A new sequential importance sampling method and its application to the two-dimensional hydrophobic–hydrophilic model

Junni L. Zhang and Jun S. Liu[a)]
Department of Statistics, Harvard University, Cambridge, Massachusetts 02138

(Received 11 February 2002; accepted 24 May 2002)

The sequential importance sampling method and its various modifications have been developed intensively and used effectively in diverse research areas ranging from polymer simulation to signal processing and statistical inference. We propose a new variant of the method, sequential importance sampling with pilot-exploration resampling (SISPER), and demonstrate its successful application in folding polypeptide chains described by a two-dimensional hydrophobic-hydrophilic (HP) lattice model. We show by numerical results that SISPER outperformed several existing approaches, e.g., a genetic algorithm, the pruned-enriched Rosenbluth method, and the evolutionary Monte Carlo, in finding the ground folding states of 2D square-lattice HP sequences. In a few difficult cases, the new method can find the ground states without using any prior structural information on the chain. We also discuss the potential applications of SISPER in more general problems. © *2002 American Institute of Physics.* [DOI: 10.1063/1.1494415]

I. INTRODUCTION

The protein folding problem, i.e., the prediction of the native structure of a protein molecule from its amino acid sequence, has attracted much attention from the scientific community in the past 30 years. Despite the many years' assaults from top scientists, the problem is still largely unsolved. Recently, scientists have turned to the much simpler hydrophobic–hydrophilic (HP) lattice model[1,2] in order to gain some insight. It has been demonstrated that the HP model exhibits many important proteinlike properties. However, the folding prediction problem is ''NP-complete'' even for the HP model.[3] The difficulty lies in the rugged energy landscape of the large conformation space, which is characterized by many local minima and an exceedingly small number of global optimal states. Traditional methods such as molecular dynamics and Metropolis Monte Carlo have been widely used for predicting the native fold.[4] But these methods tend to get stuck in energy traps and usually take a long time to run for a chain of reasonable size. Many new Monte Carlo methods have therefore been proposed to improve the search for the lowest energy conformation, of which a significant portion are iterative and the others are progressive. The former class includes the simulated annealing,[5] Monte Carlo minimization,[6] the genetic algorithm,[7] and the recently developed evolutionary Monte Carlo.[8] The latter class, referred to as the chain growth methods, includes the core-directed chain growth method (CG) (Ref. 9) and pruned-enriched Rosenbluth method (PERM).[10] Various ways of incorporating structural information have also been suggested by heuristics to improve the performance of these algorithms.[8–12]

Chain growth methods are variants of sequential importance sampling (SIS), also known as the Rosenbluth method, which dates back to the 1950s.[13,14] The method was first developed to compute the partition function of a long-chain polymer modeled as a self-avoiding walk (SAW) on a k-dimensional lattice space. It was suggested that the self-avoiding conformation can be built up sequentially by adding one monomer at a time. The simple application of the sequential buildup, however, is not ideal in most cases when the chain is of moderate size—the simulated SAW can easily run into cages before it ends. Some improvement strategies have been proposed, including the lookahead strategies,[15,16] PERM,[17] and SIS with resampling (SISR).[18] In this paper, we propose a new SIS scheme, SIS with pilot-exploration resampling (SISPER) and test it on the 2D HP lattice model. A distinctive new feature of the method is that, in addition to a lookahead strategy when adding an amino acid residue, a small sample of pilot paths are sent out to gather future information, and this pilot information is used in weighting the partial chains for enrichment and pruning. Numerical results showed that this method is superior to previous methods in finding the ground state of a HP chain, even for long ones, without imposing any structural constraints.

Section II introduces the 2D HP model and the basic SIS method for constructing chain polymers. Section III describes a few strategies for improving the efficiency of SIS, including the lookahead strategy, the resampling method, PERM, and our new scheme pilot-exploration resampling (PER). Section IV reports in detail the application of SISPER to nine benchmark HP sequences we found in the literature. Section V probes into the ways of incorporating the secondary structure information in the folding simulation. Section VI concludes with a brief discussion on the potential application of SISPER in other problems.

II. THE 2D HP MODEL AND SEQUENTIAL IMPORTANCE SAMPLING

In the 2D HP model, a protein is abstracted as a sequence of hydrophobic (H for nonpolar) and hydrophilic (P

a)Author to whom correspondence should be addressed.

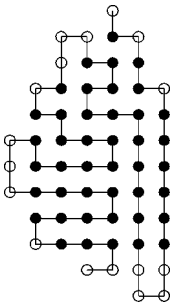

FIG. 1. A conformation of putative ground energy −36 found by SISPER for sequence 5 (the 60-mer sequence).

for polar) residues. The sequence occupies a string of adjacent sites on a two-dimensional square lattice. Only the self-avoiding conformations are valid with a simple interacting energy function: $\epsilon_{HH}=-1$ and $\epsilon_{HP}=\epsilon_{PP}=0$ for contacts between noncovalently bounded neighbors. The native structure of the sequence is defined as the conformation with the minimum energy. The 2D HP model tries to capture the phenomenon that in the folding of a real protein, hydrophobic residues tend to form a core in the structure shielded from the surrounding solvent by hydrophilic residues.[1,2] This knowledge has been incorporated in some search algorithms to improve efficiency.[9,11,12]

A conformation for a polypeptide chain of length $d+2$ can be represented by a vector of torsion angles $\boldsymbol{x}_d=(x_1,\ldots,x_d)$, where $x_t=0$ if turning 90° left, 1 if turning 90° right, and 2 if continuing ahead. The total energy of a conformation $\boldsymbol{x}_d$, $U_d(\boldsymbol{x}_d)$, is simply the sum of all pairwise interacting energies. For example, the total energy of the conformation in Fig. 1 is equal to −36. The native structure of the chain corresponds to the mode of the Boltzmann distribution,

$$\pi_d(\boldsymbol{x}_d)\propto\exp\{-U_d(\boldsymbol{x}_d)/\tau\}. \tag{1}$$

In addition to finding the native structure, it is also of interest to simulate from and to compute mathematical expectations with respect to Eq. (1).[4] In general, for any t that satisfies $1\leqslant t\leqslant d$, we call $\boldsymbol{x}_t=(x_1,\ldots,x_t)$ a partial conformation, and correspondingly, we define the energy function $U_t(\boldsymbol{x}_t)$ and the Boltzmann distribution $\pi_t(\boldsymbol{x}_t)$ as in Eq. (1). We also define $\pi_{t+k}(\boldsymbol{x}_t)$ as the marginal distribution of $\boldsymbol{x}_t$ under the distribution $\pi_{t+k}(\boldsymbol{x}_{t+k})$.

Iterative Monte Carlo methods simulate from the Boltzmann distribution by making a small change to the conformation $\boldsymbol{x}_d$ at each step so that the series of conformations form a Markov chain with Eq. (1) as its equilibrium distribution.[19] The SIS method does not simulate from Eq. (1) directly, but constructs $\boldsymbol{x}_d$ by adding one residue a time according to a series of conditional sampling distributions. Mathematically, the conformation $\boldsymbol{x}_d$ follows the distribution

$$p_d(\boldsymbol{x}_d)=q_1(x_1)q_2(x_2|x_1)\cdots q_d(x_d|\boldsymbol{x}_{d-1}),$$

where q_t is the conditional sampling distribution of x_t given previous partial conformation $\boldsymbol{x}_{t-1}$, and p_d is the sampling distribution for $\boldsymbol{x}_d$. In order to correct the bias introduced by such constructions, the conformation is weighted by

$$w(\boldsymbol{x}_d)=\pi_d(\boldsymbol{x}_d)/p_d(\boldsymbol{x}_d).$$

This is just the importance sampling, except that the sampling distribution is constructed sequentially. If we generate a sample $\boldsymbol{x}_d^{(1)},\ldots,\boldsymbol{x}_d^{(N)}$ from the procedure, we can approximate $\langle h\rangle_{\pi_d}$, the expectation of $h(\,)$ with respect to π_d, by the weighted average of the $h(\boldsymbol{x}_d^{(j)})$. An appropriate design of the conditional sampling distributions $q_t(\,)$ is the key to an innovative and effective SIS method.[19,20]

The Rosenbluth method[14] for estimating the total number of distinct conformations of a chain polymer is a special SIS with $q_t(x_t|\boldsymbol{x}_{t-1})=1/n_t$, where n_t is the number of available sites for placing x_t conditional on the first $t-1$ torsion angles. Each successfully constructed chain is weighted by $n_1\times\cdots\times n_d$ and the unsuccessful ones weighted by 0. The average of these weights gives rise to an unbiased estimate of the total number of conformations. More informatively, we can rewrite the conditional sampling distribution as

$$q_t(x_t|\boldsymbol{x}_{t-1})=\pi_t(x_t|\boldsymbol{x}_{t-1}),$$

where π_t is the uniform distribution on all self-avoiding walks with t torsion angles (corresponding to the Boltzmann distribution of $\boldsymbol{x}_t$ with a constant total energy). One can easily extend the Rosenbluth method to simulate foldings of a 2D HP sequence under the nonuniform Boltzmann distribution (1). A simple method is to let

$$q_t(x_t|\boldsymbol{x}_{t-1})=\pi_t(x_t|\boldsymbol{x}_{t-1})\equiv\frac{\exp\{-U_t(\boldsymbol{x}_t)/\tau\}}{\Sigma_{x_t'}\exp\{-U_t(\boldsymbol{x}_{t-1},x_t')/\tau\}},$$

where $U_t(\boldsymbol{x}_t)$ is the total energy (sum of all pairwise interacting energies) of the partial conformation $\boldsymbol{x}_t$.[20]

III. STRATEGIES FOR IMPROVING SEQUENTIAL IMPORTANCE SAMPLING

A. The δ-step lookahead

The self-avoiding chain constructed by the Rosenbluth method has a serious attrition problem. Typically it cannot run long enough. For example, on average it took more than 100 trials to generate a chain polymer of length 48.[21] Its application to the HP model performed even worse. Meirovitch[15,16] proposed a lookahead method to improve the plain SIS. A δ-step lookahead method uses the set of sampling distributions,

$$q_t(x_t|\mathbf{x}_{t-1}) = \pi_{t+\delta-1}(x_t|\mathbf{x}_{t-1}) \equiv \frac{\Sigma_{x_{t+1},\ldots,x_{t+\delta-1}} \exp\{-U_{t+\delta-1}(\mathbf{x}_{t-1},x_t,x_{t+1},\ldots,x_{t+\delta-1})/\tau\}}{\Sigma'_{x_t}\Sigma_{x_{t+1},\ldots,x_{t+\delta-1}} \exp\{-U_{t+\delta-1}(\mathbf{x}_{t-1},x'_t,x_{t+1},\ldots,x_{t+\delta-1})/\tau\}}.$$

In other words, we first explore all the possible continual conformations of the next δ residues exhaustively. These conformations are then grouped according to whether the immediate next residue is added by turning left, turning right, or continuing ahead. A conformation of the immediate next residue (x_t) is then sampled with probability proportional to the total sums of probabilities in each of the three groups.

B. Resampling and PERM

For long chains or low temperature τ, many chains ''die out'' (i.e., run into cages) before it can be grown to the full size, even when the lookahead method is used. Among the successfully constructed complete chains, their importance weights $w(\mathbf{x}_d)$ can be very skewed, resulting in many unrepresentative samples for π_d. To overcome some of these difficulties, Wall and Erpenbeck[22] described an enrichment method, which enriches those partial conformations (i.e., making more copies) that look promising. Grassberger[17] introduced an important modification of the enrichment method, named the Pruning Enrichment Rosenbluth Method (PERM), which stochastically prunes away partial conformations with weights lower than a threshold $W^<$, and enriches those with weights greater than another threshold, $W^>$.

In order to improve the SIS for a class of statistical computing problems, Liu and Chen[23] proposed a resampling method (SISR), which was later applied to the simulation of the HP model. A similar but more specialized technique was also developed independently for nonlinear filtering in signal processing.[24] The SISR has recently attracted much attention from the engineering community.[25] In this method, by comparing the intermediate importance weights of a set of partial conformations simulated in parallel, one resamples a new set of partial conformations (which effectively prunes away the ones with relatively small weights and multiply the ones with large weights). More precisely, suppose we have constructed a set of partial conformations, $\mathcal{S}_t=\{\mathbf{x}_t^{(1)},\ldots,\mathbf{x}_t^{(N)}\}$, with their partial importance weights, $w_t^{(1)},\ldots,w_t^{(N)}$, respectively. Then, we create a new set of partial conformations by sampling from $\mathcal{S}_t$ with probability $a_t^{(j)}, j=1,\ldots,N$, and weight each resampled conformation by $w_t^{(j)}/a_t^{(j)}$. The resampling probabilities can be chosen as proportional to $w_t^{(j)}$, but we show in the next section that some other choices can be more beneficial.

A main distinction between PERM and SISR is that with PERM one has to set the values for $W^<$ and $W^>$ *a priori*, which may require a few rounds of preprocessing and may take a substantial effort in order to get appropriate threshold values. The SISR method, on the other hand, considers all the partial conformations in a pool and enriches or prunes the partial conformations after considering their relative weights. This comparative strategy appears to be easier to automate. PERM, however, can always be run in serial, whereas the SISR has to grow multiple partial conformations in parallel.

C. Pilot-exploration resampling

In the sequential buildup of a long chain polymer, intuitively, we can approximate better the target distribution π_d and get closer to the global mode if we use more ''future'' information (i.e., the possible placements of the later residues). The δ-step lookahead strategy needs to examine all the possible paths of length δ and, thus, can only work for small δ. Our new strategy is to send out a pilot exploration ''team'' to spy on the future information, and compare this information across a set of partial conformations in order to decide the resampling probabilities.

Formally, suppose we have constructed a set of partial conformations $\mathcal{S}_t=\{\mathbf{x}_t^{(j)}, j=1,\ldots,N\}$. The Pilot-Exploration Resampling (PER) scheme consists of the following steps:

(1) For each partial conformation $\mathbf{x}_t^{(j)}$ in $\mathcal{S}_t$, a team of m ''members'' are sent out to explore Δ steps ahead. More precisely, we build on $\mathbf{x}_t^{(j)}$ the next Δ residues m independent times by the SIS to get $\{(x_{t+1}^{(j)l},\ldots,x_{t+\Delta}^{(j)l}), l=1,\ldots,m\}$. A ρ-step lookahead method can be applied in the generation of these pilot paths.
(2) For each generated pilot path l of conformation $\mathbf{x}_t^{(j)}$, compute its Boltzmann weight with ρ-step lookahead $b_t^{(j)l}=\pi_{t+\Delta+\rho-1}(\mathbf{x}_t^{(j)},x_{t+1}^{(j)l},\ldots,x_{t+\Delta}^{(j)l})$.
(3) The (unnormalized) resampling probability $a_t^{(j)}$ for conformation $\mathbf{x}_t^{(j)}$ is calculated as the αth power of the average of $b_t^{(j)l}, l=1,\ldots,m$.
(4) A resampling step for the set $\mathcal{S}_t$ is performed with the probability vector proportional to $\{a_t^{(1)},\ldots,a_t^{(N)}\}$.

The standard resampling methods include the simple random sampling and residual resampling[20] [see Sec. 2 of the Appendix]. Resampling can be conducted after every λ residues are added. After resampling, we again use the δ-step lookahead method to place the next residue.

The algorithm has eight user-set parameters, namely, τ, N, δ, λ, m, Δ, ρ, and α. The larger the temperature parameter τ is, the smaller the energy barriers. However, if τ is too large, we may not be able to obtain the conformation with minimum energy. In all of the examples, we explored with $\tau=0.1k, 1\leqslant k\leqslant 10$. The next parameter, N, is the number of conformations that are constructed in parallel except in the resampling step. Ideally, it could be as large as possible. Increasing N, however, will increase memory use and computation time. We tested with N ranging from 5 000 to 10 000 for all of the examples. We should also consider computational efficiency in setting the value for δ, since when we place a residue, for each of the N partial conformations, all the possible paths of length δ should be examined. The num-

TABLE I. 2D HP test sequences.

Sequence No.	Length[a]	Sequence[b]
1	20	HPHPPHHPHPPHPHHPPHPH
2	36	PPPHHPPHHPPPPPHHHHHHHPPHHPPPPHHPPHP
3	48	PPHPPHHPPHHPPPPPHHHHHHHHHHPPPPPHHPPHHPP HPPHHHHH
4	50	HHPHPHPHPHHHHPHPPPHPPPHPPPPHPPPHPPPHPHHH HPHPHPHPHH
5	60	PPHHHPHHHHHHHHPPPHHHHHHHHHHPHPPPHHHHHHHH HHHHPPPPHHHHHHPHHPHP
6	64	HHHHHHHHHHHHPHPHPPHHPPHHPPHPPHPPHHPPHPP HHPPHHPPHPHPHHHHHHHHHHHH
7	85	HHHHPPPPHHHHHHHHHHHHPPPPPPHHHHHHHHHHHPP PHHHHHHHHHHHPPPHHHHHHHHHHHPPPHPPHHPPHH PPHPH
8	100	PPPPPPHPHHPPPPPHHHPHHHHPHHPPPPHHPPHHPH HHHPHHHHHHHHHPHPHHPHHHHHHPPPPPPPPPPHHHH HHHPPHPHHHPPPPPPHPHH
9	100	PPPHHPPHHHHPPHHHPHHPHPHHHHHPPPPPPPHHHHH HPPHHHHHHPPPPPPPPPHPHHPHHHHHHHHHHPPHHHP HHPHPPHPHHHPPPPPPHHH

[a]The length denotes the number of residues of the sequence.
[b]Sequences 1–7 are taken from Ref. 8. Sequences 8 and 9 are taken from Ref. 28.

ber of such paths grows exponentially with δ (3^{δ} in the worst case, since there could be three possible ways to place the next residue at each step).

The rest of the user-set parameters are all related to the PER scheme. Parameter λ controls the frequency of resampling. We explored with $\lambda=2$ and $\lambda=4$ for all of the examples. Another strategy[23] is to monitor the coefficient of variation of the sequential importance weights [see Eq. (A1) in the Appendix]. In step (3) of the PER scheme, the average of $b_t^{(j)l}, l=1,\ldots,m$ gives an indication of the average target probability of future conformations that we will obtain if we further carry out Δ steps of SIS starting from $\boldsymbol{x}_t^{(j)}$ for pilot exploration. At each such SIS step, if we still use δ-step lookahead, the resampling step will be very slow, so instead we set ρ to be smaller than δ (e.g., $\rho=1$ or 2) and use ρ-step lookahead. The parameter Δ indicates how much "future information" we want to use in the resampling. Intuitively, the larger Δ is, the better the resampling scheme works. For longer sequence, larger Δ could be used, in order to better escape from local energy trap. The value of m should be set such that the average of Boltzmann weights from m paths is a reasonable approximation to the average target probability of future conformations. A rule of thumb is to set m to be of comparable magnitude to Δ (e.g., between Δ and 2Δ). The parameter α indicates how much confidence we want to put on the "future information;" the larger α is, the more confidence we have. Generally, α should be smaller than 1, since we do not want to "overtrust" the estimated future information based on a small pilot sample.

TABLE II. Comparison of SISPER with the genetic algorithm (GA), the evolutionary Monte Carlo (EMC), and PERM when no structural information is considered. For GA and EMC, the reported energy values are the lowest among five independent runs, and the values in the parentheses are the numbers of valid conformations scanned before the lowest energy values were found. For SISPER, $N=5000$ were used for each sequence. The CPU times spent on SPARC Ultra machines with 167 MHz are reported in the parenthese for PERM when available.

Sequence No.	Ground energy[a]	GA[b]	EMC[c]	PERM[d]	SISPER[e]
1	−9	−9 (30 492)	−9 (9 374)	−9	−9
2	−14	−14 (301 339)	−14 (12 447)	−14	−14
3	−23	−22 (126 547)	−23 (165 791)	−23	−23
4	−21	−21 (592 887)	−21 (74 613)	−21	−21
5	−36	−34 (208 781)	−35 (203 729)	−36	−36
6	−42	−37 (187 393)	−39 (564 809)	−40 (4 h)	−39
7	−52				−52
8	−47			−47 (1–2 days)	−48
9	−49			−48 (1–2 days)	−49

[a]Ground energy refers to the putative lowest energy by design or that found by previous methods. The putative ground energies for sequences 1–7 are recorded in Ref. 8. The putative ground energies for sequences 8 and 9 are recorded in Ref. 10.
[b]The results of the genetic algorithm reported in Ref. 7. The GA was run with the population size 200 for 300 generations.
[c]The results of the evolutionary Monte Carlo reported in Ref. 8. For sequence 1, EMC was run for 5000 iterations with the population size 100. For sequences 2–6, EMC was run for 1000 iterations with the population size 500.
[d]The results of PERM reported in Ref. 10.
[e]The results obtained by SISPER. The reported results are obtained with $\tau=0.2$ for sequence 4, $\tau=1.0$ for sequence 6, and $\tau=0.5$ for the other sequences.

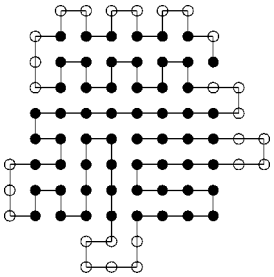

FIG. 2. A conformation of putative ground energy −52 found by SISPER for sequence 7 (the 85-mer sequence).

IV. NUMERICAL RESULTS

SISPER is applied to simulate the foldings of the nine 2D HP sequences in Table I, with $N=5{,}000$, $\delta=5$, $\lambda=2$, $m=20$, $\Delta=20$, $\rho=1$, and $\alpha=0.5$. We compare our method with a genetic algorithm (GA) implemented in Ref. 7, the evolutionary Monte Carlo (EMC) implemented in Ref. 8, and PERM in Ref. 10. The number of valid conformations scanned in GA and EMC for each sequence was listed in Ref. 8. Comparing this number with the number of conformations N in SISPER is not all that fair. In SISPER, each conformation is sequentially built up by the δ-step lookahead method; whereas in GA and EMC, each conformation results from a Markov Chain Monte Carlo step. Thus, SISPER takes more time in constructing each valid conformation, and it takes more time in resampling steps. Because of resampling, SISPER needs to store all of the partial conformations; whereas in GA and EMC, one only needs to store the current conformation for each member in the population (e.g., 100 or 500 in Ref. 8). The results are summarized in Table II. There was no information about time cost for EMC, so we could not make such a comparison. For sequences 8 and 9 (the 100-mer sequences), SISPER spent about 48 min in finding the putative ground energy conformation, on a 600 MHz Sony VAIO laptop with 128 MB memory.

The EMC failed to find the putative ground energy for sequence 5 (the 60-mer sequence), above the value −36 first found by PERM.[10] It has been argued[8] that direct comparison of PERM with EMC is unfair, because PERM may build up its chain from any part of a sequence, and thus making use of more information of the sequence. A direct comparison of SISPER with EMC, however, is fair since SISPER does not use further information in this regard. One of the ground energy states for sequence 5 is displayed in Fig. 1. For sequence 7 (the 85-mer sequence), the lowest energy found by the GA (Ref. 26) was −47, far from the putative ground energy value; EMC found the putative ground energy only when a strong secondary structure constraint (on 36 or 44 torsion angles in $\boldsymbol{x}_d$) was employed; SISPER found the putative ground energy easily without using any structural constraint (see Fig. 2). GA and EMC have not been tested on sequences 8 and 9 (the 100-mer sequences), but PERM has been applied to them. It is reported in Ref. 10 that the lowest energy obtained by PERM was −47 for sequence 8 and −48 for sequence 9 within 1-2 days of CPU time. With some structural constraint, PERM folded sequence 9 to a conformation with lower energy −49. For sequence 8, SISPER found a new conformation with lower energy −48 (see Fig. 3). SISPER also folded sequence 9 to a conformation with energy −49 (see Fig. 4) without using any structural constraint.

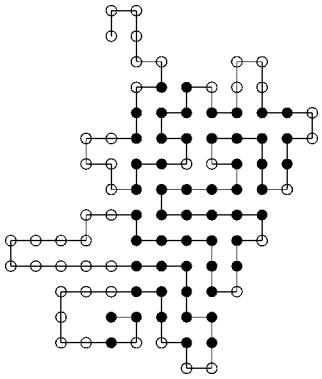

FIG. 3. A new conformation of lower energy −48 found by SISPER for sequence 8 (one of the 100-mer sequences).

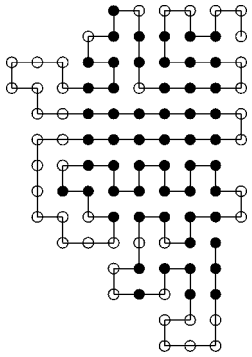

FIG. 4. A conformation of putative ground energy −49 found by SISPER for sequence 9 (one of the 100-mer sequences).

The most difficult sequence (and the only difficult sequence for SISPER) is sequence 6 (the 64-mer sequence). Without any structural constraints, all the available methods failed to fold the sequence to the putative ground state. It is argued in Ref. 10 that this sequence acts as a bottleneck for PERM due to the lack of a folding center in the protein. As shown in Fig. 5, the putative ground-state conformation does not show any advantage at first, but achieves the low energy by placing the end residues symmetrically with the beginning residues. This situation is especially unfavorable for SISPER. We show in the next section, however, that SISPER can also find the ground energy state by incorporating some structural information.

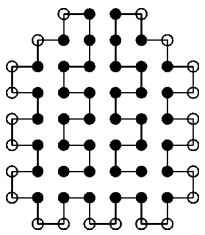

FIG. 5. A conformation of putative ground energy −42 for sequence 6 (the 64-mer sequence).

FIG. 6. Secondary structures folded by a subsequence of hydrophobic residues. (a) Alpha helix with direction 1. (b) Alpha helix with direction 2.

V. INCORPORATING STRUCTURAL INFORMATION

Various heuristics for incorporating structural information have been suggested. For example, the Core-Directed Chain Growth (CG) method[9] introduces search biases based on the knowledge that proteins have hydrophobic cores. It constructs first a core, which is as nearly square as possible, that can contain all hydrophobic monomers, and then use lookahead strategy to place segments of residues, penalizing hydrophobic residues outside the core and hydrophilic residues inside the core. CG found the ground energy state for sequence 6 easily. It is well-known, however, that HP sequences usually have ground states that are not maximally compact.[27] The ground energy conformation for sequence 5 (the 60-mer sequence) has an odd-shaped core, and thus the lowest energy obtained by CG was -35 instead of the putative ground energy -36. We suspect that the CG does not work well for sequence 8 or 9 either. EMC (Ref. 8) folded sequence 5, 6, and 7 to their putative ground energy states by incorporating secondary structure information, which, however, makes its performance depend on a secondary structure prediction procedure. By forbidding noncovalently bonded HP contacts, PERM (Ref. 10) folded sequence 6 and 9 to their putative ground energy states.

We can also incorporate easily into SISPER the secondary structure information. In both the δ-step lookahead sampling and PER, we can treat the constrained subsequence as a block of residues, and add the whole block to the previous partial conformation in one step. For sequence 6, if we constrain residues 1–10 to the alpha helix structure with direction 1, and residues 55–64 to the alpha helix structure with direction 2 as in Ref. 8 (see Fig. 6), SISPER can easily fold the chain to the putative ground energy state.

Thus, incorporating structural information has the potential of improving the performance of conformational search algorithms. The extent of improvement, however, depends strongly on the particular sequence of interest and the particular algorithm. Without any structural information, we have shown that SISPER works so far the best for the 2D HP benchmark sequences.

VI. CONCLUSION AND DISCUSSION

We have shown that SISPER is a top performer among the few available methods for folding the HP sequences in a two-dimensional lattice space. The extension of SISPER to work with 3D HP model is straightforward. With appropriate modifications and enhancements, we expect the method to be useful for dealing with real protein sequences. It is also worthwhile to note that the general SISR framework as outlined in the Appendix and the pilot exploration resampling idea can be applied more broadly to many other optimization and integration problems, such as those in statistical computing, signal processing, bioinformatics, artificial intelligence, etc.[19] Key aspects in the application of SISR to these problems are (a) a good design of the series of sampling distributions, the $q_t(\,)$'s, and (b) an innovative resampling scheme. We hope that the encouraging results reported in this article can interest some researchers to develop more sophisticated SIS methods.

ACKNOWLEDGMENTS

We are grateful to Dr. Rong Chen for his important suggestions regarding pilot-exploration resampling. This work was partly supported by the NSF Grants Nos. DMS-9803649 and DMS-0094613.

APPENDIX: MATHEMATICAL DEFINITIONS AND THE GENERAL FRAMEWORK

1. Sequential importance sampling

Suppose we are interested in computing the mathematical expectation of $h(\boldsymbol{x})$ with respect to a target distribution π, i.e.,

$$\langle h\rangle_\pi = \int h(\boldsymbol{x})\pi(\boldsymbol{x})d\boldsymbol{x}.$$

The importance sampling method suggests us to draw a random sample $S=\{\boldsymbol{x}^{(1)},\ldots,\boldsymbol{x}^{(N)}\}$ from a *trial* distribution, $p(\boldsymbol{x})$, and then estimate $\langle h\rangle_\pi$ by

$$\hat{\theta}=\frac{1}{W}\sum_{j=1}^{N} w^{(j)}h(\boldsymbol{x}^{(j)}),$$

where $w^{(j)}=\pi(\boldsymbol{x}^{(j)})/p(\boldsymbol{x}^{(j)})$ and $W=\Sigma_{j=1}^{N}w^{(j)}$.

Definition 1 (Ref. 18): *A set of weighted random samples* $\{(\boldsymbol{x}^{(j)},w^{(j)}\}_{j=1}^{m}$ *is called proper with respect to* π *if for any square integrable function* $h(\cdot)$,

$$\langle h\cdot w\rangle_p = c\langle h\rangle_\pi,$$

where c *is a normalizing constant common to all the samples*.

Thus, importance sampling is a procedure that creates a set of random samples properly weighted with respect to π. Suppose $\boldsymbol{x}$ can be decomposed into d-components, i.e., $\boldsymbol{x}=(x_1,\ldots,x_d)$. We need the following concept.[18]

Definition 2: *A probabilistic dynamic system is a sequence of probability distributions defined on spaces with increasing dimensions*: $\pi_t(\boldsymbol{x}_t)$ *for* $t=0,1,\ldots,d$, *where* $\boldsymbol{x}_t=(x_1,\ldots,x_t)$, *and* $\pi_d\equiv\pi$ *is the target distribution*.

In the SIS framework, $\pi_t(\boldsymbol{x}_t)$ should be a reasonable approximation to the marginal distribution $\pi_d(\boldsymbol{x}_t)$. For example, in the protein folding problem, we use

$$\pi_t(\boldsymbol{x}_t)\propto\exp\{-U_t(\boldsymbol{x}_t)/\tau\}, \tag{A1}$$

where $U_t(\boldsymbol{x}_t)$ is the energy function for the partial conformation $\boldsymbol{x}_t$.

The SIS is a recursive procedure for generating a set of random samples properly weighted with respect to π_t at all steps t:

(1) At stage t, generate x_t from $q_t(x_t|\boldsymbol{x}_{t-1})$, and let $\boldsymbol{x}_t=(\boldsymbol{x}_{t-1},x_t)$;

(2) Compute incremental weight,

$$u_t=\frac{\pi_t(\boldsymbol{x}_t)}{\pi_{t-1}(\boldsymbol{x}_{t-1})q_t(x_t|\boldsymbol{x}_{t-1})}$$

and let weight $w_t=w_{t-1}u_t$. Thus $\boldsymbol{x}_t$'s are properly weighted by w_t's with respect to π_t.

2. Re-examining the δ-step lookahead and resampling

In the δ-step lookahead, we sample x_t according to $q_t(x_t|\boldsymbol{x}_{t-1})=\pi_{t+\delta-1}(x_t|\boldsymbol{x}_{t-1})$. This step needs one to marginalize $x_{t+1},\ldots,x_{t+\delta-1}$ in the distribution $\pi_{t+\delta-1}$, which may not be feasible for large δ in many problems. Indeed, if we could let $\delta=d-t+1$ at each step in the lookahead method, the SIS would have produced a conformation $\boldsymbol{x}_d$ that follows the target distribution π exactly.

Suppose at step t, we have a collection of partial samples $\mathcal{S}_t=\{\boldsymbol{x}_t^{(j)},j=1,\ldots,N\}$, which are properly weighted by $W_t=\{w_t^{(j)},j=1,\ldots,N\}$ with respect to π_t. Given any (unnormalized) resampling probability vector $A_t=\{a_t^{(1)},\ldots,a_t^{(N)}\}$, we can conduct resampling and get a set of N^* samples that are (approximately) properly weighted with respect to π_t. The simple random sampling just draws N^* i.i.d. samples from $\mathcal{S}_t$ with the probability vector A_t. The residual resampling[20] with probability vector A_t proceeds as follows:

(1) Retain $k_j=[N^*a_t^{(j)}]$ copies of $\boldsymbol{x}_t^{(j)}$, $j=1,\ldots,N$. Let $N_r=N^*-k_1-\cdots-k_N$.
(2) Obtain N_r iid draws from $\mathcal{S}_t$ with probabilities proportional to $N^*a_t^{(j)}-k_j$, $j=1,\ldots,N$.
(3) For a newly obtained sample $\boldsymbol{x}_t^{(*j)}$, its new weight is $w_t^{(*j)}=w_t^{(l)}/a_t^{(l)}$, if $\boldsymbol{x}_t^{(*j)}$ is a resample of $\boldsymbol{x}_t^{(l)}$ in $\mathcal{S}_t$.
(4) Return the new set of samples $\mathcal{S}_t^*=\{\boldsymbol{x}_t^{(*j)},j=1,\ldots,N^*\}$ with weights $W_t^*=\{w_t^{(*j)}),j=1,\ldots,N^*\}$.

PER uses future information to assign the resampling probability vector A_t.

[1] K. F. Lau and K. A. Dill, Macromolecules **22**, 3986 (1989).
[2] K. A. Dill, S. Bromberg, K. Yue, K. M. Fiebig, D. P. Yee, P. D. Thomas, and H. S. Chan, Protein Sci. **4**, 561 (1995).
[3] B. Berger and T. Leighton, J. Comput. Biol. **5**, 27 (1998).
[4] A. R. Leach, *Molecular Modelling: Principles and Applications* (Longman, Harlow, England, 1996).
[5] S. Kirkpatrick, C. D. Gelatt, Jr., and M. P. Vecchi, Science **220**, 671 (1983).
[6] Z. Li and H. A. Scheraga, Proc. Natl. Acad. Sci. U.S.A. **84**, 6611 (1987).
[7] R. Unger and J. Moult, J. Mol. Biol. **231**, 75 (1993).
[8] F. Liang and W. H. Wong, J. Chem. Phys. **115**, 3374 (2001).
[9] T. C. Beutler and K. A. Dill, Protein Sci. **5**, 2037 (1996).
[10] U. Bastolla, H. Frauenkron, E. Gerstner, P. Grassberger, and W. Nadler, Proteins: Struct., Funct., Genet. **32**, 52 (1998).
[11] K. M. Fiebig and K. A. Dill, J. Chem. Phys. **98**, 3475 (1993).
[12] L. Toma and S. Toma, Protein Sci. **5**, 147 (1996).
[13] J. M. Hammersley and K. W. Morton, J. R. Stat. Soc. Ser. B. Methodol. **38**, 205 (1954).
[14] M. N. Rosenbluth and A. W. Rosenbluth, J. Chem. Phys. **23**, 356 (1955).
[15] H. Meirovitch, J. Phys. A **15**, L735 (1982).
[16] H. Meirovitch, Phys. Rev. A **32**, 3699 (1985).
[17] P. Grassberger, Phys. Rev. E **56**, 3682 (1997).
[18] J. S. Liu, R. Chen, and T. Logvinenko, in *Sequential Monte Carlo Methods in Practice*, edited by A. Doucet, N. de Freitas, and N. Gordon (Springer, Berlin, 2001), p. 225.
[19] J. S. Liu, *Monte Carlo Strategies in Scientific Computing* (Springer, Berlin, 2001).
[20] J. S. Liu and R. Chen, J. Am. Stat. Assoc. **93**, 1032 (1998).
[21] J. W. Lyklema and K. Kremer, J. Phys. A **19**, 279 (1986).
[22] F. T. Wall and J. J. Erpenbeck, J. Chem. Phys. **30**, 634 (1959).
[23] J. S. Liu and R. Chen, J. Am. Stat. Assoc. **90**, 567 (1995).
[24] N. J. Gordon, D. J. Salmond, and A. F. M. Smith, IEE Proc. F, Commun. Radar Signal Process. **140**, 107 (1993).
[25] *Sequential Monte Carlo Methods in Practice*, edited by A. Doucet, N. de Freitas, and N. Gordon (Springer, Berlin, 2001).
[26] R. König and T. Dandekar, BioSystems **50**, 17 (1999).
[27] 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).
[28] R. Ramakrishnan, B. Ramachandran, and J. F. Pekney, J. Chem. Phys. **106**, 2418 (1997).