algorithm in all the ranges below 3.0 RMSD from native.

Assessment of energy functions

We have shown that PETALS is effective at finding loop conformations with energies quite close to the native conformation. Thus as an important application, the method can be used to gain insight into the accuracy of different energy functions. For loop reconstruction datasets, an ensemble of low-energy loop conformations sampled according to an accurate energy function should contain conformations with low RMSDs to the native structure. An ideal energy function will be minimized

Table IV
Average Number of Distinct Conformations Sampled by PETALS and the DiSGro Algorithm in Each RMSD Range for the LoopBuilder Dataset, out of 5000 Sampled Conformations for Each Loop Target

	RMSD to native						
	0.0–0.5	0.5–1.0	1.0–1.5	1.5–2.0	2.0–2.5	2.5–3.0	3.0+
PETALS	0.3	23.7	158	363	483	471	1886
DiSGro	0.1	7.6	76	224	360	391	2801

Conformations are considered to be substantively distinct if no pair is within 0.5 RMSD of each other.

Table V
Minimum RMSD in the Ensemble of Loop Conformations Sampled by PETALS for the LoopBuilder Dataset

Length	Targets	(1)	(2)	(3)	(4)
8	63	0.78 [9]	0.70 [24]	0.65 [10]	0.62 [20]
9	56	1.02 [15]	0.82 [30]	0.82 [9]	0.73 [25]
10	40	1.32 [11]	0.99 [19]	1.02 [5]	0.83 [16]
11	54	1.92 [13]	1.36 [24]	1.46 [9]	1.08 [21]
12	40	2.28 [17]	1.50 [21]	1.55 [6]	1.21 [17]
13	40	3.22 [16]	2.19 [24]	2.64 [12]	1.64 [20]

Averages for each loop length are shown, using the different energy functions as guidance: (1) DFIRE, (2) DiSGro, (3) DFIRE + BBT, (4) DiSGro + BBT. The number of cases for each energy function where the native conformation has a lower energy than all sampled conformations is shown in square brackets.

Table VI
Comparison of Different Sampling Methods' Minimum RMSD from Native in an Ensemble of 5,000 Loop Conformations, Averaged over the Length 8, 11, and 12 Loops in Ref. 12

Method	8-res	11-res	12-res
Random Tweak	1.22	2.22	2.64
CCD	1.20	2.11	2.57
Wriggling	1.43	2.24	2.68
PLOP-build	0.99	2.18	2.69
Direct Tweak	0.69	1.20	1.48
LOOPYbb	0.89	1.51	1.80
DISGRO	0.80	1.19	1.28
PETALS	**0.62**	**1.12**	**1.20**

The best performing method in each length is boldfaced.

near the native structure, and thus few sampled conformations should have a below-native energy.

We illustrate this assessment on the LoopBuilder set by using PETALS with the four energy functions we have implemented. Table V shows the minimum RMSDs in the ensemble of loop conformations sampled by PETALS for each energy function, averaged by loop length. It is clear that the energy function used has an effect on loop conformation quality, and that backbone torsion angles (BBT) is a useful component. The DiSGro-based energy functions also have more cases where the native conformation has a lower energy than all sampled conformations, compared to DFIRE. These numbers are indicated within the square brackets in Table V. From these results, the DiSGro + BBT composite energy appears to be the most accurate among the four tested, for low energies to translate effectively to low RMSDs for loop reconstruction.

Loop sampling and prediction

For obtaining comparisons with other methods, we use PETALS with the DiSGro + BBT composite energy based on the results of the previous subsection.

First, we compare loop sampling methods according to the minimum RMSD found, where the different methods are used to generate an ensemble of 5000 loop conformations for each loop target. Table VI lists the minimum RMSDs on the loop targets in the Soto set averaged by loop length, and Table VII lists the minimum RMSDs for each loop target in the Canu set. We expect methods that incorporate steric interactions and scoring during sampling to perform the best according to this metric; Direct Tweak, DiSGro, and PETALS fall into this category. Among these three methods, PETALS has the lowest RMSD average for each loop length considered. In particular, PETALS compares favorably to the DiSGro algorithm which is also designed to sample low energy conformations from its energy function during loop construction.

Second, we compare loop prediction methods according to the RMSD of the lowest energy conformation found. Results for the Fiser set are shown in Table VIII, where four methods are compared. The RMSDs of the lowest energy conformation are shown in the "Pred." column. Overall, PETALS compares favorably to RAPPER,[34] FALCm, and DiSGro on loop prediction accuracy. In addition to prediction results, Table VIII also shows the minimum RMSDs and average RMSDs of the sampled ensemble for each method, in the "Min." and "Avg." columns respectively. These statistics are useful for summarizing the overall quality of conformations sampled by each method. PETALS has overall better results than the other methods on these summary statistics as well.

Table VII
Comparison of Different Sampling Methods' Minimum RMSD from Native for the Length 8 and 12 Loops in the Canu Set

Length	Loop	CCD	CJSD	SOS	FALC	FALCm	DiSGro	PETALS
8-res	1cruA_85	1.75	0.99	1.48	**0.60**	0.62	1.34	1.64
	1ctqA_144	1.34	0.96	1.37	0.62	**0.56**	0.70	0.60
	1d8wA_334	1.51	**0.37**	1.18	0.96	0.78	0.93	0.39
	1ds1A_20	1.58	1.30	0.93	0.80	0.73	**0.62**	1.30
	1gk8A_122	1.68	1.29	0.96	0.79	0.62	1.08	**0.60**
	1i0hA_145	1.35	0.36	1.37	0.88	0.74	0.80	**0.28**
	1ixh_106	1.61	2.36	1.21	0.59	0.57	0.39	**0.37**
	1lam_420	1.60	0.83	0.90	0.79	0.66	0.63	**0.46**
	1qopB_14	1.85	0.69	1.24	0.72	0.92	0.87	**0.46**
	3chbD_51	1.66	0.96	1.23	1.03	1.03	0.67	**0.51**
	Average	*1.59*	*1.01*	*1.19*	*0.78*	*0.72*	*0.80*	***0.66***
12-res	1cruA_358	2.54	2.00	2.39	2.27	2.07	1.84	**1.46**
	1ctqA_26	2.49	1.86	2.54	1.72	1.66	1.36	**0.86**
	1d4oA_88	2.33	1.60	2.44	0.84	**0.82**	1.50	**0.82**
	1d8wA_46	4.83	2.94	2.17	2.11	2.09	1.17	**1.01**
	1ds1A_282	3.04	3.10	2.33	2.16	2.10	1.82	**0.69**
	1dysA_291	2.48	3.04	2.08	1.83	1.67	1.45	**0.60**
	1eguA_508	2.14	2.82	2.36	1.68	1.71	2.13	**1.31**
	1f74A_11	2.72	1.53	2.23	1.33	1.44	1.46	**0.82**
	1qlwA_31	3.38	2.32	1.73	2.11	2.20	0.79	**0.65**
	1qopA_178	4.57	2.18	2.21	2.37	2.36	1.77	**1.32**
	Average	*3.05*	*2.34*	*2.25*	*1.84*	*1.81*	*1.53*	***0.96***

All methods sample 5,000 loop conformations for each loop target. Results for the first five columns are taken from Table II of Ref. 35. For PETALS, the energy function used is the composite E = DiSGro + BBT. The best performing method for each loop target is boldfaced. The averages for each length are also shown in italics.

Table VIII
Comparison of the Loop Conformations Sampled by RAPPER, FALCm4, DISGRO and PETALS on the Fiser Set, Where Each Method is Used to Generate an Ensemble of 1,000 Loop Conformations

		RAPPER			FALCm			DISGRO			PETALS		
Len.	#Targets	Min.	Avg.	Pred.	Min.	Avg.	Pred.	Min.	Avg.	Pred.	Min.	Avg.	Pred.
4	35	0.43	1.65	0.86	0.33	0.92	0.54	**0.21**	0.66	0.48	0.23	**0.61**	**0.34**
5	35	0.53	2.27	1.00	0.44	1.63	0.92	**0.25**	1.11	0.84	0.38	**1.00**	**0.66**
6	26	0.69	3.06	1.85	0.47	2.34	1.36	0.44	**1.74**	1.22	**0.38**	1.96	**0.90**
7	38	0.78	3.79	1.51	0.58	2.74	1.17	0.55	2.23	1.08	**0.48**	**2.02**	**0.98**
8	32	1.11	4.16	2.11	0.84	3.69	1.87	0.80	2.87	1.72	**0.67**	**2.29**	**1.34**
9	37	1.29	5.00	2.58	0.95	4.21	2.08	0.94	3.64	1.82	**0.82**	**2.75**	**1.79**
10	37	1.67	5.66	3.60	1.45	5.07	3.09	1.15	3.96	2.33	**0.95**	**2.91**	**2.27**
11	33	1.99	6.71	4.25	1.47	5.76	3.43	1.39	4.96	2.98	**1.12**	**3.74**	**2.55**
12	34	2.21	6.96	4.32	1.74	6.31	3.84	1.53	5.23	**2.99**	**1.33**	**3.73**	3.21

"Min.", "Avg.", and "Pred." denote the minimum RMSD, the average RMSD of the ensemble, and the RMSD of the conformation selected as the prediction, respectively, averaged over the loop targets of that length. The best performing method in each category is shown in boldface. The RAPPER, FALCm4, and DISGRO results are reported in Table III of Ref. 13.

DISCUSSION AND CONCLUSIONS

We applied our novel loop ensemble sampler, PETALS, on loop reconstruction datasets and achieved good results on a variety of criteria. Its key contribution is the ability to discover near-native, low-energy loop conformations with a small computational budget. This is achieved by sequentially building the ensemble of loop conformations in parallel with both backbone and side-chain atoms, and probabilistically filtering to retain the most promising conformations.

The speed of PETALS compares well with other loop prediction methods. By default, PETALS is multithreaded through the C++ openMP implementation, making use of all available cores on the system for the computation. For benchmarking purposes, we ran PETALS with a single Xeon 3.2 GHz core only, on the 35 length 12 loops in the Fiser set; with default settings the average computation time is 9.3 min (SD 2.6). Therefore PETALS achieves speeds comparable to those reported in the DiSGro article of 10 min for modeling length 12 loops, but is much more effective at locating low-energy conformations. PETALS is significantly faster than FALCm,[35] which requires 3 hr for modeling a 12-residue loop. Decoy sets generated by PETALS contain conformations that have low RMSDs to the native structure, low energies, and side chains included. Many other methods first generate only the backbones of loop conformations, requiring separate steps for building side chains and scoring. The calculations in Table III of Ref. 12 indicate that the methods Random Tweak,[36] CCD,[32] Wriggling,[37] PLOP-build,[19] Direct Tweak ,[12] and LOOPYbb[9] would require 5, 38, 11, 36, 38, and 30 min, respectively, to generate 5000 backbone conformations free of steric clashes for a length 12 loop. CJSD[26] and SOS[25] are very fast at generating the backbones of loop conformations, requiring 3.6 and 95 s respectively to generate 5000 backbone conformations for a length 12 loop; however the authors do not report the additional computing time that would be needed for adding side chains and performing energy minimization.

PETALS is highly extensible. The sampler can be paired with any energy function that can be incrementally evaluated as the loop is built. The DFIRE and DiSGro energy functions were chosen for use in this study as they were publicly available and previously used for loop modeling, but may limit performance because of the inaccuracies in these energy functions. From our results we expect PETALS to have further RMSD improvements on loop reconstruction tests, if paired with more accurate energy functions than those tested in this study. Further work can incorporate such additional energy functions into PETALS. We also note here that the choice of energy function to pair with a given sampling method must be made judiciously to obtain good results. For example, if a sampling method uses rigid rotamer representations, then an energy function that strongly penalizes side-chain atomic clashes would be unlikely to perform well, as the rigid rotamers limit the conformational space that can be sampled; thus, the lowest energy conformations may not be found by the sampling method, and ranking sampled conformations by energy may not be informative. Likewise, if an energy function includes terms involving hydrogen atoms explicitly, then a sampling method using that energy function also needs to sample hydrogen atom positions to properly explore the low-energy conformational space. In the current setting, DiSGro and PETALS have very similar protein structure representations, and hence a direct comparison of sampling efficacy according to the DiSGro energy function can be made. If a different energy function is chosen for use with PETALS, appropriate modifications can be made to our method to sample flexible bond lengths, angles, and hydrogen placements, and so forth as necessitated by that energy function.

Adaptations can be easily made for loop modeling applications in inexact environments. For example, if some parts in the protein outside the loop region are uncertain, in particular side chains, those atoms can be excluded from energy evaluation during loop construction. Even in these cases, the simultaneous construction of side chains on the loop is still recommended, as their interactions with the rest of the backbone will continue to provide useful guidance.

The ability of PETALS to rapidly discover low-energy loop conformations is also useful for further energy function development. For example, sampled loop conformations that have energies below that of the native loop conformation could be considered decoys. A common approach to training statistics-based energy functions is discriminating natives from a reference state,[38] and decoys that PETALS generates can be directly applied for that purpose.

PETALS provides loop conformations that can be used as starting points for further refinement, for example through force field minimization or molecular dynamics simulations. Since the energies of our sampled conformations are already of high quality, the efficiency of such downstream procedures should also be improved.

In computational drug design, 3 D structural information of a target protein is required in order to screen for its ability to dock with ligands; homology models are often used for this purpose when an experimental structure based on X-ray crystallography or nuclear magnetic resonance (NMR) spectroscopy is unavailable.[39,40] Loop modeling is an important step for improving the quality of homology models. Thus, we expect that PETALS can be easily incorporated as an intermediate step in protocols for such applications as well.

ACKNOWLEDGMENT

We thank Kevin Bartz and Jinfeng Zhang for helpful discussions, and Sarah Lotspeich for assisting with data preparation.

REFERENCES

1. Anfinsen C. Principles that govern the folding of protein chains. Science 1973;181:223–230.
2. Roy A, Kucukural A, Zhang Y. I-TASSER: A unified platform for automated protein structure and function prediction. Nat Protoc 2010;5:725–738.
3. Remmert M, Biegert A, Hauser A, Söding J. HHblits: lightning-fast iterative protein sequence searching by HMM-HMM alignment. Nat Methods 2012;9:173–175.
4. Choi Y, Deane CM. FREAD revisited: accurate loop structure prediction using a database search algorithm. Proteins 2010;78:1431–1440.
5. Messih MA, Lepore R, Tramontano A. Looping: a template-based tool for predicting the structure of protein loops. Bioinformatics 2015;31:3767–3772.
6. Rohl CA, Strauss CE, Chivian D, Baker D. Modeling structurally variable regions in homologous proteins with rosetta. Proteins 2004;55:656–677.
7. Zhu K, Pincus DL, Zhao S, Friesner RA. Long loop prediction using the protein local optimization program. Proteins 2006;65:438–452.
8. Spassov VZ, Flook PK, Yan L. LOOPER: a molecular mechanics-based algorithm for protein loop prediction. Protein Eng Des Sel 2008;21:91–100.
9. Xiang Z, Soto C, Honig B. Evaluating conformational free energies: The colony energy and its application to the problem of loop prediction. Proc Natl Acad Sci U S A 2002;99:7432–7437.
10. Ko J, Lee D, Park H, Coutsias EA, Lee J, Seok C. The FALC-Loop web server for protein loop modeling. Nucleic Acids Res 2011;39:W210–W214.
11. Liang S, Zhang C, Sarmiento J, Standley DM. Protein loop modeling with optimized backbone potential functions. J Chem Theory Comput 2012;8:1820–1827.
12. Soto CS, Fasnacht M, Zhu J, Forrest L, Honig B. Loop modeling: Sampling, filtering, and scoring. Proteins 2008;70:834–843.
13. Tang K, Zhang J, Liang J. Fast protein loop sampling and structure prediction using distance-guided sequential chain-growth Monte Carlo method. PLoS Comput Biol 2014;10:e1003539.
14. Liang S, Zhang C, Standley DM. Protein loop selection using orientation-dependent force fields derived by parameter optimization. Proteins 2011;79:2260–2267.
15. Li J, Abel R, Zhu K, Cao Y, Zhao S, Friesner RA. The VSGB 2.0 model: a next generation energy model for high resolution protein structure modeling. Proteins 2011;79:2794–2812.
16. Mandell DJ, Coutsias EA, Kortemme T. Sub-angstrom accuracy in protein loop reconstruction by robotics-inspired conformational sampling. Nat Methods 2009;6:551–552.
17. Liang S, Zhang C, Zhou Y. LEAP: Highly accurate prediction of protein loop conformations by integrating coarse-grained sampling and optimized energy scores with all-atom refinement of backbone and side chains. J Comput Chem 2014;35:335–341.
18. de Bakker PI, DePristo MA, Burke DF, Blundell TL. Ab initio construction of polypeptide fragments: Accuracy of loop decoy discrimination by an all-atom statistical potential and the AMBER force field with the Generalized Born solvation model. Proteins 2003;51:21–40.
19. Jacobson M, Pincus D, Rapp C, Day T, Honig B, Shaw D, Friesner R. A hierarchical approach to all-atom protein loop prediction. Proteins 2004;55:351–367.
20. Vendruscolo M. Modified configurational bias Monte Carlo method for simulation of polymer systems. J Chem Phys 1997;106:2970–2976.
21. Wick C, Siepmann J. Self-adapting fixed-end-point configurational-bias Monte Carlo method for the regrowth of interior segments of chain molecules with strong intramolecular interactions. Macromolecules 2000;33:7207–7218.
22. Zhang J, Kou SC, Liu JS. Biopolymer structure simulation and optimization via fragment regrowth Monte Carlo. J Chem Phys 2007;126:225101.
23. Tang K, Wong SW, Liu JS, Zhang J, Liang J. Conformational sampling and structure prediction of multiple interacting loops in soluble and β-barrel membrane proteins using multi-loop distance-guided chain-growth Monte Carlo method. Bioinformatics 2015;31:2646–2652.
24. Das R. Atomic-accuracy prediction of protein loop structures through an RNA-inspired Ansatz. PloS ONE 2013;8:e74830.
25. Liu P, Zhu F, Rassokhin DN, Agrafiotis DK. A self-organizing algorithm for modeling protein loops. PLoS Comput Biol 2009;5:e1000478.
26. Coutsias E, Seok C, Jacobson M, Dill K. A kinematic view of loop closure. J Comput Chem 2004;25:510–528.
27. Zhou H, Zhou Y. Distance-scaled, finite ideal-gas reference state improves structure-derived potentials of mean force for structure selection and stability prediction. Protein Sci 2002;11:2714–2726.

28. Ramachandran G, Ramakrishnan C, Saisekharan V. Stereochemistry of polypeptide chain configurations. J Mol Biol 1963;7:95–99.
29. Kabsch W, Sander C. Dictionary of protein secondary structure - pattern-recognition of hydrogen-bonded and geometrical features. Biopolymers 1983;22:2577–2637.
30. Wang G, Dunbrack RL. PISCES: a protein sequence culling server. Bioinformatics 2003;19:1589–1591.
31. Shapovalov MV, Dunbrack RL. A smoothed backbone-dependent rotamer library for proteins derived from adaptive kernel density estimates and regressions. Structure 2011;19:844–858.
32. Canutescu A, Dunbrack R. Cyclic coordinate descent: A robotics algorithm for protein loop closure. Protein Sci 2003;12:963–972.
33. Fiser A, Do RK, Sali A. Modeling of loops in protein structures. Protein Sci 2000;9:1753–1773.
34. DePristo MA, de Bakker PIW, Lovell SC, Blundell TL. Ab initio construction of polypeptide fragments: efficient generation of accurate, representative ensembles. Proteins 2003;51:41–55.
35. Lee J, Lee D, Park H, Coutsias EA, Seok C. Protein loop modeling by using fragment assembly and analytical loop closure. Proteins 2010;78:3428–3436.
36. Shenkin P, Yarmush D, Fine R, Wang H, Levinthal C. Predicting antibody hypervariable loop conformation. 1. Ensembles of random conformations for ring-like structures. Biopolymers 1987;26:2053–2085.
37. Cahill S, Cahill M, Cahill K. On the kinematics of protein folding. J Comput Chem 2003;24:1364–1370.
38. Chuang G-Y, Kozakov D, Brenke R, Comeau SR, Vajda S. DARS (decoys as the reference state) potentials for protein-protein docking. Biophys J 2008;95:4217–4227.
39. Combs SA, DeLuca SL, DeLuca SH, Lemmon GH, Nannemann DP, Nguyen ED, Willis JR, Sheehan JH, Meiler J. Small-molecule ligand docking into comparative models with rosetta. Nat Protoc 2013;8:1277–1298.
40. Sliwoski G, Kothiwale S, Meiler J, Lowe EW. Computational methods in drug discovery. Pharmacol Rev 2014;66:334–395.