Harnessing Graphical Structure in Markov Chain Monte Carlo Learning

Paul E. Stolorz
Jet Propulsion Laboratory
California Institute of Technology
Pasadena, CA 91109
pauls@aig.jpl.nasa.gov

Philip C. Chew *
University of Pennsylvania
Philadelphia PA
chew@upenn5.hep.upenn.edu

Abstract

The Monte Carlo method is recognized as a useful tool in learning and probabilistic inference methods common to many datamining problems. Generalized Hidden Markov Models and Bayes nets are especially popular applications. However, the presence of multiple modes in many relevant integrands and summands often renders the method slow and cumbersome. Recent mean field alternatives designed to speed things up have been inspired by experience gleaned from physics The current work adopts an approach very similar to this in spirit, but focusses instead upon dynamic programming notions as a basis for producing systematic Monte Carlo improvements. The idea is to approximate a given model by a dynamic programming-style decomposition, which then forms a scaffold upon which to build successively more accurate Monte Carlo approximations. Dynamic programming ideas alone fail to account for non-local structure, while standard Monte Carlo methods essentially ignore all structure However, suitably-crafted hybrids can successfully exploit the strengths of each method, resulting in algorithms that combine speed with accuracy. The approach relies on the presence of significant "local" information in the problem at hand. This turns out to be a plausible assumption for many important applications. Example calculations are presented, and the overall strengths and weaknesses of the approach are discussed.

Introduction

The Monte Carlo method has been used for a number of years to estimate complex multidimensional integrals for systems containing many degrees of freedom [5]. It has been particularly successful when applied to systems in which each variable interacts with other variables with a constant interaction strengh that remains the same, or similar, throughout the system. However, when significant inhomogeneities appear in the interaction strengths over different parts of the system, Monte Carlo methods become much less efficient. The problem is basically a manifestation of the multiple modes that appear in the integrand in this regime, which makes it difficult for the Monte Carlo procedure to jump between the various modes.

One well-known example of this distinction is supplied by the notorious Ising model [9]. Although the Ising model was originally developed as an abstraction to model physical systems, it turns out to be essentially equivalent to Hidden Markov Models and several other architectures useful in machine learning [2, 3, 8, 19]. It is therefore an excellent testing ground for describing and testing new Monte Carlo sampling ideas. A standard Ising model consists of a collection of binary elements with constant energetic interactions between all its elements. It can be simulated with reasonable efficiency by a Monte Carlo sampling procedure. However, as soon as the interaction strengths become heterogeneous (a common situation in the machine learning context), corresponding to a physical system known as an Ising spin glass, Monte Carlo simulations exhibit huge equilibration problems. In fact, understanding the extreme case of zero temperature for an Ising spin glass, which amounts to finding the minimum energy states, has been proven to be an NP-Complete problem.

The situation becomes even worse when various recently developed improvements, such as clustering methods, are considered [6]. These methods, based upon the simultaneous re-arrangement of blocks of variables, are often thought of as the introduction of auxiliary variables to the problem [2, 3]. They have been outstandingly successful at improving equilibration times for regular Ising models, but they break down completely in the spin glass case. Modifications that attempt to address the disorder present in spin glasses have been developed, but the results are somewhat disappointing [17].

A complementary approach to Monte Carlo methods that has existed in the physics literature for many years is the use of "transfer matrices" [9]. Transfer matrices can be used when the system under investigation consists only of local interactions in a small number dimensions. In this case the integrands or summands of interest can be calculated exactly by recursively building up the model under consideration a few units at a

*Current address:EECS Department, Massachusetts Institute of Technology

time. Recursive approaches such as this are extremely powerful when only local interactions are involved, because they are immune to equilibration problems, regardless of whether disorder is present. Of course, NP-Completeness does not vanish as a difficulty. It manifests itself in the need for an exponentially growing memory requirement in at least one dimension. However, for many problems such as the investigation of graphical models in data mining, this is not always a severe drawback, as the problems are often organized roughly as causal chains. Hidden Markov Models, for example, are arranged as linear chains. The difficulty is that for many interesting and useful models there may be a small number of "non-local" interactions between units quite distant in the chain. In this case a transfer matrix approach obviously cannot be used.

The Monte Carlo method described in this paper is designed to combine the best features of Monte Carlo and transfer matrix methods, and to apply them to problems where neither approach is successful on its own. The method is a variant of the Hastings generalization [4] of the classic Metropolis method [5]. It applies to models with discrete variables, in which a substantial number of interactions are local, but with a smaller number of non-local interactions present.

Review of Metropolis Monte Carlo

In this section standard Monte Carlo methods such as the Metropolis method will be reviewed, with an emphasis on the crucial points at which computational overhead due to heterogeneity enters the picture. Some of the ideas will be presented in a slightly non-standard way, which will help to motivate the modifications and alternatives that follow in the next section.

Suppose we have a system described by a cost function $E(C)$ (or energy function, or Hamiltonian function), associated with each of an enormous number of configurations C. These functions can be related by the Hammersly-Clifford theorem to an enormous class of probability distributions useful in the learning context, so they good starting point for a general Monte Carlo discussion. For concreteness, consider the calculcation of the "average energy" over the entire set of configurations, defined by

$$< E >= \sum_C E_C e^{-\beta E_C}/Z \qquad (1)$$

where β is a positive fixed parameter, and

$$Z = \sum_C e^{-\beta E_C} \qquad (2)$$

is a normalization constant. The average energy occurs frequently in the study of statistical mechanics, where β is interpreted as the inverse temperature. It is also extensively used in image processing applications, where β denotes the amount of noise in an image.

The question we want to answer is: how can $< E >$ be estimated without considering each and every configuration C? One simple way to do this is to evenly sample the space of configurations N times, and to estimate $< E >$ by

$$< E > \approx \sum_{i=1}^{N} E_i e^{-\beta E_i} / \sum_{i=1}^{N} e^{-\beta E_i} \qquad (3)$$

The main problem with this approach can be seen by rewriting $< E >$ in the following form:

$$< E >= \sum_E E N(E) e^{-\beta E}/Z \qquad (4)$$

where $N(E)$ is the number of states of energy E. The trade-off between the functions $N(E)$ and $e^{-\beta E}$ lies at the heart of statistical mechanics. Consider for example their product Π, which represents the probability of obtaining energy E,

$$\Pi(E) = N(E) e^{-\beta E}/Z \qquad (5)$$

Since the product is extremely peaked, all the important contributions to $< E >$ will come from a small band of the energy spectrum. Unfortunately, the even sampling of configuration space will select a great number of states with high energy (since there are a large number of these!), which will subsequently make very little contribution because of their small exponential weighting factor.

The essence of the importance sampling idea is to *generate* states in proportion to the probability distribution $e^{-\beta E_C}/Z$, hereafter referred to as the Boltzmann distribution. The average energy is then estimated from

$$< E >= 1/N \sum_i^N E_i \qquad (6)$$

In this way most of the generated states can be made to fall within the range of important energies, and they will all make meaningful contributions to the sum.

Let's see in a little more detail how this works in the Metropolis method, a widely used method for generating the Boltzmann distribution. We generate a chain of states, begining with state C_1, by randomly making some change to C_1 to create a new configuration C_2, This new configuration is accepted or rejected according to the transition probability distribution

$$W(C_1 -> C_2) = T(C_1 -> C_2) A(C_1 -> C_2) \qquad (7)$$

where $T(C_1 -> C_2)$ is the probability of selecting state C_2 given C_1, and

$$A(C_1 -> C_2) = min(1, P(C_2)/P(C_1)) \qquad (8)$$

is the probability of accepting the choice C_2 in the Markov chain. $T(C_1 -> C_2)$ can be almost any distribution, provided only that it is symmetric with respect to the choice of C_1 and C_2.

Under mild conditions, it can be proved that the states of this Markov chain will be asymptotically distributed with Boltzmann weight. Satisfied now that the chain of states $C_1, C_2, C_3 \ldots$ is producing states in

proportion to the Boltzmann distribution, we ask ourselves the following question: "What is the probability P_E, given energy E_1, of going to a new energy level E_2 by a Metropolis move?" Suppose that the Metropolis move only allows one variable in the whole state to be changed. Well, we have

$$P_E(E_1 -> E_2) = T_E(E_1 -> E_2)A(C_1 -> C_2) \quad (9)$$

where $T_E(E_1 -> E_2)$ is the probability of selecting a state of energy E_2 given a state of energy E_1, and where

$$T_E(E_1 -> E_2) \propto N(E) \quad (10)$$

since by randomly choosing a new state in the configuration space, there will be slightly more chance of obtaining a higher energy state than a lower energy state (there are more of them available). Hence the procedure samples energy E near the peak of the product distribution $\Pi(E)$: there is more chance of choosing a higher energy than lower energy by random selection, which is compensated for by smaller Boltzmann weight. The process of randomly selecting a new state "near" an old one in state space is really a *de facto* way of sampling the density of states $N(E)$ around the energy E. By choosing a nearby state, we make sure that the allowed range of E that is explored remains small for trial new states, so that we remain near the peak of $\Pi(E)$ when the Boltzmann weight is factored in, and so that the whole procedure is reasonably efficient.

Why is the above interpretation of the Metropolis procedure illuminating? Because it tells us precisely what motivates the first step in a Metropolis algorithm of generating new configurations by making relatively small changes to old configurations: namely, in the absence of any other information, *it is the only way we know of to sample the configuration space in proportion to the density of states N(E) in a controlled (usually small) energy interval around the old energy E.* This observation is crucial to understanding how Monte Carlo methods break down when applied to disordered systems. For a homogeneous system, for example, the configuration space can be searched ergodically by making small changes to a series of configurations. One will eventually wander over a large region of the configuration space. However, when disorder is introduced, this is no longer the case: by considering only small changes to successive configurations, we can easily end up being stuck in areas of configuration space separated by entropic barriers (i.e. large configuration changes) from other areas.

The interpretion also suggests what must be done in order to to generate a more efficient Monte Carlo procedure: find other ways of searching configuration space while retaining condition in italics above.

New method - Theory

Consider to begin with the concrete example of a simple 1-dimensional Ising model of discrete units S_i connected by nearest-neighbor interactions, together with a dilute concentration of long-range interactions. Denote this cost function by

$$H = H_0 + H_{nl} \quad (11)$$

$$= -\sum_{<ij>} J_{ij}S_iS_j - \sum_{>ij<} J_{ij}S_iS_j \quad (12)$$

where H_0 decribes the nearest-neighbor interactions (the summation $< ij >$), and H_{nl} the long-range interactions (the summation $> ij <$. When the interaction strengths J_{ij} in this model vary along the 1D chain, the energy surface defined by H_0 is in general a highly complex one, displaying many local minima, and gives rise to severe ergodicity problems when a Monte Carlo procedure is applied. For small system sizes, the difficulty can be cirmcumvented by using a computer to generate the exact partition function Z recursively, although this approach can no longer be used when long-range interactions are introduced. However, when the number of long-range interactions introduced is quite dilute, it can be expected that their main effect will be to alter the H_0 energy landscape in a limited way. It follows that a Monte Carlo procedure which makes maximum explicit use of the information contained in H_0 (as the transfer matrix does) will be extremely useful, particularly if it is able to sample H_0 efficiently across a large portion of the state space. It is just such an algorithm that is described here.

The first step towards this goal is to construct the exact density of states for the local cost function H_0. This can be achieved by adding one discrete variable at a time to build up a large system recursively. Given a model containing $L-1$ variables, the appropriate recursion is

$$\begin{aligned} N_0[L][E][s] &= N_0[L-1][E-\Delta(s)][s] \\ &+ N_0[L-1][E-\Delta'(s')][s'] \end{aligned} \quad (13)$$

where s and s' denote the possible values of the last variable in the system (assumed to have two values in this case), and E labels the various possible energy levels. The procedure is illustrated in Figure 1.

The idea of building up the density of states recursively by computer is due to Bhanot and Creutz [14]. Related recursions have been used by many authors. However, the uniquely attractive feature of the Bhanot and Creutz formalism is the fact that it can be generalized to perform the following task. By retaining in memory every step $N_0[L][E][s]$ of the build-up process, states of any given energy E can be straightforwardly reconstructed by simply back-tracking through the array $N_0[L][E][s]$, starting at the desired energy level E. This observation has been applied to the analysis of short-range spin glasses [12] and protein models [13]. It is a broadly useful extension for the deceptively simple reason that it allows states of identical energy H_0, but radically different structure, to be generated easily in a way that is directly controlled by the exact density of states. A small set of possible reconstructions is displayed in Figure 2. Note the similarity of the method to the classic technique of dynamic programming [11].

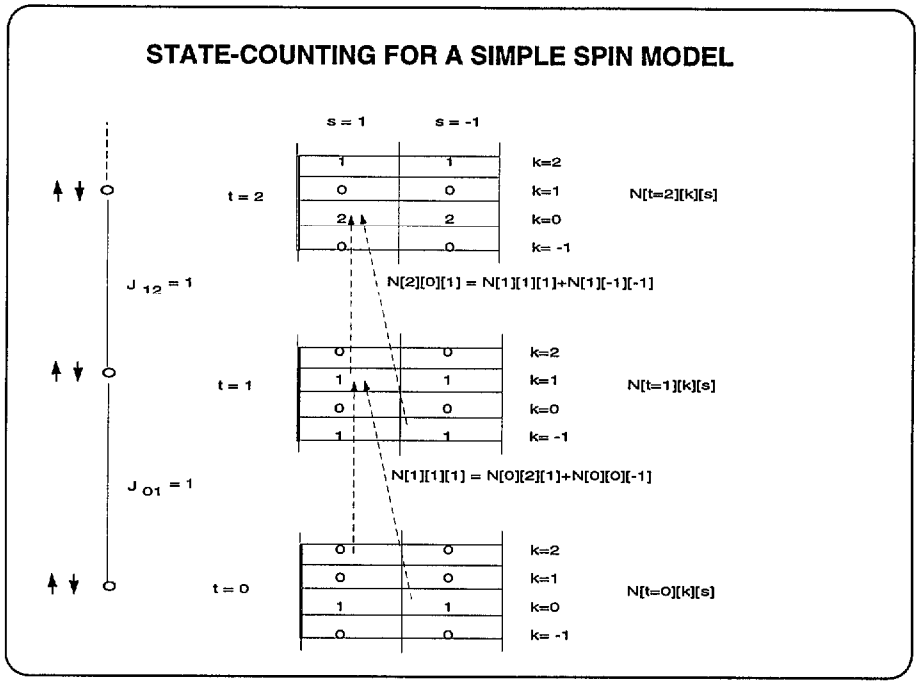

Figure 1: On the left are the first 3 elements of a 1D chain with binary variables (up or down), with the associated density of states on the right, showing the basic recursion for the first 3 steps.

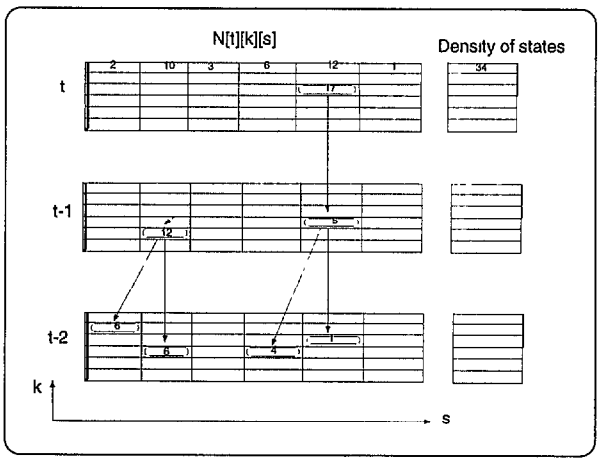

Figure 2: Recursive reconstruction of states with well-controlled energies from density of states array

The array $N_0[L][E][s]$, and its associated backtracking procedure, can now be used to embed the following unusual form of Monte Carlo method for the full Hamiltonian H. Given a state C_i, choose a new state C_j according to the transition matrix

$$T^0_{ij} = \frac{N_0(E_j)}{\sum_{k=i,i\pm1} N_0(E_k)} if k = i, i \pm 1 \quad (14)$$

$$= 0; \quad otherwise \quad (15)$$

This sampling is straightforward to implement by computer using the information contained in N_0. Now apply a generalized Metropolis-like acceptance criterion, with acceptance probability

$$A_{ij} = min(1, \frac{T^0_{ji} p_j}{T^0_{ij} p_i}) \quad (16)$$

$$p_j = e^{-\beta(H_0 + H_{nl})} \quad (17)$$

This type of generalized Metropolis procedure was first introduced by Hastings [4], and has also been used in physics computations based on renormalization ideas [10]. The key idea in our implementation is that of choosing the local density of states as the Hastings transition function. Because this function is built up recursively, while retaining a large amount of structural information about the configuration space of the system at each stage, this in turn allows large configurational rearrangements to be attempted with a high chance of success.

In order to guarantee asymptotic convergence to the Boltzmann distribution, it is advisable that the algorithm possess the property of detailed balance. It is easy to see that detailed balance is satisfied by this procedure: the overall transition probability W_{ij} satisfies

$$\frac{W_{ij}}{W_{ji}} = \frac{T^0_{ij} A_{ij}}{T^0_{ji} A_{ji}} = \frac{p_j}{p_i} \quad (18)$$

Any choice of selection probability of course satisfies detailed balance under this generalised Metropolis procedure. The original Metropolis method is recovered

by using a symmetric selection matrix. What is the advantage of using the asymmetric selection matrix T^0_{ij} instead of the symmetric Metropolis form? It is the fact that a single step of the sampling procedure is now capable of generating a new state which is completely different than the current state, and yet has identical local energy H_0. The algorithm is therefore totally impervious to the "landscape complexity" of the local Hamiltonian H_0, and is able to traverse the state space of H_0 across many local minima with high efficiency. It is only the modications that are introduced by the non-local portion of the Hamiltonian H_{nl} that can lead to trouble in the form of inefficient sampling of the state space. The method thus provides a mechanism for generating highly efficient non-local moves in the state space. An obvious caveat is of course that the local portion of the Hamiltonian must reflect some reasonable fraction of the structure of the overall full Hamiltonian.

The method has been introduced here in a manner familiar from statistical physics, in which the density of states is separated from an exponential Boltzmann factor. However, the general idea is by no means restricted to the situation of an integrand or summand separable in this way. A perfectly feasible alternative would be to generate the entire product $N_0(E)exp(-\beta E)$ recursively, and to use this function as the Hastings transition function. The overriding point is simply the ability to distinguish between a local portion of the Monte Carlo summand that can be built up recursively, and a separate non-local portion that can be dealt with as a Monte Carlo correction.

New Method - Simulations

We have investigated the properties of this new algorithm by introducing modifications to the simplest type of Markov model, namely a 1D Ising model. Local 1D Ising models containing 1000 spins were prepared, with random nearest-neighbor interactions. Non-local interactions were then inserted randomly between any two spins in the system, creating models in which on average 20% of the interactions were non-local. Note that 2D and 3D Ising models correspond to very particular choices for non-local interactions, so the formalism is actually quite general in scope. The model is a prototype of the most natural kind of generalized Hidden Markov Model. For example, it can be applied to model and predict 3-dimensional protein folding [13], in which non-local interactions are known to play an important role. Hidden Markov Models have been applied in the past to perform protein sequence alignment using local information only.

Monte Carlo simulations were run using the Metropolis algorithm, and various window sizes for the hybrid procedure discussed here. The window size represents the size of 1D subsystem to which the density of states procedure was applied. In order to measure the convergence properties of the various algorithms, the autocorrelation function of the energy and magnetization was then measured. The results are shown in Figure 3. This is a standard method of characterising equilibration times. Notice the dramatic improvement in equilibration times displayed in this figure as the "window size" W is increased from 1 (the regular Metropolis method) up to the limit of the system size. The procedure is clearly able to capture the structure of the local 1D energy function easily, and is only slowed down by the relatively small number of non-local interactions. Overall, an improvement of at least 2 orders of magnitude is supplied by the method.

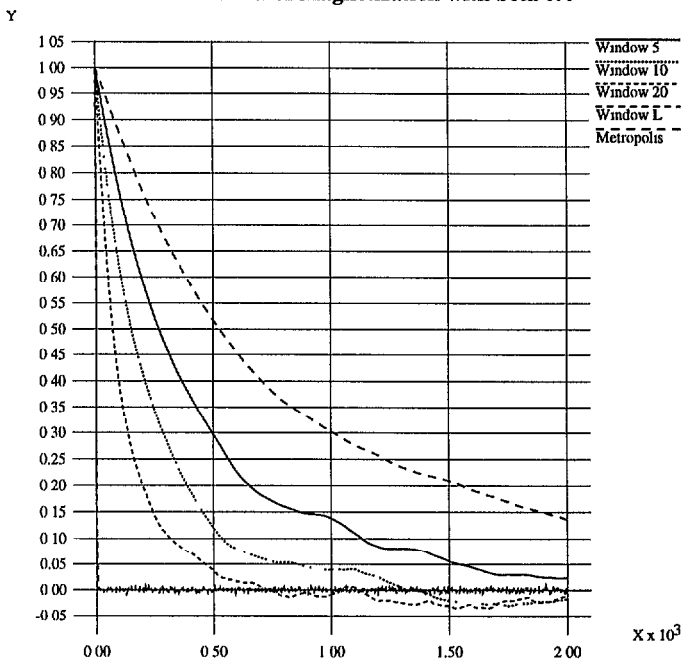

Figure 3: Results for 1D spin model with 20% non-local interactions. Shown are the magnetization autocorrelation functions for simulations using different window sizes (see text), including the Metropolis case and a window of size L covering the whole spin system.

We have also experimented with model Ising systems containing 10% non-local interactions, with similar results. In addition, equilibration times are known to depend strongly on the temperature at which the system is simulated. At high enough tempatures, all simulations are straightforward - it is only at low temperature that these issues become problematic. We find similar behaviour in the autocorrelation function over a range of temperatures, confirming the value of the approach as a general method.

Conclusions

What are the limitations and drawbacks of this approach to Monte Carlo simulation? To begin with, it requires the use of discrete variables. Secondly, the

problem tackled must have a reasonable amount of linear structure in its graphical representation. However, this condition is already substantially looser than the restriction to purely local interactions required by many approaches to learning in graphical models In fact, the basic recursive computational apparatus used as the underpinning of the hybrid Monte Carlo presented here is analagous to exact recursive EM procedures used in HMM learning, and to exact inference methods used by Bayesian nets [1, 13]. The unique feature of the current work is their use as a scaffold on which to build a general Monte Carlo procedure for more complex models as well.

One particular advantage of the method is that it enables extensive use of the Hastings Monte Carlo approach without being restricted to a choice of new configurations from univariate distributions. Historically this has been the main limitation hampering the liberal use of the Hastings approach. This aspect is in fact the key to the development of genuinely efficient Monte Carlo methods for complex interacting systems. It is not enough in general to generate new configurations by making substantial changes to just one variable, no matter how clever the change. Rather, large configurational changes in several variables simultaneously are required, while at the same time avoiding large alterations to the energy of the states involved. The method outlined here is a step towards the realization of this goal for certain classes of graphical models.

Acknowledgements

The research described in this paper was performed at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration. Support for PCC was provided by the University Scholars Program at the University of Pennsylvania.

References

[1] D. Heckerman, M. Jordan and P. Smyth (1995), "Conditional Independence graphs for Hidden Markov Probability Models", *Tech. Report submitted for publication.*

[2] A.F. Smith and G.O. Roberts (1993). "Baycsian computation via the Gibbs sampler and related Markov chain Monte Carlo methods (with discussion)", *J. Roy. Stat. Soc.* B55, 3-23.

[3] J.E. Besag and P.J. Green (1993). "Spatial Statistics and Bayesian computation (with Discussion)", *J. Roy. Stat. Soc.* B55, 25-37.

[4] W.K. Hastings (1970). "Monte Carlo sampling methods using Markov chains and their applications", *Biometrika*, 57, 97-109.

[5] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller and E. Teller (1953). "Equations of state calculations by fast computing machines", *J. Chem. Phys.* 21, 1087-1091.

[6] R.H. Swendsen and J.S. Wang (1987). "Nonuniversal critical dynamics in Monte Carlo simulations", *Phys. Rev. Lett.* 58, 86-88.

[7] R.M. Neal 1994. "Sampling from Multimodal Distributions Using Tempered Transitions", *Univ. of Toronto Tech. Report 9421.*

[8] J. York and D. Madigan (1992). "Markov Chain Monte Carlo Methods for Hierarchical Bayesian Expert Systems", *Proc. 4th Int. Workshop on AI and Statistics*, 433-439.

[9] R.P. Feynman (1972). *Statistical Mechanics: A Set of Lectures.*

[10] K.E. Schmidt (1983). *Phys. Rev. Lett.* 51, 2175-2178.

[11] R. Bellman (1957). *Dynamic Programming.*

[12] P. Stolorz (1993). "Recursive Approaches to Short-range Disordered Systems in the Low-temperature Regime", *Phys. Rev.* B48, 3085-3091.

[13] P. Stolorz (1994). "Recursive Approaches to the Statistical Physics of Lattice Proteins", *Proc. 27th Hawaii International Conf. on System Sciences*, Vol V, 316-325.

[14] G. Bhanot, M. Creutz and J. Lacki (1992), "*Phys. Rev. Lett.* 69, 1841-1843.

[15] L. Saul and M. Jordan (1995). "Exploiting Tractable Substructure in Intractable Networks", *Adv. in Neur. Inf. Proc.Syst.* to appear.

[16] E. Marinari and G. Parisi (1992). "Simulated tempering: a new Monte Carlo scheme", *Europhys. Lett.* 19, 451-458.

[17] D. Kandel, E. Domany and A. Brandt (1989). "Simulations withoug critical slowing down: ising and three-state Potts models:, *Phys. Rev.* B40, 330-344.

[18] B.A. Berg and T. Celik (1992). "New approach to spin-glass simulations", *Phys. Rev. Lett.* 69, 2292-2295.

[19] R.M. Neal (1994). "Probabilistic inference using Markov Chain Monte Carlo methods", *Univ. of Toronto Tech. Report CRG-TR-93-1.*