Biopolymer structure simulation and optimization via fragment regrowth Monte Carlo

Jinfeng Zhang, S. C. Kou, and Jun S. Liu[a)]
Department of Statistics, Harvard University, Science Center, Cambridge, Massachusetts 02138

(Received 14 February 2007; accepted 9 April 2007; published online 11 June 2007)

An efficient exploration of the configuration space of a biopolymer is essential for its structure modeling and prediction. In this study, the authors propose a new Monte Carlo method, fragment regrowth via energy-guided sequential sampling (FRESS), which incorporates the idea of multigrid Monte Carlo into the framework of configurational-bias Monte Carlo and is suitable for chain polymer simulations. As a by-product, the authors also found a novel extension of the Metropolis Monte Carlo framework applicable to all Monte Carlo computations. They tested FRESS on hydrophobic-hydrophilic (*HP*) protein folding models in both two and three dimensions. For the benchmark sequences, FRESS not only found all the minimum energies obtained by previous studies with substantially less computation time but also found new lower energies for all the three-dimensional *HP* models with sequence length longer than 80 residues. © *2007 American Institute of Physics*. [DOI: 10.1063/1.2736681]

I. INTRODUCTION

Predicting a protein's tertiary structure from its primary amino acid sequence is a long standing problem in biology. Two major difficulties have been challenging researchers: the design of appropriate energy functions and the exploration of the vast space of all possible structures; the latter has been suggested as the current bottle neck.[1] Main strategies for exploring a complex configuration space include molecular dynamics simulations, Markov chain Monte Carlo (MCMC) methods,[2] and other heuristics-based approaches. In a typical MCMC simulation, a new conformation is proposed at each step (known as a "proposal" or a "proposed move"), and the proposed move is either accepted or rejected according to a probability rule pioneered by Metropolis and co-workers.[3] Hastings[4] later generalized the method so that a "biased" move is allowed to be proposed. Since trapping at local energy minima is a general difficulty facing all sampling methods, designing a move set that can quickly traverse the configuration space is crucial to the success of any Monte Carlo strategy. Most existing MCMC moves for biopolymer simulations tend to be too rigid or too local. For example, the popular pivot move[5] has a very low acceptance rate at compact states. Other moves such as corner moves, end moves, and crankshaft moves designed for lattice polymers[6] only emulate a subset of real protein motions and are of a very local nature.

We introduce a new MCMC method, fragment regrowth via energy-guided sequential Sampling (FRESS), for protein structure simulation. A key ingredient of FRESS is to regrow from the current conformation a randomly selected fragment of varying length in each iteration. This regrowth of the fragment is carried out by energy-guided importance sampling so that conformations with lower energies adjacent to the current conformation have higher probabilities to be sampled. An example of the fragment growth is given in Fig. 1(a). FRESS embodies strengths of both configurational-bias Monte Carlo[7] (CBMC) and multigrid Monte Carlo (MGMC).[8] First, by employing sequential importance sampling to account for its energy "environment" when regrowing the selected fragment (as with a typical CBMC algorithm), FRESS shares the capability of sequentially probing the local energy landscape with CBMC. Second, by regrow-

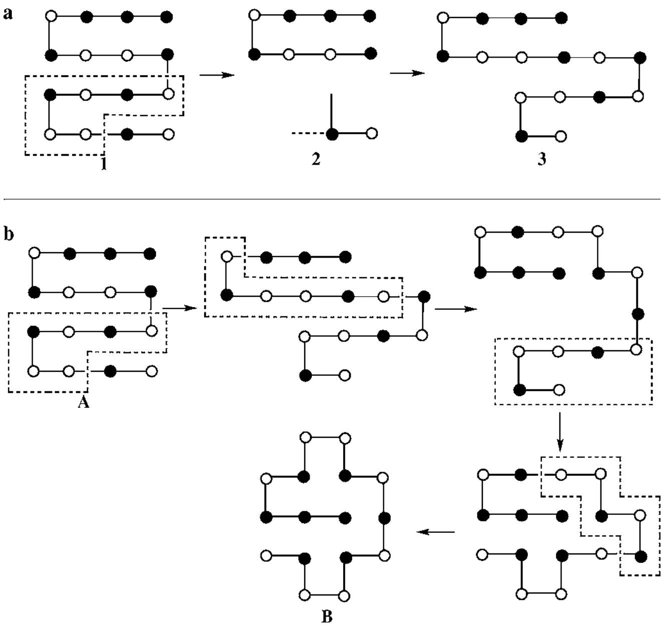

FIG. 1. An example of fragment growth on the 2D square lattice. (a) One step of FRESS move, with dashed border in (1) indicating the regrown segment. (b) Under FRESS the global minimum energy conformation B can be reached from a compact conformation A in just four steps when the maximum allowed fragment length is set to six. Residues enclosed in the dashed lines are the fragments to be regrown. The sequence and its ground state were taken from Ref. 32.

[a)] Author to whom correspondence should be addressed. Electronic mail: jliu@stat.harvard.edu

TABLE I. Comparison of performances of different methods on ten benchmark 3D *HP* sequences (Ref. 10). The setting for FRESS: starting temperature $T_h=3.5$, minimum temperature $T_l=0.1$, and temperature decreasing by 0.98 geometrically; fragment lengths between 2 and 12; and 50 000 moves at each temperature. ST: standard local moves with 2 000 000 moves at each temperature; other settings are the same as FRESS. Rows 2–11: the minimum energy (time spent on each run in minutes) found by the corresponding method. Row 12: the number of sequences for which the minimum energy conformations were found (the average time spent on the searches). The CPUs of the computers used to obtain the results are: FRESS and ST, 1.4 GHz PC; nPERMis, 167 MHz PC; nPERMh, 1.84 GHz PC; ACO, 2.4 GHz PC; CG, SPARC I workstation.

Seq.	FRESS	ST	CG[a]	nPERMis[b]	nPERMh[c]	ACO[d]
1	−32(0.72)	−32(2.03)	−32(9.4)	−32(0.13)	−32(1.22)	−32(30)
2	−34(0.88)	−32(4.02)	−34(35)	−34(0.23)	−34(1.45)	−34(420)
3	−34(0.77)	−32(4.02)	−34(62)	−34(0.71)	−34(0.37)	−34(120)
4	−33(0.53)	−31(3.95)	−33(29)	−33(6.57)	−33(1.83)	−33(300)
5	−32(0.72)	−31(3.97)	−32(12)	−32(2.55)	−32(1.78)	−32(15)
6	−32(0.68)	−31(3.93)	−32(460)	−32(1.44)	−32(0.58)	−32(720)
7	−32(1.12)	−30(3.97)	−32(64)	−32(3.35)	−32(0.50)	−32(720)
8	−31(0.80)	−30(3.73)	−31(38)	−31(0.46)	−31(2.01)	−31(120)
9	−34(0.73)	−33(3.88)	−33(26)	−34(10.53)	−34(32.7)	−34(450)
10	−33(0.73)	−33(1.90)	−33(1.1)	−33(0.08)	−33(0.34)	−33(60)
Summ.	10(0.77)	2(3.54)	9(73.6)	10(2.61)	10(4.28)	10(296)

[a] Reference 13.
[b] Reference 11.
[c] Reference 12.
[d] Reference 14.

ing fragments of *different* lengths, FRESS benefits from MGMC's insight of balancing local exploration and global moves.

II. RESULTS

A. Folding *HP* sequences

We applied FRESS to both two-dimensional (2D) and three-dimensional (3D) *HP* protein folding models, in which the amino acid residues are simplified to two types: a hydrophilic type (P type), which likes water, and a hydrophobic type (H type), which dislikes water, and the structure is described by a self-avoiding walk on a 2D square or 3D cubic lattice. Energies $e_{HH}=-1$ and $e_{HP}=e_{PP}=0$ are assigned to interactions between noncovalently bound neighbors on the lattice. The potential energy of a conformation is simply the sum of energy contributions from the (noncovalently) interacting lattice neighbors. This energy assignment leads to the desirable feature that hydrophobic residues tend to form a compact core surrounded by a hydrophilic shell.

The problem of finding the ground state of an *HP* sequence has been proven to be NP-complete[9] and, thus, heuristic methods such as those based on Monte Carlo are necessary. Conformation spaces of proteins not only have many local energy minima but are also very constrained around compact conformations. All these difficulties are present in the *HP* model. Searching minimum energy conformations of *HP* sequences therefore presents a very challenging test for protein structure prediction and optimization methods. Indeed, although several *HP* model sequences have been proposed and studied for many years, researchers can still find new conformations with lower energies from time to time.

We first considered ten benchmark 48-residue sequences designed for 3D cubic lattice (Table I).[10] We compared the optimization performance of FRESS with those of an MCMC algorithm using only standard local moves (including the end, corner, and crankshaft moves), nPERMis (Ref. 11) and nPERMh (Ref. 12) (two Rosenbluth type chain-growth-based methods), the core-guided (CG) search,[13] and ant colony optimization (ACO).[14] As shown in Table I, FRESS found the minimum energies for all ten sequences in less than 1 min on average. In contrast, the standard MCMC algorithm (with all other conditions identical to those of FRESS) found the minimum energies for only two of the ten sequences even with a fourfold increase in the average running time. CG was not able to find the minimum energy for sequence 9. ACO had to spend a substantially longer time than other algorithms to reach the minimum energies. Both nPERMh and nPERMis performed as consistently well as FRESS for these short sequences.

We also tested FRESS on ten 64-residue benchmark sequences reported in Unger and Moult[15] for 3D folding. For each sequence FRESS found minimum energy conformations that match the best known result to date,[16] but did not find any new lower energies. We suspect that for these relatively short sequences the globally lowest energies have already been reached. Representative conformations found by FRESS (one for each sequence) can be found online.[17]

An intuitive reason why FRESS may have a better capability in exploring conformation spaces of proteins is that the fragment regrowth moves with variable fragment length are capable of escaping local energy traps. As shown in Fig. 1(b), in just four FRESS moves one can reach the 2D global minimum energy conformation from a very different compact conformation for a 16-residue *HP* sequence. In contrast, it may take many steps of standard local MCMC moves to do so.

We next assess FRESS's ability in finding minimum en-

TABLE II. Benchmark *HP* sequences longer than 50 residues.

Seq. code	Length	Sequence
2D50	50	*HHPHPHPHPHHHHPHPPPHPPPHPPPPHPPPHPPPHPHHHHPHPHPHPHH*
2D60	60	*PPHHHPHHHHHHHHPPPHHHHHHHHHHPHPPPHHHHHHHHHHHHPPPPHHHHHHPHHPHP*
2D64	64	*HHHHHHHHHHHHPHPHPPHHPPHHPPHPPHHPPHHPPHPPHHPPHHPPHPHPHHHHHHHHHHHH*
2D85	85	*HHHHPPPPHHHHHHHHHHHHPPPPPPHHHHHHHHHHHHPPPHHHHHHHHHHHHHHPPPHHHHHHHHHHHHHPPPHPPHHPPHHPPHPH*
2D100a	100	*PPPPPPHPHHPPPPPHHHPHHHHHPHHPPPPHHPPHHPHHHHHPHHHHHHHHHHHHPHHPHHHHHHHHPPPPPPPPPPPHHHHHHHPPHPHHHPPPPPPHPHH*
2D100b	100	*PPPHHPPHHHHPPHHHPHHPHHPHHHHPPPPPPPPHHHHHHPPHHHHHHPPPPPPPPPHPHHPHHHHHHHHHHHPPHHHPHHPHPPHPHHHPPPPPPHHH*
3D58	58	*PHPHHHPHHHPPHHPHPHHPHHHPHPHPHHPPHHHPPHPHPPPPHPPHPPHHPPHPPH*
3D64	64	*PHHPHHPHHHPPHPHPPHPHPPHHHPHHPHHPPHHPHHPHHHPPHPHPPHPHPPHHHPHHPHHP*
3D67	67	*PHPHHPHHPHPPHHHPPPHPHHPHHPHPPHHHPPPHPHHPHHPHPPHHHPPPHPHHPHHPHPPHHHP*
3D88	88	*PHPHHPHHPHPPHHPPHPPHPPHPPHPPHPPHHPPHHHPPHHHPPHHHPPHHHPPHPHHPHHPHPPHPPHPPHHPPHPPHPPHHPPHP*
3D103	103	*PPHHPPPPPHHPPHHPHPPHPPPPPPPHPPPHHPHHPPPPPPHPPHPHPPHPPPPPHHHPPPPHHPHHPPPPPHHPPPPHHHHPHPPPPPPPPPHHHHHPPHPP*
3D124	124	*PPPHHHPHPPPPHPPPPPHHPPPPHHPPHHPPPPHPPPPHPPHPPHHPPPHHPHPHHHPPPPHHHPPPPPPHHPPHPPHPHPPHPPPPPPPHPPHHHPPPPHPPPHHHHHPPPPHHPHPHPHPH*
3D136	136	*HPPPPPHPPPPHPHHPHHPPPPHPHHHPPPPHPHPHHHHPPPPPPPPPPPHPPHPPPHPHHPPPHHPPHPPHPHPHPPPPPPPPHPPPHHHHHHPPPHHPPHHHPPPHHPHHHHHPPPPPPPPPHPPPPHPHPPPP*

ergy conformations for long *HP* sequences. We collected from the literature 13 sequences with more than 50 residues (listed in Table II). They have been actively studied on 2D square lattice or 3D cubic lattice by many researchers. Methods we considered here include the evolutionary Monte Carlo (EMC),[18] sequential importance sampling with pilot-exploration resampling (SISPER),[19] equienergy sampler (EES),[20] modified pruned-enriched Rosenbluth methods[21] (nPERMis[11] and nPERMh[12]), guided simulated annealing (GSA),[22] and contact interactions (CI) method.[23]

For all the 2D sequences (Table III), FRESS was able to find minimum energies that match the best known results using very little computation time—less than 20 min for each on a 1.4 GHz PC. For 3D sequences (Table IV), in addition to finding the lowest energies obtained previously, FRESS found energies lower than the best known ones for all sequences longer than 80 residues. Sequence 3D88 is an 88-residue long sequence designed to have the ground state energy of −72.[13] FRESS found this ground energy [Fig. 2(a)], whereas neither nPERMis nor nPERMh reached it. It was argued[12] that the failure of PERM-based algorithms to reach the designed optimum for this sequence is likely due to a very severe long-range interaction effect, similar to that in sequence 2D64. Sequences 3D103, 3D124, and 3D136 were modeled after real proteins cytochrome C, ribonuclease A, and staphylococcal nuclease, respectively.[24] Previously known lowest energies for them are −56,[25] −71[11,12] (by both nPERMis and nPERMh), and −80[11] (by nPERMis), respec-

TABLE III. Comparison of performances of different methods on 2D HP sequences. NA means data not available. The number in each cell is the minimum energy obtained by the corresponding method for the respective HP sequence.

2D seq.	EMC[a]	SISPER[b]	GSA[c]	nPERMis[d]	EES[e]	FRESS
2D50	−21	−21	NA	NA	−21	−21
2D60	−35	−36	−36	−36	−36	−36
2D64	−39	−39	−42	−42	−42	−42
2D85	NA	−52	−52	−53	−53	−53
2D100a	NA	−48	−48	−48	−48	−48
2D100b	NA	−49	−50	−50	−49	−50

[a] Reference 18.
[b] Reference 19.
[c] Reference 22.
[d] Reference 11.
[e] Reference 20.

TABLE IV. Comparison of performances of different methods on 3D sequences longer than 50 residues. The numbers are the minimum energies found by a particular method; and the numbers in parentheses are times in hours for the searches. "NA" means data not available. The parameter setting of FRESS for 3D88 and 3D103 are starting temperature $T_h=3.5$, lowest temperature $T_l=0.1$, and temperature decreasing by 0.995 geometrically; 10^6 moves at each temperature; and fragment lengths from 2 to 16. The setting for 3D124 and 3D136 is the same as above except the number of moves at each temperature is 10^7 and 5×10^6, respectively. The CPUs of the computers used to obtain the results: FRESS, 1.4 GHz PC; nPERMh, 1.84 GHz PC; nPERMis, 667 MHz PC. The reported lowest energies for sequences 3D124 and 3D136 were found in less than two weeks.

3D sequence	CI[a]	nPERMis[b]	nPERMh[c]	FRESS
3D58	−42	−44 (0.19)	−44 (1.10)	−44 (0.09)
3D64	NA	−56 (0.45)	−56 (0.47)	−56 (0.53)
3D67	NA	−56 (1.10)	−56 (0.33)	−56 (1.41)
3D88	NA	−69 (NA)	−69 (0.45)	−72 (5.03)
3D103	−49	−55 (3.12)	−55 (0.25)	−57 (4.47)
3D124	−58	−71 (12.3)	−71 (1.19)	−75[d]
3D136	NA	−80 (110)	NA	−83[e]

[a] Reference 23.
[b] Reference 11.
[c] Reference 12.
[d] Conformations with energy of −74 were found in 4.83 hours.
[e] Conformations with energy of −82 were found in 6.42 hours.

tively. We found new lower energies for all of them as −57, −75 and −83, respectively. Representative conformations with these new lowest energies are shown in Fig. 2.

B. Generalized Metropolis-Hastings framework

We found a novel generalization of the Metropolis-Hastings (MH) algorithm,[3,4] of which FRESS is a special case. The classic MH algorithm for sampling from the Boltzmann distribution $\pi(C)\propto\exp\{-E(C)/T\}$ starts from an initial configuration $C^{(0)}$ and then iterates as follows: at iteration $k+1$, we (a) generate a new configuration C' from a transition function $K(C^{(k)}\rightarrow C')$ chosen at will, and (b) let the next configuration $C^{(k+1)}=C'$ with probability p and let $C^{(k+1)}=C^{(k)}$ with probability $1-p$, where

$$p=\min\left\{1,\frac{\exp[-E(C')/T]K(C'\rightarrow C^{(k)})}{\exp[-E(C^{(k)})/T]K(C^{(k)}\rightarrow C')}\right\}.$$

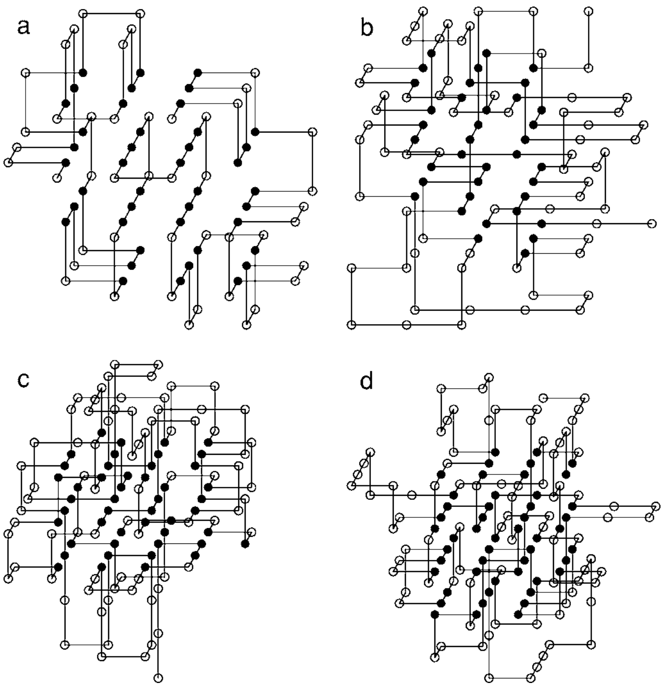

FIG. 2. Sample conformations with the minimum energies newly discovered by FRESS. (a) A conformation of sequence 3D88 with $E=-72$. (b) A conformation of sequence 3D103 with $E=-57$. (c) A conformation of sequence 3D124 with $E=-75$. (d) A conformation of sequence 3D136 with $E=-83$.

In some cases (such as FRESS), we may need the following scheme to make a proposal: (i) draw an auxiliary variable V from the current configuration $C^{(k)}$ according to $A(C^{(k)}\rightarrow V)$; (ii) with the help of the auxiliary variable V, draw C' from $T(\{C^{(k)},V\}\rightarrow C')$. If we have to follow the MH routine, we need to compute the probability of $C^{(k)}\rightarrow C'$ by integrating out V, which can be difficult or even impossible. To overcome this difficulty, we consider the *augmented* distribution $(1/Z)\exp(-E(C)/T)A(C\rightarrow V)$ in the *expanded* space of (C,V). The detailed balance condition on the augmented distribution suggests the following generalized MH rule: accept the proposed move with probability

$$p=\min\left\{1,\frac{\exp(-E(C')/T)A(C'\rightarrow V)}{\exp(-E(C^{(k)})/T)A(C^{(k)}\rightarrow V)}\times\frac{T(\{C',V\}\rightarrow C^{(k)})}{T(\{C^{(k)},V\}\rightarrow C')}\right\}. \qquad (1)$$

This generalization circumvents the need of integrating out V.

Note that the classic MH algorithm can also be formulated as above. For example, in the Ising model simulation, V can indicate the randomly selected spin for updating. However, variable V is always integrated out in computing the standard MH ratio. The usefulness of the generalized rule (1) can be best illustrated by the FRESS algorithm, in which the new configuration C' is obtained from the old configuration C by having a randomly selected segment deleted and then sequentially regrown (see Method for more details). Since the segment is selected at random and the regrowth process is carried out by sequential importance sampling, there are in general many possible ways to reach C' from C (unlike the spin selection in the Ising model). For example, multiple overlapping fragments could be regrown from C to reach the same C'. The standard MH recipe requires us to enumerate and add up all these possibilities, which is extremely difficult. In contract, the generalized MH rule allows us to take the segment selection as the auxiliary variable V (see Method) so as to avoid the difficult computation.

An interesting variation to the above approach is to propose also a new auxiliary variable V' from $A(C'\rightarrow V')$, and then accept (C',V') jointly with probability

$$p=\min\left\{1,\frac{\exp(-E(C')/T)}{\exp(-E(C^{(k)})/T)}\times\frac{T(\{C',V'\}\rightarrow C^{(k)})}{T(\{C^{(k)},V\}\rightarrow C')}\right\}. \qquad (2)$$

To see that this rule also maintains the detailed balance for the augmented distribution, we note that the new proposal

TABLE V. Performances of FRESS on ten benchmark 3D sequences under five parameter settings. Last row: the number of sequences for which the minimum energy conformations were found (the average time spent on the searches) in the end. Each cell contains the minimum energy (and the time spent in minutes on each run) reached under the respective condition for the respective sequence. For all the sequences, starting temperature T_h=3.5, minimum temperature T_l=0.1, and temperature decreasing by 0.98 geometrically. FRESS (best): fragment lengths are chosen between 2 and 12, with 5×10^4 moves at each temperature; $L_{\max}$=4: fragment lengths are chosen from 2 to 4, 2×10^5 moves at each temperature; L=12: fragment length is fixed to 12, 2×10^4 moves at each temperature; NIS: no importance sampling used, 8×10^4 moves at each temperature; NR: without DFS for fragment regrowth, 6×10^4 moves at each temperature.

Seq.	FRESS (the best setting)	$L_{\max}$=4	L=12	NIS	NR
1	−32(0.72)	−32(1.05)	−32(1.02)	−32(0.93)	−32(0.72)
2	−34(0.88)	−32(2.60)	−33(2.07)	−33(1.45)	−34(0.75)
3	−34(0.77)	−33(2.58)	−34(1.23)	−34(0.87)	−34(1.08)
4	−33(0.53)	−33(1.35)	−33(1.53)	−32(1.65)	−33(0.80)
5	−32(0.72)	−32(1.33)	−32(1.32)	−32(1.00)	−32(0.72)
6	−32(0.68)	−32(1.38)	−32(0.83)	−31(1.58)	−32(0.78)
7	−32(1.12)	−31(2.62)	−32(1.32)	−31(1.58)	−32(0.92)
8	−31(0.80)	−30(2.57)	−31(1.53)	−30(1.57)	−31(0.73)
9	−34(0.73)	−33(2.63)	−34(1.18)	−32(1.57)	−34(0.72)
10	−33(0.73)	−33(1.33)	−32(2.15)	−32(1.52)	−32(1.45)
Summ.	10(0.77)	5(1.94)	8(1.42)	3(1.37)	9(0.87)

attempts to move $(C^{(k)},V)$ in the expanded space to (C',V') according to $T(\{C^{(k)},V\}\rightarrow C')\times A(C'\rightarrow V')$, which suggests that (C',V') should be jointly accepted with probability

$$p=\min\left\{1,\frac{\exp(-E(C')/T)A(C'\rightarrow V')}{\exp(-E(C^{(k)})/T)A(C^{(k)}\rightarrow V)}\times\frac{T(\{C',V'\}\rightarrow C^{(k)})A(C^{(k)}\rightarrow V)}{T(\{C^{(k)},V\}\rightarrow C')A(C'\rightarrow V')}\right\}.$$

The cancellation of $A(C^{(k)}\rightarrow V)$ and $A(C'\rightarrow V')$ in the above formula leads to Eq. (2). As with Eq. (1), auxiliary variables V and V' are used here to avoid the difficulty of integrating out all the possibilities of reaching C' from C. One therefore only needs to record the configurations $C^{(k)}$, but not the V's along the Monte Carlo iterations (the sole function of V and V' is to make the move proposal and its acceptance more efficient).

Although some special forms of formulas (1) and (2) have been observed (e.g., in orientational-bias Monte Carlo[26] and the multiple-try Metropolis,[27] where variable V corresponds to the multiple proposed configurations), we were not able to find such a general extension of the MH framework in the literature.

III. DISCUSSION

FRESS has a few tuning parameters. Using the ten benchmark 48-residue sequences, we studied performances of FRESS under various parameter settings. The best setting we found for the fragment length range is $L_{\min}$=2 and $L_{\max}$=12 (see also the legend of Table V). The length sampling distribution $p(l)\propto 1/l$ seems to strike a good balance between computation efficiency and optimization performance. We note that when the fragments selected in each iteration were no longer than four residues, the method performed significantly worse; whereas when long fragments (l=12) are always selected, the method performed well but took significantly longer time. We also observed that without using importance sampling for regrowth, the method performed much poorly. But the method worked fine without using the depth-first-search[28] (DFS) (see Method) for fragment regrowth.

It is not always desirable to regrow long fragments because large scale moves tend to be rejected more frequently when the conformation is compact. To better understand how FRESS explores the conformation space, we clustered all distinct conformations visited by FRESS based on their contact maps.[29] Conformations with similar contact maps normally have similar overall topology. The number of clusters thus approximates the number of *conformation types*. The number of *distinct* conformations is another measure on how the space is explored. We found that with longer fragment lengths, the algorithm visited more clusters but fewer distinct conformations, whereas with shorter fragment lengths the algorithm visited more distinct conformations but fewer clusters. This indicates that updating long fragments helps the algorithm jump out of energy basins, whereas updating shorter fragments helps the algorithm better explore the local area. The combination of different fragment lengths allows FRESS to locate an energy basin more efficiently, but not being trapped there for long, which is quite analogous to MGMC (Ref. 8) and is largely responsible for FRESS' effectiveness. In comparison, the original CBMC (Ref. 7) only regrows a terminal portion of the conformation and the modification of Vendruscolo[30] allows the regrowth to take place in an internal portion, but only of fixed length. Selecting the fragment length l with probability $p(l)\propto 1/l$ in FRESS further accommodates the intuition that for real polymers moves involving many residues are less frequent than moves involving just a few.

In this study we proposed a new MCMC method that combines the benefit of traditional CBMC and the insight of MGMC, and obtained attractive results for *HP* sequence folding with notable computation efficiency. The flexibility of stochastic fragment regrowth as a single Monte Carlo move for chain polymer simulations and the theoretical framework we employed here to justify the new move allow the current method to be readily extendable to more realistic protein/biopolymer models. We have focused here mainly on structural optimization. We expect the method to be also useful for general sampling purpose after incorporating the weights of the old and new fragments into the generalized MH criteria as shown in formulas (1) and (2).

IV. METHOD

A. FRESS algorithm for chain polymer simulation

Suppose temperature T is fixed for now. At each iteration, FRESS first selects the fragment length $l \in [L_{\min}, L_{\max}]$ with probability proportional to $1/l$, where $L_{\min}$ and $L_{\max}$ (e.g., 2 and 12) are minimum and maximum allowed lengths of a fragment. An end position p of the fragment is then uniformly sampled between the first and $(n-l+1)$th residue, where n is the overall sequence length. The fragment containing residues from p to $p+l-1$ is then deleted from the current configuration and will be regrown. If the fragment does not contain a terminal residue, one of the two growth directions (forward or backward) is chosen at random. Let the starting residue be s and the end residue be e. Without loss of generality, we assume $s<e$ (forward growth) in the following discussion.

Let a conformation of the target chain polymer be denoted by $C=(x_1, \ldots, x_i, \ldots, x_n)$, where x_i is the coordinate of residue i, and let its partial conformation, in which residues from t to e are deleted temporarily, be denoted by $C_{t,e} = (x_1, \ldots, x_{t-1}, x_{e+1}, \ldots, x_n)$. Starting from s (until reaching e) the fragment is regrown in FRESS one residue at a time. To regrow residue t, we place it at one of the available positions adjacent to residue $t-1$ according to the probability $p_t^j \propto \exp(-(E_{t+1}^j - E_t)/T)$, where E_t is the energy of the partial conformation $C_{t,e}$, and E_{t+1}^j is the energy of the partial conformation $C_{t+1,e}^j = (x_1, \ldots, x_{t-1}, x_t^j, x_{e+1}, \ldots, x_n)$, where x_t^j denotes the jth possible position for residue t. Occupied positions and positions that cannot make residue e connected to $e+1$ no matter how one places residues from $t+1$ to e are given zero probability. In our implementation, a condition (necessary but not sufficient) for judging whether the chain can connect is $D(x_t^j, x_{e+1}) \leq |e-t+1|$, where $D(x_t^j, x_{e+1})$ is the lattice distance between the position of residue $e+1$ and the position of x_t^j.

Finally, if the regrowth is unsuccessful in the end, we just return to the old configuration, record it, and move on to the next iteration; if, on the other hand, we successfully regrow the fragment, the new conformation C' is accepted with probability

$$p = \min\{1, \exp(-(E_{C'} - E_C)/T) w(C)/w(C')\}, \tag{3}$$

where $E_{C'}$ and E_C are the energies of C' and the old conformation C, respectively, and $w(C')$ and $w(C)$ are the Rosenbluth weights of the regrown and the original fragments, respectively. The Rosenbluth weight $w(C')$ is computed by tracing the placement of each individual residue in the regrowth process and multiplying their placement probabilities together, as in CBMC.

B. Justification and modification

Because there are more than one way to reach a new configuration C' from the old configuration C (for example, overlapping fragments could be regrown to reach the same C'), the classical MH recipe is difficult to use but the generalized MH framework described in the Result section can be easily applied. More precisely, in FRESS, the auxiliary variable V for the proposal step consists of the fragment's length and starting position, which is independent of the actual configuration C. Thus, $A(C \to V)$ cancels out with $A(C' \to V)$ in formula (1), and $T(\{C, V\} \to C')$ is just the Rosenbluth weight $w(C')$, which leads to formula (3).

Since our primary goal here is not sampling but finding the optimal structural configuration, we implemented a few shortcuts to improve the computation efficiency. First, the weights $w(C)$ are ignored, and only energies of the new and old configurations are used to determine the acceptance probability of the proposed move. This saves 50% of the computing time and does not appear to affect the algorithm's ability to explore the space. Second, in order to improve the survival rate of the fragment growth, we adopted a DFS strategy, which guarantees that a valid conformation for the fragment can be found. During the placement of a residue of the fragment, we store all its possible positions that have nonzero sampling probabilities and are not currently selected for growth. When the growth of the fragment goes to a dead end at residue t, we come back to the residue that is closest to t and has at least one stored, unvisited position. This process is repeated until a valid conformation is found for the selected fragment. The employment of DFS improves the efficiency of ground state search, but makes it difficult to maintain the detailed balance for the sampling algorithm. Third, simulated annealing[31] is adopted to finally locate the optimal configuration.

ACKNOWLEDGMENTS

One of the authors (S.C.K.) was supported in part by a NSF CAREER Award No. DMS-0449204. Two of the authors (J.S.L. and J.Z.) were supported in part by the NSF Grant No. DMS-0244638 and NIH Grant Nos. R01-GM078990 and R01-HG02518-02.

[1] P. Bradley, K. M. Misura, and D. Baker, Science **309**, 1868 (2005).
[2] J. S. Liu, *Monte Carlo Strategies in Scientific Computing* (Springer-Verlag, New York, 2001).
[3] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, and A. H. Teller, J. Chem. Phys. **6**, 1087 (1953).
[4] W. K. Hastings, Biometrika **57**, 97 (1970).
[5] N. Madras and A. D. Sokal, J. Stat. Phys. **50**, 109 (1988).
[6] A. Sali, E. Shakhnovich, and M. Karplus, J. Mol. Biol. **235**, 1614 (1994).
[7] J. I. Siepmann and D. Frenkel, Mol. Phys. **75**, 59 (1992).
[8] J. Goodman and A. D. Sokal, Phys. Rev. Lett. **56**, 1015 (1986); J. S. Liu and C. Sabatti, Biometrika **87**, 353 (2000).
[9] B. Berger and T. Leighton, J. Comput. Biol. **5**, 27 (1998).
[10] K. Yue, K. M. Fiebig, P. D. Thomas, H. S. Chan, E. I. Shakhnovich, and K. A. Dill, Proc. Natl. Acad. Sci. U.S.A. **92**, 325 (1995).
[11] H. P. Hsu, V. Mehra, W. Nadler, and P. Grassberger, Phys. Rev. E **68**, 021113 (2003).

[12] W. Huang, Z. Lu, and H. Shi, Phys. Rev. E **72**, 016704 (2005).
[13] T. C. Beutler and K. A. Dill, Protein Sci. **5**, 2037 (1996).
[14] A. Shmygelska and H. H. Hoos, BMC Bioinf. **6**, 30 (2005).
[15] R. Unger and J. Moult, 5th Proc. Int'l. Conf. on Genetic Algorithms, 1993, p. 581.
[16] P. Grassberger, arXiv:cond-mat/0408571.
[17] www.fas.harvard.edu/~junliu/SupplementaryMaterials/HP3D64_figures.pdf
[18] F. Liang and W. Wong, J. Chem. Phys. **115**, 3374 (2001).
[19] J. L. Zhang and J. S. Liu, J. Chem. Phys. **117**, 3492 (2002).
[20] S. C. Kou, J. Oh, and W. H. Wong, J. Chem. Phys. **124**, 244903 (2006).
[21] U. Bastolla, H. Frauenkron, E. Gerstner, P. Grassberger, and W. Nadler, Proteins **32**, 52 (1998).
[22] C. I. Chou, R. S. Han, S. P. Li, and T. K. Lee, Phys. Rev. E **67**, 066704 (2003).
[23] L. Toma and S. Toma, Protein Sci. **5**, 147 (1996).
[24] E. E. Lattman, K. M. Fiebig, and K. A. Dill, Biochemistry **33**, 6158 (1994).
[25] M. Bachmann and W. Janke, J. Chem. Phys. **120**, 6779 (2004).
[26] D. Frenkel and B. Smit, *Understanding Molecular Simulation: From Algorithms to Applications* (Academic, San Diego, 1996).
[27] J. S. Liu, W. H. Wong, and F. Liang, J. Am. Stat. Assoc. **95**, 121 (2000).
[28] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, 2nd ed. (MIT, Cambridge, MA, 2001).
[29] M. Vendruscolo and E. Domany, Vitam. Horm. (San Diego, CA, U. S.) **58**, 171 (2000).
[30] M. Vendruscolo, J. Chem. Phys. **106**, 2970 (1997).
[31] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, Science **220**, 671 (1983).
[32] S. Kachalo, H. M. Lu, and J. Liang, Phys. Rev. Lett. **96**, 058106 (2006).