LU TP 98-06
February 19, 1998

Dynamical-Parameter Algorithms for Protein Folding

Anders Irbäck[a]

Complex Systems Group, Department of Theoretical Physics
University of Lund, Sölvegatan 14A, SE-223 62 Lund, Sweden
http://thep.lu.se/tf2/complex/

To appear in *Proceedings of the HLRZ Workshop*
"Monte Carlo Approach to Biopolymers and Protein Folding",
Jülich, Germany, December 3–5, 1997

Abstract:
Two different dynamical-parameter algortihms are discussed, simulated tempering and the multisequence method. Using simulated tempering, the folding properties of 300 random sequences in a simple off-lattice model with only two amino-acid types, hydrophobic and hydrophilic, are investigated. A careful statistical analysis shows that the hydrophobic monomers are anticorrelated along the chains for good folding sequences. The multisequence method is the basis for a novel procedure for maximizing the stability of a given target structure. Tests of this sequence design procedure on the HP lattice model show that it can be extremely efficient.

[a] irback@thep.lu.se

1 Introduction

Studies of the statistical mechanics of protein folding are usually based on relatively short lattice chains. To be able to properly explore the conformational space of more realistic chains, it is of great interest to develop improved Monte Carlo (MC) methods. Although there has been some recent progress in this area,[1] this remains an important task.

In this paper I discuss two dynamical-parameter algorithms, simulated tempering [2,3,4] and the multisequence method.[4] It is shown that both these methods can speed up thermodynamic folding simulations by large factors compared to standard methods. In addition, I discuss a new sequence design procedure [5] which is based on the multisequence method.

As an example of an application of simulated tempering, I briefly discuss a study [6] of 300 randomly selected sequences in a simple two-dimensional (2D) off-lattice model with two amino-acid types, hydrophobic and hydrophilic. These sequences were subjected to both thermodynamic and kinetic simulations. About 10% of the sequences are found to meet criteria for good folding sequences. A careful analysis shows that the statistical distribution of hydrophobicity along the chains is nonrandom for good folding sequences. Interestingly, qualitatively similar deviations from randomness, corresponding to anticorrelations, are observed for a large group of real protein sequences.[7]

To further test the efficiency of simulated tempering, it has been applied to the standard HP model [8] on the square lattice, which is a widely used testbed for new algorithms. As an example, I consider a chain with 64 monomers that has recently been studied with two different methods.[9,10] In contrast to these algorithms, it turns out that simulated tempering is able to find the ground state of this sequence.

Next, I turn to the problem of designing sequences that have a given target structure as their unique native state. This goal can sometimes be accomplished by energy minimization with respect to sequence, for fixed structure.[11] A more generic but computationally much more difficult approach, is to optimize the Boltzmann weight of the target structure.[12,13] This problem requires exploration of both conformational and sequence degrees of freedom, and has in earlier methods [13,14,15] been approached by using simulated annealing in sequence space. This leads to a nested procedure in which the conformational space is examined over and over again for different fixed sequences.

Since each of these calculations for fixed sequence by itself is a non-trivial task, it is important to look for a different strategy. An obvious candidate is the multisequence method. In this method one simulates a joint probability distribution in sequence and structure. This single simulation replaces simulations of the Boltzmann distribution for a number of different sequences, and can be much faster than *one* of these. The joint distribution studied contains a set of tunable parameters that are crucial for the efficiency of the simulation. It turns out that these can be chosen in a convenient way. The number of sequences that can be studied in a multisequence simulation is of course limited. It is therefore of interest to incorporate a procedure for elimination of bad sequences. Two simple but useful ways of doing this elimination are discussed in Sec. 5.2. The full design algorithm, the multisequence method combined with this elimination, has been tested on the HP model. The results show that it can be extremely efficient.

The paper is organized as follows. In Sec. 2 I define the models studied. The dynamical-parameter method is described in Sec. 3. In Sec. 4 I discuss the study of folding properties in an off-lattice model. In Sec. 5 the sequence design procedure is presented; a fairly detailed description of the method is given, and the results of the numerical tests are discussed.

2 The Models

Two models are studied in this paper, the HP model [8] on the square lattice and the AB model.[16,6] Both these models have two amino-acid types, hydrophobic (H/A) and hydrophilic (P/B). The interaction parameters are chosen so as to favor the formation of a core of H/A monomers. Throughout the paper the sequence and conformation of a chain are denoted by $\sigma = \{\sigma_1, \ldots, \sigma_N\}$ and $r = \{\mathbf{r}_1, \ldots, \mathbf{r}_N\}$, respectively.

The HP model is defined by the energy function

$$E = \sum_{i<j} \epsilon(\sigma_i, \sigma_j) \Delta(\mathbf{r}_i - \mathbf{r}_j), \tag{1}$$

where $\Delta(\mathbf{r}_i - \mathbf{r}_j) = 1$ if monomers i and j are non-bonded nearest neighbors, and 0 otherwise. The interaction parameter $\epsilon(\sigma_i, \sigma_j)$ is -1 for a HH pair, and 0 for HP and PP pairs.

The AB model is a simple 2D off-lattice model, in which adjacent monomers along the chain are connected by rigid unit-length bonds. The energy function is given by

$$E = \frac{1}{4} \sum_{i=2}^{N-1} (1 - \cos\theta_i) + 4 \sum_{i=1}^{N-2} \sum_{j=i+2}^{N} \left[\frac{1}{r_{ij}^{12}} - \frac{C(\sigma_i, \sigma_j)}{r_{ij}^{6}} \right], \tag{2}$$

where θ_i denotes the bend angle at monomer i, and r_{ij} is the distance between monomers i and j. The sequence dependent coefficient $C(\sigma_i, \sigma_j)$ is taken to be 1 for an AA pair, 1/2 for a BB pair, and $-1/2$ for an AB pair.

3 The Dynamical-Parameter Method

The dynamical-parameter approach can be used both for calculating free energies [2] and for accelerating simulations of systems with a rugged energy landscape.[3] The basic idea is to enlarge the configuration space by promoting some parameter of the model to a dynamical variable. The most popular example is the simulated-tempering algorithm,[2,3,4] where the temperature is made a dynamical variable.

In simulated tempering one simulates the joint probability distribution

$$P(r, k) \propto \exp(-g_k - E/T_k), \tag{3}$$

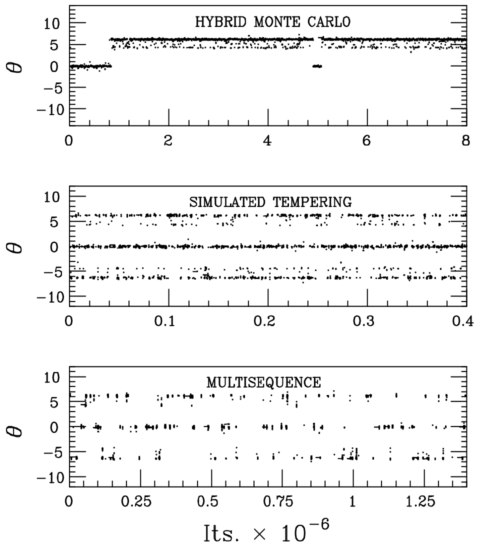

Figure 1: Evolution of $\theta = \sum_i \theta_i$ in three different simulations of an $N = 10$ AB chain. In the simulated-tempering run there are twelve allowed temperatures. The multisequence run covers all sequences with composition 7A+3B. All data shown correspond to the same temperature (0.15) and sequence (AABAAABBA).

where $\{T_k\}$ is a fixed set of temperatures that the system is allowed to visit. The tunable parameters g_k determine the marginal distribution of k, which can be written as

$$P(k) \propto \exp[-g_k - F(T_k)/T_k], \tag{4}$$

where $F(T_k)$ is the free energy. Hence, in order to have good mobility in k, these parameters must be chosen carefully. This is done by means of trial runs. The simulation of $P(r,k)$ can be carried out by using separate ordinary updates of r and k. The desired Boltzmann distribution, the conditional probability distribution $P(r|k) = P(r,k)/P(k)$, can be obtained directly from such a simulation, without any reweighting in r.

This algorithm can be immediately generalized. By making the sequence degrees of freedom a dynamical variable, one obtains the multisequence method.[4]

Simulated tempering and the multisequence method have both been successfully applied to the AB model (see Sec. 2).[4] They turn out to be much more efficient than standard methods. This is illustrated in Fig. 1, where these two methods are compared to the hybrid Monte Carlo method.[17,4] The conformation updates are the same, hybrid Monte Carlo, in all these three runs, which means that the CPU time per iteration is roughly the same. The chain studied here has ten monomers. It exhibits three distinct states at the temperature used. From Fig. 1 it can be seen that jumps between these three states occur much more frequently in the two dynamical-parameter runs than in the hybrid Monte Carlo run. One of the three states is, in fact, never visited in the hybrid Monte Carlo run, although this is longer than the other two. Furthermore, it should be stressed that each dynamical-

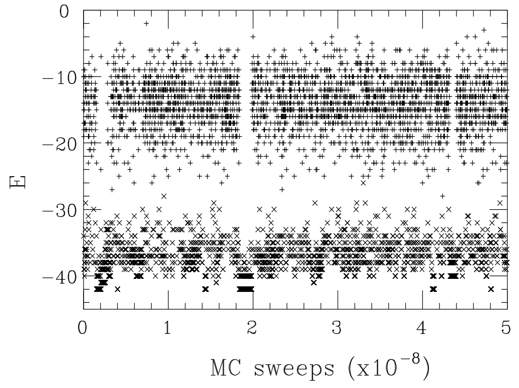

Figure 2: Evolution of the energy in a simulated-tempering run for a $N = 64$ HP chain. The ground state energy is -42. Shown are data corresponding to the lowest (0.4;×) and highest (1;+) allowed temperatures.

parameter run covers a number of different temperatures or sequences.

In order to compare simulated tempering to a genetic algorithm [9] and to the prune-enriched Rosenbluth method PERM,[10] it was also applied to the HP model, using a $N = 64$ chain that has been studied with both these methods (see Fig. 5 in the paper of Bastolla *et al.* [10]). The lowest energies that were found with the genetic algorithm and PERM are -37 and -40, respectively, while the ground state energy is -42. Figure 2 shows the evolution of the energy in a simulated-tempering run which took about 40 CPU hours on a DEC Alpha 200. In this run the ground state level is visited several times; the number of "independent" visits appears to be six. This example shows that simulated tempering is able to sample low-energy conformations in an efficient way in this model too.

A method closely related to simulated tempering is parallel tempering.[18,19,20] In parallel tempering there is again a fixed set of allowed temperatures, but instead of $P(r, k)$ in Eq. (3) one simulates the distribution

$$P(r_1, \ldots, r_K) \propto \prod_{k=1}^{K} \exp[-E(r_k)/T_k]. \tag{5}$$

This is done by letting K copies of the system evolve in parallel (r_k denotes the conformation of one chain). Two types of updates are used: ordinary updates of each r_k and swaps $r_k \leftrightarrow r_l$. A Metropolis test ensures that the swaps fulfill detailed balance. Notice that Eq. (5) does not contain any g_k parameters. These are not needed here since, by construction, there is always one system at each T_k. Another advantage of this algorithm is that it is very easy to parallelize. Parallel tempering has recently been used for simulations of a small peptide, Met-enkephalin.[21] We have tested this algorithm on a 3D extension [22] of the AB model, and found the efficiency to be comparable to that of simulated tempering. It is clear that a parallel version of the multisequence method, or any other dynamical-parameter algorithm, can be obtained in the same way.

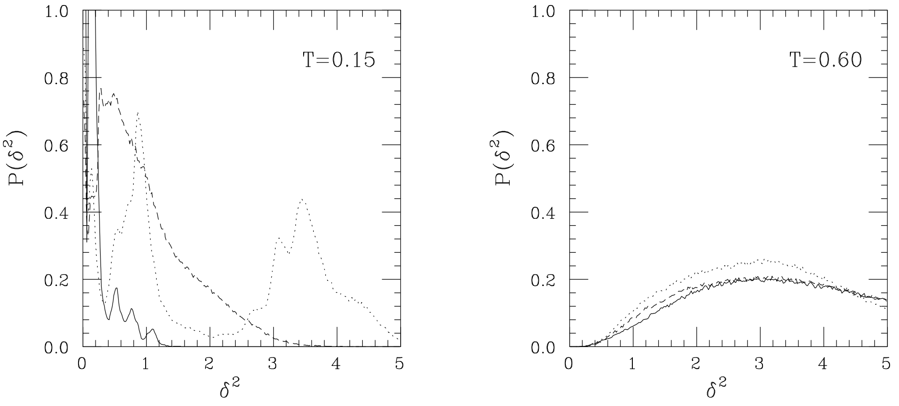

Figure 3: The δ^2 distribution for three different $N = 20$ AB chains at $T = 0.15$ and $T = 0.60$.

4 Nonrandom Hydrophobicity Patterns

In this section I briefly discuss a study[6] of 300 random sequences with composition 14 A + 6 B in the AB model. Both thermodynamic and kinetic simulations were performed for each of these sequences. The thermodynamic simulations were carried out by using simulated tempering with thirteen allowed temperatures, ranging from 0.15 to 0.60. Over this temperature interval a gradual compactification of the chains takes place. As expected, the size is found to be only weakly sequence dependent. The precise behavior in the compact low temperature regime is, by contrast, strongly sequence dependent. To see this, it is convenient to consider the probability distribution of the mean-square distance between two configurations, δ^2, for fixed sequence and temperature. In Fig. 3 three examples of δ^2 distributions are shown for $T = 0.15$ and $T = 0.60$. One of the three sequences (corresponding to the solid line) has a well-defined structure at $T = 0.15$. The other two have folding temperatures $T_f < 0.15$. Not unexpectedly, the folding temperature is found to be strongly sequence dependent.

The ground states for these sequences were determined by quenching a large number of different low-temperature conformations to zero temperature. This was done by means of a conjugate gradient method. It should be stressed that our ability to map out the ground states for these sequences relies heavily upon the efficiency of simulated tempering.

Having obtained the ground states, we measured the probability of finding these within a given number of ordinary (fixed-T) MC steps, starting from random coils. Twenty-five such experiments were performed for each sequence and temperature. The results of these kinetic calculations showed a much weaker sequence dependence than the δ^2 measurements. At first sight, this may seem to contradict the results of a recent, more detailed kinetic study by Veitshans *et al.*[23] It should therefore be stressed that the statement that the sequence dependence is weak refers to fixed absolute temperature rather than fixed physical temperature.

Based on these results, we decided to use a simplified, entirely thermodynamic criterion for good folding sequences. Those sequences with a high folding temperature ($T_f > 0.15$) were classified as good folders. This criterion is met by 37 of the 300 sequences.

With this sequence database at our disposal, a statistical analysis of the sequence patterns of folding and non-folding sequences was performed. A number of different variables, e.g. block and Fourier variables, were formed in order to look for characteristic features of good folders. The results of this analysis show that the hydrophobic monomers are anticorrelated along the chains for good folders. A simple manifestation of this is that the the number of clumps of A and B monomers along the chains tends to be larger for good folders than for random sequences. A similar analysis has also been performed for real protein sequences,[7] using the SWISS-PROT database [24] and binary hydrophobicity assignments. Again, clear evidence for nonrandomness was found. Furthermore, for a large group of proteins, it turns out that the deviations from randomness are qualitatively similar to those observed for good folders in the AB model, corresponding to anticorrelations along the chains.

5 Sequence Design

5.1 Basic formalism

In this section I discuss a new procedure, based on the multisequence method, for sequence design. The problem addressed is to find sequences σ that are thermodynamically stable in a given target structure r_0. The search takes place at some temperature T, the design temperature, and the desired goal is to maximize with respect to σ the Boltzmann weight

$$P(r_0|\sigma) = \frac{\exp[-E(r_0,\sigma)/T]}{Z(\sigma)}, \tag{6}$$

$$Z(\sigma) = \sum_r \exp[-E(r,\sigma)/T]. \tag{7}$$

In earlier methods [13,14,15] this optimization task has been approached by using simulated annealing in σ. The major difficulty then is to estimate $Z(\sigma)$, which has been done by using either cumulant (high-T) approximations [13,15] or a chain-growth MC method.[14] In the latter case one has a nested MC where the inner part by itself is a challenge.

The multisequence method offers a fundamentally different approach. In this method one replaces the simulations of $P(r|\sigma)$ for a number of different fixed σ by a single simulation of the joint probability distribution [cf Eq. (3)]

$$P(r,\sigma) = \frac{1}{Z}\exp[-g(\sigma) - E(r,\sigma)/T], \tag{8}$$

$$Z = \sum_\sigma \exp[-g(\sigma)]Z(\sigma). \tag{9}$$

The parameters $g(\sigma)$ determine the marginal distribution

$$P(\sigma) = \frac{1}{Z}\exp[-g(\sigma)]Z(\sigma) \tag{10}$$

and must, just as in simulated tempering, be chosen carefully. At first sight, it may seem that one would need to estimate $Z(\sigma)$ in order to obtain reasonable $g(\sigma)$. However, a convenient choice is

$$g(\sigma) = -E(r_0, \sigma)/T. \tag{11}$$

For this choice, one has

$$P(r_0|\sigma) = \frac{P(r_0, \sigma)}{P(\sigma)} = \frac{1}{ZP(\sigma)}, \tag{12}$$

so maximizing $P(r_0|\sigma)$ is in this case equivalent to minimizing $P(\sigma)$. This implies that bad sequences are visited more frequently than good ones in the simulation, which of course is unwanted in a sense. On the other hand, this property can be used to eliminate bad sequences.

Let me stress that the idea of using the multisequence method for this problem is attractive not only because of the simplicity of the scheme, but that there are good reasons to believe that this approach can be very efficient. First, the system tends to move more efficiently through conformational space if the sequence degrees of freedom are allowed fluctuate. As a result, simulating many sequences with the multisequence method can be much faster than simulating a single sequence with standard methods (see Fig. 1). Another appealing feature is that the optimization of the desired quantity $P(r_0|\sigma)$, which refers to a single structure, can be replaced by an optimization of the marginal probability $P(\sigma)$ [Eq. (12)].

5.2 Reducing the sequence set

The simple scheme outlined above is normally of little use on its own. With a large number of sequences, it becomes impracticable, especially since bad sequences tend to dominate in the simulation. It is therefore essential to incorporate a procedure for removal of bad sequences. This elimination can done in different ways. I will discuss two possibilities which will be referred to as $P(\sigma)$- and E-based elimination, respectively.

$P(\sigma)$-based elimination relies on the fact that bad sequences have high $P(\sigma)$ [see Eq. (12)]. The full procedure consists in this case of a number of ordinary multisequence runs. After each of these runs $P(\sigma)$ is estimated for all the N_r remaining sequences, and those having

$$P(\sigma) > \Lambda/N_r \tag{13}$$

are removed. Typical values of the parameter Λ are 1–2.

The procedure referred to as E-based removes sequences that do not have the target structure r_0 as their unique ground state. For each conformation $r \neq r_0$ visited in the simulation, it is checked, for each of the remaining sequences, whether $E(r, \sigma) \leq E(r_0, \sigma)$. Those sequences for which this is true are removed. With this type of elimination, it can happen that one removes the sequence that actually maximizes $P(r_0|\sigma)$ at the design temperature — the best sequence at this temperature does not necessarily have r_0 as its unique ground state. This should not be viewed as a shortcoming of the method. If it happens, it rather means that the design temperature is too high. E-based elimination is free from statistical errors in the sense that a sequence that does have r_0 as its unique ground state cannot be removed. Hence, in a very long simulation the surviving sequences are, by construction, precisely those that have r_0 as their unique ground state.

In our calculations stochastic moves in both r and σ are performed throughout the simulation. With E-based elimination, it is only the second part, the final estimation of $P(r_0|\sigma)$, that represents an actual multisequence simulation. For the first part, the elimination process, one could use an ordinary fixed-σ MC in r instead. Not surprisingly, it turns out, however, that the additional σ moves tend to speed up the elimination process.[5] In particular, they make the efficiency less T dependent, and, as a result, it becomes possible to work at a lower design temperature.

It should be pointed out that the elimination process serves two purposes. In addition to bringing down the number of sequences to a manageable level, it also changes the shape of the distribution $P(\sigma)$, which tends to become more uniform (see below). This is crucial for the performance of the multisequence method. With a random set of sequences, the simple choice of $g(\sigma)$ in Eq. (11) would be inappropriate.

5.3 Tests

This design procedure, based on the multisequence method with $P(\sigma)$- or E-based elimination, has been explored on the HP model using chains of length $N = 16$, 18 and 32. For $N = 16$ and 18 the results were checked against complete enumerations, and the method was found to reproduce the exact results very rapidly.

The $N = 16$ target structure was taken from a paper by Seno *et al.* [14] (see Fig. 1 in this paper). There is one sequence that has this structure as its unique ground state, and it was shown by these authors that their method is able to find this sequence, while two earlier methods [11,13] fail to do so. Our calculation was carried out starting from the set of all 2^{16} possible sequences and using E-based elimination. After a few CPU seconds on a DEC Alpha 200, all sequences except the correct one had been removed.

The $N = 18$ calculation was also performed using E-based elimination. The target structure studied here [5] is the unique ground state for seven different sequences. Starting from the set of all 2^{18} possible sequences, it took about one CPU minute to single out these seven sequences.

I now turn to our calculations for the $N = 32$ target structure shown in Fig. 4a, which I describe in more detail. For this system size it is not feasible to explore the entire sequence space. However, a given structure typically exhibits several positions where σ_i is effectively frozen to H or P. To locate such positions, ten short trial runs were performed, each started with a set of 10^5 random sequences. For the surviving sequences, the average σ_i was calculated for different i. The results turned out to be very stable, and a clear preference for either P or H was indeed observed at many of the positions along the chain. Based on these results, the different σ_i were divided into three groups. Those σ_i in the first two groups were clamped to H (filled

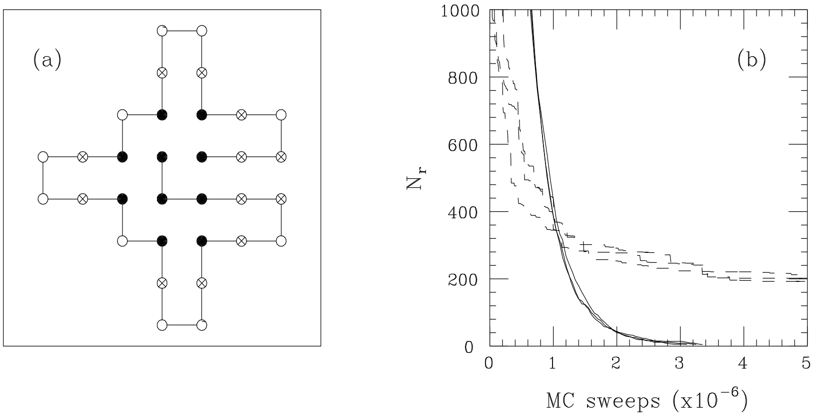

Figure 4: (a) Target structure for $N = 32$. Symbols are explained in the text. (b) The number of remaining sequences, N_r, against MC time for three runs with $P(\sigma)$-based elimination (full lines) and three with E-based elimination (dashed lines).

circles in Fig. 4a) and P (open circles), respectively, whereas the remaining twelve in the third group were left open (crosses). The design procedure was then applied to the corresponding restricted set of 2^{12} sequences, using both E- and $P(\sigma)$-based elimination.

Both these elimination methods turn out to work quite well. A comparison of the efficiencies of the two methods is given in Fig. 4b, where the number of remaining sequences, N_r, is plotted against MC time for three runs with each method ($5 \cdot 10^6$ MC sweeps correpond to about 20 CPU minutes). E-based elimination is very fast in the beginning, and a level is quickly reached at which it is easy to perform a final multisequence simulation for the remaining sequences. The curves level off at relatively high N_r, indicating that there are many sequences that have this structure as their unique ground state (these runs were continued until all three contained the same 167 sequences). The three runs with $P(\sigma)$-based elimination were carried out using $\Lambda = 2$ [see Eq. (13)] and a relatively short multisequence simulation for each elimination step. They were continued until five sequences or fewer were left. The results were checked against a very long multisequence simulation for a set of sequences obtained by E-based elimination. All the surviving sequences in the three runs with $P(\sigma)$-based elimination were among the top eight from this long simulation. This shows that results were stable, even though the eliminations were based on short runs.

The best sequence found for this structure is as follows:

$$\text{HHPP HHPP PPHP HPPP PHPH PPPP HHPP HHHH.} \qquad (14)$$

For this sequence, we estimate that $P(r_0|\sigma) = 0.42 \pm 0.02$ at the design temperature $T = 1/3$, and that the energy gap is 2.

The performance of this procedure is crucially dependent on the shape of the distribution $P(\sigma)$, especially, of course, if $P(\sigma)$-based elimination is used. One runs into problems if this distribution is dominated by a relatively small number of

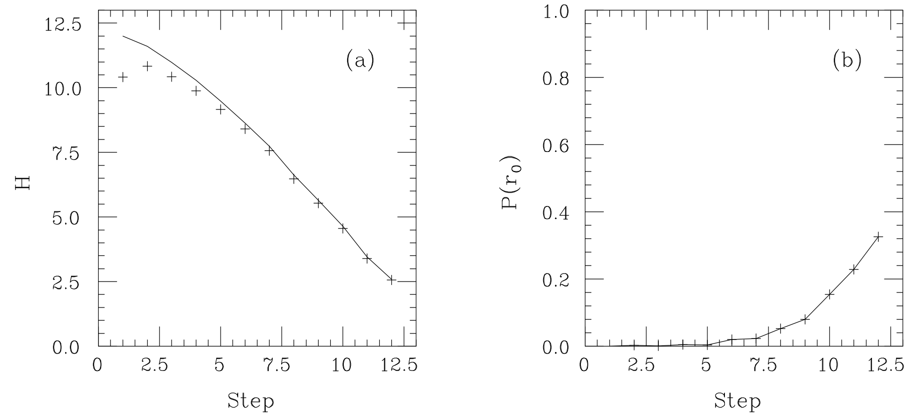

Figure 5: The evolution of (a) the entropy of $P(\sigma)$ and (b) the marginal probability $P(r_0)$ in a run with $P(\sigma)$-based elimination ($T = 1/3$, $\Lambda = 1$) for the target structure in Fig. 4a. The line in (a) shows $\log_2 N_r$, where N_r is the number of remaining sequences. After twelve elimination steps there were two sequences left.

sequences with high $P(\sigma)$. It is therefore interesting to see how the shape of $P(\sigma)$ evolves as the elimination process proceeds. Figure 5a shows the entropy of $P(\sigma)$,

$$H = -\sum_{\sigma} P(\sigma) \log_2 P(\sigma), \tag{15}$$

in a run for the target structure in Fig. 4a. With N_r remaining sequences, the maximal value of H is $\log_2 N_r$, corresponding to a uniform distribution $P(\sigma)$. As can be seen from Fig. 5a, after a few elimination steps, H is close to this limit. The desired behavior of the marginal distribution of r is in a sense the opposite, since the weight of the target structure should become large. The evolution of $P(r_0)$ in the same run is shown in Fig. 5b.

5.4 Summary and Outlook

Design of stable sequences is a computational challenge which requires exploration of both conformational and sequence degrees of freedom. The multisequence method provides a natural approach to the problem, and the tests above show that it can be implemented in a simple and very efficient way for the minimal HP model. To what extent it is useful for models with larger alphabets remains to be seen. The method can be applied to off-lattice models, and tests on a simple 3D model are under way.[25] It appears that $P(\sigma)$-based elimination, with some minor modifications, can be successfully applied in this case too.

For clarity, the discussion here has been focused on two simple schemes with either $P(\sigma)$- or E-based elimination. It is of course also possible, and probably advantageous, to use a mixture of these two. Furthermore, to facilitate the study of low design temperatures, it might be fruitful to combine the multisequence method with simulated tempering.

References

1. For a recent review, see B.J. Berne and J.E. Straub, *Curr. Opin. Struct. Biol.* **7**, 181 (1997).
2. A.P. Lyubartsev, A.A. Martsinovski, S.V. Shevkunov and P.N. Vorontsov-Velyaminov, *J. Chem. Phys.* **96**, 1776 (1992).
3. E. Marinari and G. Parisi, *Europhys. Lett.* **19**, 451 (1992).
4. A. Irbäck and F. Potthast, *J. Chem. Phys.* **103**, 10298 (1995).
5. A. Irbäck, C. Peterson, F. Potthast and E. Sandelin, preprint LU TP 97-33, cond-mat/9711092 (1997).
6. A. Irbäck, C. Peterson and F. Potthast, *Phys. Rev.* E **55**, 860 (1997).
7. A. Irbäck, C. Peterson and F. Potthast, *Proc. Natl. Acad. Sci. USA* **93**, 9533 (1996).
8. K.F. Lau and K.A. Dill, *Macromolecules* **22**, 3986 (1989).
9. R. Unger and J. Moult, *J. Mol. Biol.* **231**, 75 (1993).
10. U. Bastolla, H. Frauenkorn, E. Gerstner, P. Grassberger and W. Nadler, preprint cond-mat/9710030 (1997).
11. E.I. Shakhnovich and A.M. Gutin, *Proc. Natl. Acad. Sci. USA* **90**, 7195 (1993).
12. T. Kurosky and J.M. Deutsch, *J. Phys.* A **27**, L387 (1995).
13. J.M. Deutsch and T. Kurosky, *Phys. Rev. Lett.* **76**, 323 (1996).
14. F. Seno, M. Vendruscolo, A. Maritan and J.R. Banavar, *Phys. Rev. Lett.* **77**, 1901 (1996).
15. M.P. Morrissey and E.I. Shakhnovich, *Fold. Des.* **1**, 391 (1996).
16. F.H. Stillinger, T. Head-Gordon and C.L. Hirshfeld, *Phys. Rev.* E **48**, 1469 (1993).
17. S. Duane, A.D. Kennedy, B.J. Pendleton and D. Roweth, *Phys. Lett.* B **195**, 216 (1987).
18. C.J. Geyer and E.A. Thompson, University of Minnesota preprint (1993).
19. K. Hukushima and K. Nemoto, *J. Phys. Soc. (Jap)* **65**, 1604 (1996).
20. M.C. Tesi, E.J. Janse van Rensburg, E. Orlandini and S.G. Whittington, *J. Stat. Phys.* **82**, 155 (1996).
21. U.H.E. Hansmann, *Chem. Phys. Lett.* **281**, 140 (1997).
22. A. Irbäck, C. Peterson, F. Potthast and O. Sommelius *J. Chem. Phys.* **107**, 273 (1997).
23. T. Veitshans, D. Klimov and D. Thirumalai, *Fold. Des.* **2**, 1 (1997).
24. A. Bairoch and B. Boeckmann, *Nucleic Acid Res.* **22**, 3578 (1994).
25. A. Irbäck, C. Peterson, F. Potthast, E. Sandelin and O. Sommelius, in preparation.