Available online at www.sciencedirect.com
ScienceDirect
Mathematics and Computers in Simulation 80 (2010) 1056–1067
MATHEMATICS AND COMPUTERS IN SIMULATION
www.elsevier.com/locate/matcom

Application of Biased Metropolis Algorithms: From protons to proteins

Alexei Bazavov [a,b,*], Bernd A. Berg [a,b], Huan-Xiang Zhou [a,b,c]

[a] *Department of Physics, Florida State University, Tallahassee, FL 32306-4350, United States*
[b] *School of Computational Science, Florida State University, Tallahassee, FL 32306-4120, United States*
[c] *Institute of Molecular Biophysics, Florida State University, Tallahassee, FL 32306-4380, United States*

Received 24 July 2008; received in revised form 10 March 2009; accepted 12 May 2009
Available online 22 May 2009

Abstract

We show that sampling with a biased Metropolis scheme is essentially equivalent to using the heatbath algorithm. However, the biased Metropolis method can also be applied when an efficient heatbath algorithm does not exist. This is first illustrated with an example from high energy physics (lattice gauge theory simulations). We then illustrate the Rugged Metropolis method, which is based on a similar biased updating scheme, but aims at very different applications. The goal of such applications is to locate the most likely configurations in a rugged free energy landscape, which is most relevant for simulations of biomolecules.
Published by Elsevier B.V. on behalf of IMACS.

Keywords: Biophysics; Higher energy physics; Markov chain Monte Carlo

1. Introduction

Consider a random variable y which is sampled with a probability density function (PDF) $P(y)$ on an interval $[y_1, y_2]$. The cumulative distribution function (CDF) is defined as

$$z = F(y) = \int_{y_1}^{y} P(y')\mathrm{d}y' \quad \text{and} \quad P(y) = \frac{\mathrm{d}F(y)}{\mathrm{d}y}, \tag{1}$$

where we assume that $P(y)$ is properly normalized so that $F(1) = 1$ holds. Let us consider two popular local algorithms to achieve this sampling of y in a Markov chain Monte Carlo process.

1.1. Heatbath algorithm (HBA)

The HBA [12] generates y by converting a uniformly distributed random number $0 \le z < 1$ into

$$y = F^{-1}(z). \tag{2}$$

* Corresponding author at: Department of Physics, University of Arizona, Tucson, AZ 85721, United States.
E-mail address: bazavov@physics.arizona.edu (A. Bazavov).

0378-4754/$36.00 Published by Elsevier B.V. on behalf of IMACS.
doi:10.1016/j.matcom.2009.05.005

We define the *acceptance rate* by the number of accepted changes divided by the total number of proposed moves. Thus the acceptance rate of the HBA is always 1 (a new value of y is generated on *every* step). In simulations the inversion of the CDF (1) may be unacceptably slow or the CDF itself may not be *a priori* known. Then one has to rely on other approaches.

1.2. Metropolis algorithm

In the conventional Metropolis scheme [21] (for historical accounts see Ref. [17] and for a textbook treatment [5]) y_{new} is generated uniformly in the range $[y_1, y_2]$ (we refer to this as *proposal*) and then accepted with probability (*accept/reject step*)

$$p_{\text{Met}} = \min\left\{1, \frac{P(y_{\text{new}})}{P(y_{\text{old}})}\right\}. \tag{3}$$

This process may have a low *acceptance rate* in the region of interest. Possible remedies are to decrease the proposal range, which makes the moves small, or propose a move multiple times (i.e., *multi-hit*) Metropolis, which needs a fixed number of hits. Both remedies are worse than an efficient HBA, which for many systems is the optimal solution in the considered class of local algorithms.

We also note that in certain cases faster decorrelation is achieved by using an overrelaxation algorithm [13,10,1] in which the proposed value is chosen as far as possible from the previous one. For such cases as $U(1)$ and $SU(2)$ gauge theories the overrelaxation is microcanonical, i.e., $P(y_{\text{new}}) = P(y_{\text{old}})$, thus it has to supplement Metropolis, HBA or BMA. In a simulation one normally tunes the ratio between overrelaxation and other algorithms for optimal performance. For instance, in a recent study of $U(1)$ gauge theory at finite temperature [6] on large volumes one BMA sweep was supplemented by two overrelaxation sweeps. The performance of the overrelaxation algorithm mixed with HBA and BMA was also studied for the case of the fundamental-adjoint $SU(2)$ lattice gauge theory [3].

1.3. Biased Metropolis Algorithm (BMA)

Hastings [19] identified proposal probabilities, which are more general than those of the conventional Metropolis scheme, but gave no guidance whether some probabilities may be preferable over others.

If one does not propose y_{new} uniformly, the name *Biased Metropolis Algorithm (BMA)* is often used. Some biased Metropolis simulations can be found in the literature where the bias is introduced in an ad hoc way [11,22,14,16,26]. However, it appears that the answer to the question, when to use biased Metropolis updating and when not, is far from clear.

The biased Metropolis scheme [4,9,2] we discuss in the following makes it possible to approximate heatbath probabilities. Like the conventional Metropolis scheme it can be constructed for more general situations than the HBA, but it achieves the performance which is typical for an efficient HBA.

Let us discretize y into n bins as

$$y_1 = y^0 < y^1 < y^2 < \ldots < y^n = y_2 \tag{4}$$

where *lengths* of the bins are

$$\Delta y^j = y^j - y^{j-1}, \quad \text{with} \quad j = 1, \ldots, n. \tag{5}$$

A BMA can then be summarized by the following steps:

- Propose a new value y_{new} by first randomly picking a bin j_{new} and then proposing y_{new} uniformly in the given bin. (r_1, r_2 are uniformly distributed):

$$j_{\text{new}} = 1 + \text{Int}[n\, r_1] \quad \text{and} \quad y_{\text{new}} = y^{j_{\text{new}}-1} + \Delta y^{j_{\text{new}}} r_2, \tag{6}$$

  where $\text{Int}[n\, r_1]$ denotes rounding to the largest integer $\leq n\, r_1$.
- Locate the bin to which y_{old} belongs: find j_{old} which satisfies the condition

$$y^{j_{\text{old}}-1} \leq y_{\text{old}} \leq y^{j_{\text{old}}}. \tag{7}$$

- Accept y_{new} with probability:

$$p_{\text{BMA}} = \min\left\{1, \frac{P(y_{\text{new}})}{P(y_{\text{old}})}\frac{\Delta y^{j_{\text{new}}}}{\Delta y^{j_{\text{old}}}}\right\}. \tag{8}$$

p_{BMA} in (8) differs from p_{Met} in (3) by the *bias* $\Delta y^{j_{\text{new}}}/\Delta y^{j_{\text{old}}}$.

The scheme outlined in (6)–(8) satisfies the same balance or detailed balance conditions (defined, e.g., in Ref. [5]) as the original Metropolis algorithm. *The bias influences only the acceptance rate.* Choosing, for example, equidistant partitioning for y ($\Delta y^j = \Delta y^k$ for any j, k) would turn the bias into 1 and get us back to the original Metropolis algorithm.

So far the partitioning y^j has not been introduced explicitly. A particular choice that achieves *equidistant partitioning on the CDF axis* is:

$$\frac{j}{n} = F(y^j) \quad \text{or} \quad y^j = F^{-1}\left(\frac{j}{n}\right). \tag{9}$$

Let us pick a bin initially labeled j and take the limit $n \to \infty$ so that this bin collapses into a point labeled z. This corresponds to the limit:

$$n \to \infty,\ j \to \infty \quad \text{so that} \quad \frac{j}{n} \to z. \tag{10}$$

Also, as the CDF axis is partitioned into n bins of the size $\Delta z = 1/n$, we have $\Delta z \to 0$ for $n \to \infty$. In this limit

$$\frac{\Delta y^j}{\Delta z} = \frac{1}{\Delta z}\left(F^{-1}\left(\frac{j}{n}\right) - F^{-1}\left(\frac{j-1}{n}\right)\right) \to \frac{dF^{-1}(z)}{dz} = \frac{1}{P(y)} \tag{11}$$

holds. Then the probability of the accept/reject step (8) is

$$\frac{P(y_{\text{new}})}{P(y_{\text{old}})}\frac{\Delta y^{j_{\text{new}}}}{\Delta y^{j_{\text{old}}}} = \frac{P(y_{\text{new}})}{P(y_{\text{old}})}\frac{\Delta y^{j_{\text{new}}}/\Delta z}{\Delta y^{j_{\text{old}}}/\Delta z} \to \frac{P(y_{\text{new}})}{P(y_{\text{old}})}\frac{1/P(y_{\text{new}})}{1/P(y_{\text{old}})} = 1. \tag{12}$$

So, in the limit of an infinitely small discretization step this BMA approaches the HBA and the acceptance rate converges to 1. Therefore we call a BMA with a partitioning similar to (9) *Biased Metropolis–heatbath algorithm* (BMHA).

2. Application to lattice gauge theories

The fundamental interactions of Nature known nowadays are the gravitational, electromagnetic, weak and strong interactions. The last three are *gauge field theories*. For example, the Lagrangian of electrodynamics is invariant under local *gauge transformations* that belong to the $U(1)$ gauge group.

Description on the quantum level requires switching from the *classical* to the *quantum* point of view: all fields in the Lagrangian of the theory are promoted from functions to *operators* satisfying certain (anti)commutation relations. Then a physical observable of interest is evaluated as an action of some operator on the *vacuum state* of the theory. Along these lines observables can be represented as *path integrals*, i.e., integrals over all possible values of the fields that live on a four-dimensional space-time. These integrals can be evaluated using *perturbation theory* when they can be expanded in series of parameters that are "small" enough to ensure convergence. Quantum Electrodynamics (QED) provides a good example of a theory where many physical observables are calculated order by order in perturbation theory and match experiments with high accuracy. For instance the magnetic moment of the electron is known to seven significant digits [25].

The theory of strong interaction is *Quantum Chromodynamics* (QCD). The strong force is responsible for binding fundamental constituents of matter, *quarks*, into protons, neutrons and other particles observed experimentally, and, in turn, protons and neutrons into atomic nuclei. The gauge group of QCD is $SU(3)$. As this group is *non-Abelian* the theory possesses a richer structure and introduces more difficulties than QED. One of them is a *non-perturbative regime* where, as the name implies the theory cannot be expanded into a series. To overcome this difficulty *lattice gauge theory* was introduced by Wilson [28]. In principle QCD allows for calculations of low energy properties, as for instance the mass of the proton, by Markov chain Monte Carlo simulations, which are suitable for calculating path

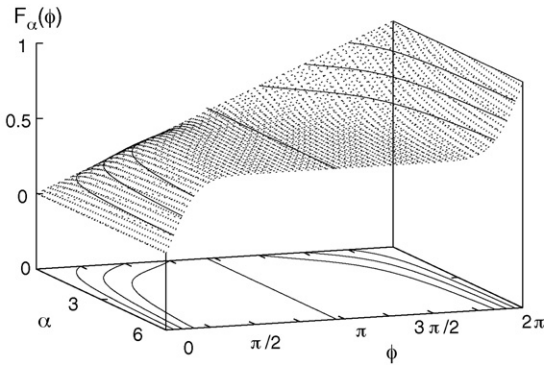

Fig. 1. Cumulative distribution function $F_\alpha(\phi)$ with the level map in the α– ϕ plane.

integrals in Euclidean space (connected by a Wick rotation to the physical Minkowski space). However, in practice such calculation require tremendous computational resources, so that (besides Moore's law at work) major progress on the algorithmic front has still to be made before ultimate answers may be computed.

In the following we illustrate lattice gauge theory calculations on a simple example: The $U(1)$ gauge group. In Ref. [2] we have applied the same method to the $SU(2)$ gauge group and it can also be extended to $SU(3)$. We should emphasize that this deals only with the pure gauge part of the theory, whereas the notorious difficulties of including fermions in these calculations remain at the moment untouched by biased Metropolis calculations.

2.1. $U(1)$ pure gauge theory

For the $U(1)$ gauge group the "matrices" are complex numbers on the unit circle, which can be parameterized by an angle $\phi \in [0, 2\pi)$. After defining the theory on the links of a four-dimensional lattice the PDF

$$P_\alpha(\phi) = N_\alpha e^{\alpha \cos(\phi)} \tag{13}$$

has to be sampled, where α is a parameter associated to the interaction of the link being updated with its environment. The corresponding CDF is

$$F_\alpha(\phi) = N_\alpha \int_0^\phi d\phi' \, e^{\alpha \cos(\phi')} \quad \text{with} \quad F_\alpha(2\pi) = 1. \tag{14}$$

For $U(1)$ HBAs of type (2) were introduced in Refs. [27,20]. As $F_\alpha^{-1}(z)$ is approximated one needs a *repeat until accepted (RUA)* step to generate the correct distribution, although the acceptance rate is always 1.[1] $F_\alpha(\phi)$ depends on the parameter α, which incorporates the effect of interaction with the neighbors, and is a function of ϕ, the variable being updated. In the following we consider $U(1)$ gauge theory at a coupling close to the critical point for which one finds $0 \le \alpha \le 6$. For this case $F_\alpha(\phi)$ is plotted on Fig. 1. Contour lines on the surface represent levels where $F_\alpha(\phi)$ increases from 0 to 1 by a chosen constant value (in this case 1/8). Lines in the α– ϕ plane are projections of these contours and constitute a *level map* similar to those used to encode height on maps in geography. To construct a BMHA we need a discretized version of this level map.

Let us discretize the parameter α into $m = 2^{n_1} = 16$ ($n_1 = 4$) bins. For simplicity we choose equidistant partitioning. Other discretizations are possible too. Then in each α^i bin we discretize ϕ using the condition (9) with $n = 2^{n_2} = 16$ ($n_2 = 4$). In this way we achieve a discretized version of the level map at the bottom of Fig. 1, which is shown in Fig. 2.

Two two-dimensional arrays are needed: one for storing $\phi^{i,j}$ (levels themselves) and another for $\Delta\phi^{i,j} = \phi^{i,j} - \phi^{i,j-1}$ (distances between levels). Let us assume that for a link being updated α falls into the 11th bin, so $i = 11$. Finding i is achieved with an operation of the form: Int$[m\,\alpha/\alpha_{\max}]$ with $\alpha_{\max} = 6$. For a given α^i it is straightforward to apply BMA step (6).

[1] In some of the literature the quantity 1/(average number of RUA heat bath iterations per update) is also called acceptance rate. It should not be confused with the acceptance rate defined here.

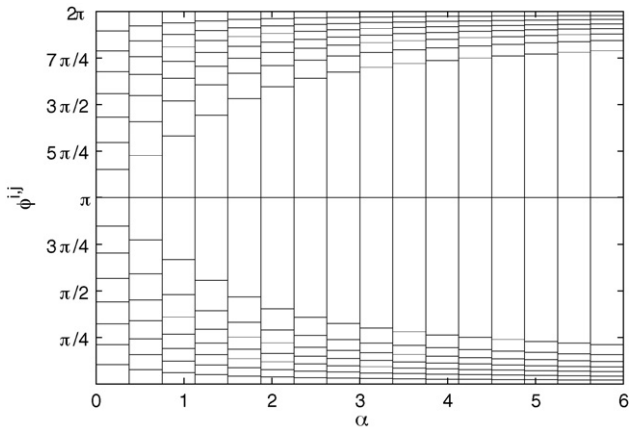

Fig. 2. $m \times n$ partitioning of $\Delta\phi^{i,j}$ for $U(1)$ at the coupling constant value discussed in the text.

The cross-section of the $F_\alpha(\phi)$ surface by the $\alpha = \alpha^{11}$ plane is shown in Fig. 3. To determine the bin label j_{old} which belongs to the (known) value ϕ_{old} (BMA step (7)) one may use the n_2-step recursion $j \to j + 2^{i_2}\text{sign}(\phi - \phi^{i,j})$, $i_2 \to i_2 - 1$. Once j_{old} is known it gives the length of the bin: $\Delta\phi^{i,j_{\text{old}}}$ and the final accept/reject step (8) can be applied:

$$p_{\text{BMA}} = \min\left\{1, \frac{\exp(\alpha\,\cos\,\phi_{\text{new}})}{\exp(\alpha\,\cos\,\phi_{\text{old}})}\frac{\Delta\phi_0^{i,j_{\text{new}}}}{\Delta\phi_0^{i,j_{\text{old}}}}\right\}. \tag{15}$$

2.1.1. Performance

In our simulations we used a finer discretization than in the figures, $m = 32$ and $n = 128$. Table 1 illustrates the performance of the $U(1)$ BMHA for a long run on a 4×16^3 lattice. At the used coupling the system exhibits critical slowing down, because of its proximity to the $U(1)$ phase transition. We used 16,384 sweeps for reaching equilibrium and, subsequently, $32 \times 20{,}480$ sweeps for measurements. Simulations were performed on 2 GHz Athlon PCs with the -O2 option of the (freely available) g77 Fortran compiler.

Our comparison is with the Hattori–Nakajima HBA [20] and with the conventional Metropolis algorithm [21]. A direct measure for the performance of an algorithm is the integrated autocorrelation time τ_{int}. Values of τ_{int} are given in the Table 1 for the Wilson plaquette, $\langle \cos\phi_\square \rangle$ (a reference physical observable whose expectation value we use to check consistency of the algorithms). Error bars are given in parenthesis and apply to the last

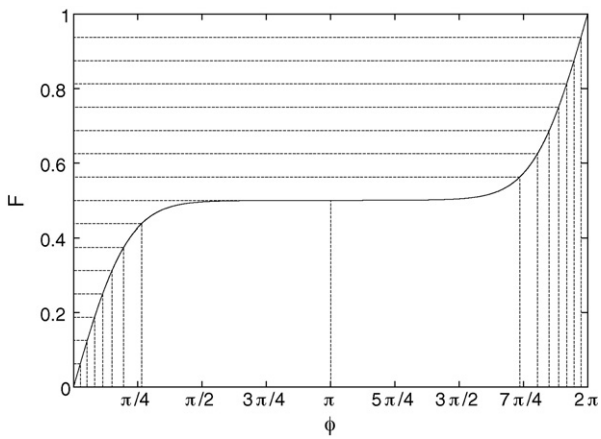

Fig. 3. Discretization of the cumulative distribution function $F_{\alpha^{11}}(\phi)$ for $U(1)$ corresponding to the 11th bin of Fig. 2.

Table 1
Efficiency of three algorithms for $U(1)$ lattice gauge theory on a 4×16^3 lattice at a coupling constant value close to the phase transition value.

$U(1)$	HBA [20]	Metropolis [21]	BMHA
CPU time [s]	131,111	84,951	107,985
Acceptance rate	1 (1.093 proposals)	0.286	0.972
$\langle \cos \phi_\square \rangle$	0.59113 (8)	0.59103 (16)	0.59106 (12)
τ_{int}	127 (7)	341 (26)	142 (10)

digits. They are calculated with respect to 32 bins (jackknife bins in case of τ_{int}) using the data analysis software of [5].

In this example the CDF is known. We have shown that sampling with the BMHA is essentially equivalent to using the HBA, but can be numerically faster, as shown here for $U(1)$. $SU(2)$ lattice gauge theory with the fundamental-adjoint action is a case for which more substantial gains are achieved by using a BMHA [3]. In the next part of the article we show how a similar biasing procedure can be used when the CDF is not known (making a HBA impossible) and how it can be extended to a multi-variable case.

3. Application to biophysics

Simulations of biomolecules remain one of the major challenges in computational science today. Rugged free energy landscapes are typical for such systems and conventional Metropolis updating suffers from low acceptance rates at the temperatures of interest.

We consider biomolecule models for which the energy E is a function of a number of dynamical variables $v_i, i = 1, \ldots, n$. The fluctuations in the Gibbs canonical ensemble are described by a probability density function $\rho(v_1, \ldots, v_n; T) = \text{const} \exp(-\beta E(v_1, \ldots, v_n))$, where T is the temperature, $\beta = 1/(kT)$, and E is the energy of the system. To be consistent with the notation of [4,9] we now use $\rho(v_1, \ldots, v_n; T)$ instead of $P(y)$ introduced in previous one-variable example. Proposing a new variable (with the other variables fixed) from the PDF constitutes a HBA. However, an implementation of a HBA is only possible when the CDF of the PDF can be controlled. In particular this requires the normalization constant in front of the $\exp(-\beta E(v_1, \ldots, v_n))$ Boltzmann factor. In practice this is often not the case. Then the following strategy provides a useful approximation.

For a range of temperatures

$$T_1 > T_2 > \ldots > T_r > \ldots > T_{f-1} > T_f \tag{16}$$

the simulation at the highest temperature, T_1, is performed with the usual Metropolis algorithm and the results are used to construct an *estimator*

$$\bar{\rho}(v_1, \ldots, v_n; T_1)$$

which is used to bias the simulation at T_2. Recursively, the estimated PDF

$$\bar{\rho}(v_1, \ldots, v_n; T_{r-1})$$

is expected to be a useful approximation of $\rho(v_1, \ldots, v_n; T_r)$. Formally this means that BMA acceptance step (8) at temperature T_r is of the form

$$P_{\text{RM}} = \min \left\{ 1, \frac{\exp\left(-\beta E'\right)}{\exp\left(-\beta E\right)} \frac{\bar{\rho}(v_1, \ldots, v_n; T_{r-1})}{\bar{\rho}(v'_1, \ldots, v'_n; T_{r-1})} \right\} \tag{17}$$

where $\beta = 1/(kT)$. For this type of BMA where the bias is constructed by using information from a higher temperature the name *Rugged Metropolis* (RM) was given in Ref. [4].

For the following illustration we use the all-atom energy function Empirical Conformational Energy Program for Peptides/2 (ECEPPs) [24] (and references given therein) as implemented in the Simple Molecular Mechanics for Proteins (SMMPs) [15] program package. Our dynamical variables v_i are the dihedral angles, each chosen to be in the range $-\pi \leq v_i < \pi$, so that the volume of the configuration space is $K = (2\pi)^n$. Details of the energy functions are

expected to be irrelevant for the algorithmic questions addressed here. Our test case is the small brain peptide Met-Enkephalin (Tyr-Gly-Gly-Phe-Met), which features 24 dihedral angels as dynamical variables (we use the conventions of Ref. [15]). Besides the ϕ, ψ angles, we keep also the ω angles unconstrained, which are usually restricted to $[\pi - \pi/9, \pi + \pi/9]$. This allows us to illustrate the RM idea for a particularly simple case.

3.1. The RM_1 approximation

To get things started, we need to construct an estimator $\bar{\rho}(v_1, \ldots, v_n; T_r)$ from the numerical data of the RM simulation at temperature T_r. Although this is neither simple nor straightforward, a variety of approaches offer themselves to define and refine the desired estimators.

In Ref. [4] the approximation

$$\bar{\rho}(v_1, \ldots, v_n; T_r) = \prod_{i=1}^{n} \bar{\rho}_i^1(v_i; T_r) \tag{18}$$

was investigated, where $\bar{\rho}_i^1(v_i; T_r)$ are estimators of reduced one-variable PDFs defined by

$$\rho_i^1(v_i; T) = \int_{-\pi}^{+\pi} \prod_{j \neq i} d\, v_j \rho(v_1, \ldots, v_n; T). \tag{19}$$

The resulting algorithm, called RM_1, constitutes the simplest RM scheme possible.

The cumulative distribution functions are defined by

$$F_i(v) = \int_{-\pi}^{v} \mathrm{d}v' \rho_i^1(v'). \tag{20}$$

The estimate of F_{10}, the cumulative distribution function for the dihedral angle Gly-3 ϕ (v_{10}), from the vacuum simulations at our highest temperature, $T_1 = 400\,\mathrm{K}$, is shown in Fig. 4. For our plots in this part of the paper we use degrees, while we use radians in our theoretical discussions and in the computer programs. Fig. 4 is obtained by sorting all n_{dat} values of v_{10} in our time series in ascending order and increasing the values of F_{10} by $1/n_{\mathrm{dat}}$ whenever a measured value of v_{10} is encountered. Using a heapsort approach, the sorting is done in $n_{\mathrm{dat}} \log_2(n_{\mathrm{dat}})$ steps (see, e.g., Ref. [5]).

Fig. 5 shows the cumulative distribution function for v_9 (Gly-2 ω) at 400 K, which is the angle of lowest acceptance rate in the conventional Metropolis updating. This distribution function corresponds to a histogram narrowly peaked around $\pm\pi$, which is explained by the specific electronic hybridization of the CO–N peptide bond. From the grid shown in Fig. 5 it is seen that the RM_1 updating concentrates the proposal for this angle in the range slightly above $-\pi$ and

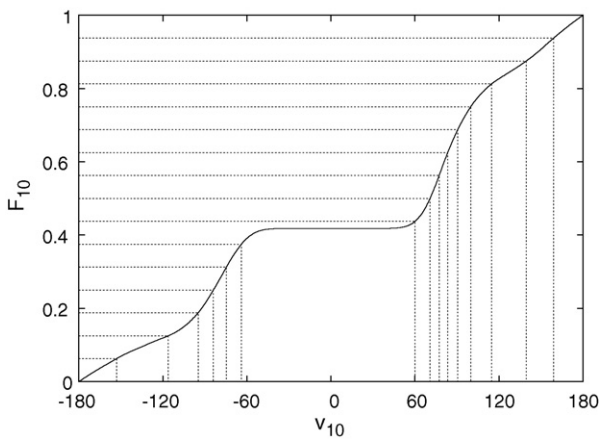

Fig. 4. Estimate of the cumulative distribution function for the Met-Enkephalin dihedral angle v_{10} (Gly-3 ϕ) at 400 K.

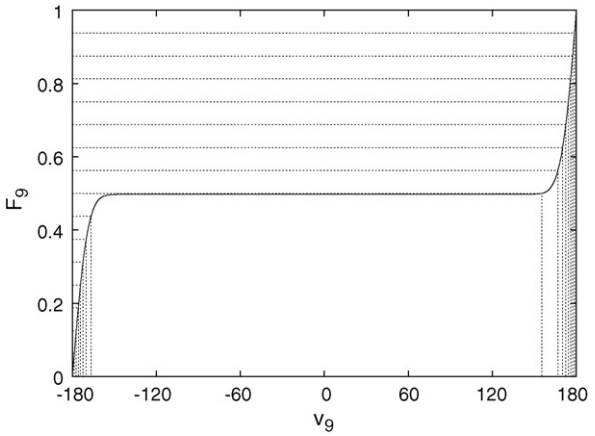

Fig. 5. Estimate of the cumulative distribution function for the Met-Enkephalin dihedral angle v_9 (Gly-2 ω) at 400 K.

slightly below $+\pi$. Thus the procedure has a similar effect as the often used restriction to the range $[\pi - \pi/9, \pi + \pi/9]$, which is also the default implementation in SMMP.

After the empirical CDFs are constructed for each angle v_i, they are discretized using the condition (9). Here we denote differences (5) needed for the bias as

$$\Delta v_{i,j} = v_{i,j} - v_{i,j-1} \quad \text{with} \quad v_{i,0} = -\pi. \tag{21}$$

The RM_1 updating of each dihedral angle v_i follows the BMA procedure (6)–(8). The accept/reject step in the $v_{i,j}$ notation is

$$p_{\mathrm{RM}_1} = \min\left\{1, \frac{\exp(-\beta E')}{\exp(-\beta E)} \frac{\Delta v_{i,j_{\mathrm{new}}}}{\Delta v_{i,j_{\mathrm{old}}}}\right\}. \tag{22}$$

3.2. The RM_2 approximation

In Ref. [9] the RM_1 scheme of Eq. (22) was generalized to the *simultaneous* updating of two dihedral angles. For $i_1 \neq i_2$ the reduced two-variable PDFs are defined by

$$\rho^2_{i_1,i_2}(v_{i_1}, v_{i_2}; T) = \int_{-\pi}^{+\pi} \prod_{j \neq i_1,i_2} d\, v_j \rho(v_j, \ldots, v_n; T). \tag{23}$$

The one-variable cumulative distribution functions F_{i_1} and the discretization $v_{i_1,j}$, $j = 0, \ldots, n$ are already given by Eqs. (20) and (21). We define conditional CDFs by

$$F_{i_1,i_2;j}(v) = \int_{-\pi}^{v} dv_{i_2} \int_{v_{i_1,j-1}}^{v_{i_1,j}} dv_{i_1}\, \rho^2_{i_1,i_2}(v_{i_1}, v_{i_2}) \tag{24}$$

for which the normalization $F_{i_1,i_2;j}(\pi) = 1/n$ holds. To extend the RM_1 updating to two variables we define for each integer $k = 1, \ldots, n$ the value $F_{i_1,i_2;j,k} = k/n^2$. Next we define $v_{i_1,i_2;j,k}$ through $F_{i_1,i_2;j,k} = F_{i_1,i_2;j}(v_{i_1,i_2;j,k})$ and also the differences

$$\Delta v_{i_1,i_2;j,k} = v_{i_1,i_2;j,k} - v_{i_1,i_2;j,k-1} \quad \text{with} \quad v_{i_1,i_2;j,0} = -\pi. \tag{25}$$

The RM_2 procedure for the simultaneous update of (v_{i_1}, v_{i_2}) is then specified as follows:

- Propose a new value $v_{i_1,\mathrm{new}}$ using two uniform random numbers r_1, r_2 (BMA step (6) for the angle i_1):

$$j_{\mathrm{new}} = 1 + \mathrm{Int}[n\, r_1] \quad \text{and} \quad v_{i_1,\mathrm{new}} = v_{i_1,j_{\mathrm{new}}-1} + \Delta v_{i_1,j_{\mathrm{new}}}\, r_2. \tag{26}$$

- Propose a new value $v_{i_2,\mathrm{new}}$ using two uniform random numbers r_3, r_4 (BMA step (6) for the angle i_2):

$$k_{\mathrm{new}} = 1 + \mathrm{Int}[n\, r_3] \quad \text{and} \quad v_{i_2,\mathrm{new}} = v_{i_1,i_2;j_{\mathrm{new}},k_{\mathrm{new}}-1} + \Delta v_{i_1,i_2;j_{\mathrm{new}},k_{\mathrm{new}}}\, r_4. \tag{27}$$

- Find the bin index j_{old} for the present angle $v_{i_1,\mathrm{old}}$ through $v_{i_1,j_{\mathrm{old}}-1} \le v_{i_1,\mathrm{old}} \le v_{i_1,j_{\mathrm{old}}}$, just like for RM_1 updating (BMA step (7) for v_{i_1}).
- Find the bin index k_{old} for the present angle $v_{i_2,\mathrm{old}}$ through $v_{i_1,i_2;j_{\mathrm{old}},k_{\mathrm{old}}-1} \le v_{i_2,\mathrm{old}} \le v_{i_1,i_2;j_{\mathrm{old}},k_{\mathrm{old}}}$ (again step (7) but for v_{i_2}).
- Accept $(v_{i_1,\mathrm{new}}, v_{i_2,\mathrm{new}})$ with the probability

$$p_{\mathrm{RM}_2} = \min\left\{1, \frac{\exp(-\beta E')}{\exp(-\beta E)} \frac{\Delta v_{i_1,j_{\mathrm{new}}}}{\Delta v_{i_1,j_{\mathrm{old}}}} \frac{\Delta v_{i_1,i_2;j_{\mathrm{new}},k_{\mathrm{new}}}}{\Delta v_{i_1,i_2;j_{\mathrm{old}},k_{\mathrm{old}}}}\right\}. \tag{28}$$

As for RM_1, estimates of the conditional CDFs and the intervals $\Delta v_{i_1,i_2;j,k}$ are obtained from the conventional Metropolis simulation at 400 K. In the following we focus on the pairs (v_7, v_8), (v_{10}, v_{11}) and (v_{15}, v_{16}). These angles correspond to the largest integrated autocorrelation times of the RM_1 procedure and are expected to be strongly correlated with one another because they are pairs of dihedral angles around a C_α atom.

The bias of the acceptance probability given in Eq. (28) is governed by the areas

$$\Delta A_{i_1,i_2;j,k} = \Delta v_{i_1,j}\, \Delta v_{i_1,i_2;j,k}.$$

For $i_1 = 7$ and $i_2 = 8$ our 400 K estimates of these areas are depicted in Fig. 6. For the RM_2 procedure these areas take the role which the intervals on the abscissa of Fig. 4 play for RM_1 updating. The small and the large areas are proposed with equal probabilities, so the *a priori* probability for our two angles is high in a small area and low in a large area. In Fig. 6 the largest area is 503.4 times the smallest area. Areas of high probability correspond to allowed regions in the Ramachandran map of a Gly residue [23].

Note that the order of the angles matters. The difference between Figs. 6 and 7 is that we plot in Fig. 6 the areas $A_{7,8;j,k}$ and in Fig. 7 the areas $A_{8,7;j,k}$ while the labeling of the axes is identical. This means that for Fig. 6 sorting is first done on the angle v_7 (regardless of the value of v_8) and then done on v_8 for which the corresponding value of v_7 is within a particular bin Δv_7, but for Fig. 7 it is first done one v_8 and then on v_7. In Fig. 7 the largest area is 396.4 times the smallest area.

Figs. 8 and 9 give plots for the (v_{10}, v_{11}) and (v_{15}, v_{16}) pairs in which the angle with the smaller subscript is sorted first. The ratio of the largest area over the smallest area is 650.9 for (v_{10}, v_{11}) and 2565.8 for (v_{15}, v_{16}). The large number in the latter case is related to the fact that (v_{15}, v_{16}) is the pair of ϕ, ψ angles around the C_α atom of Phe-4, for which positive ϕ values are disallowed [23].

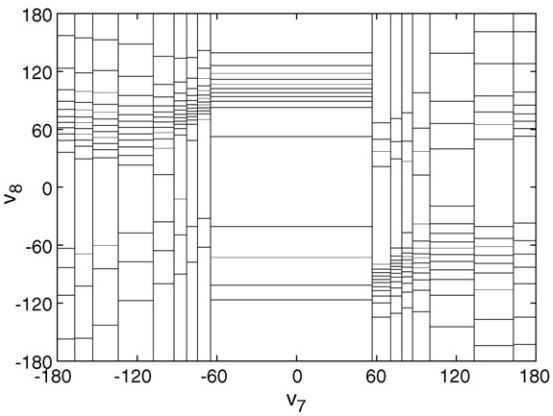

Fig. 6. Areas of equal probabilities (sorting v_7 then v_8).

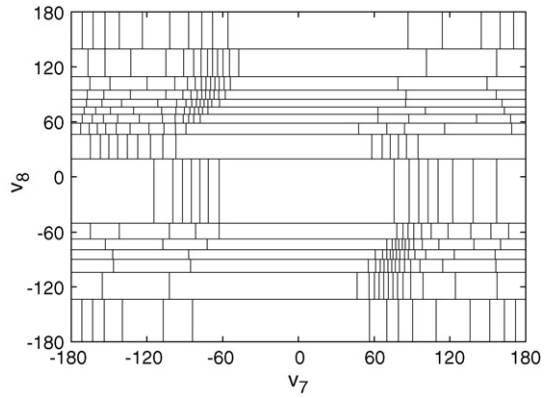

Fig. 7. Areas of equal probabilities (sorting v_8 then v_7).

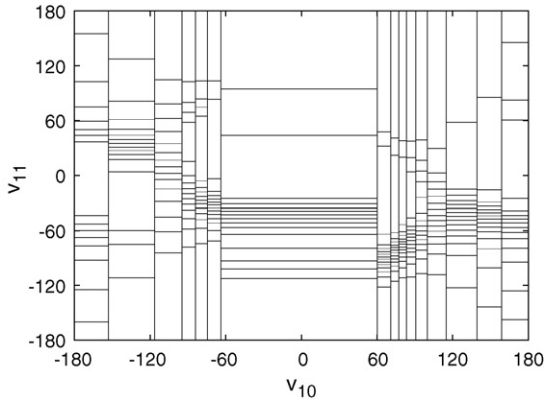

Fig. 8. Areas of equal probabilities (sorting v_{10} then v_{11}).

3.3. Performance

The RM_2 scheme which we have tested adds updates for the three pairs (v_7, v_8), (v_{10}, v_{11}) and (v_{15}, v_{16}) after one-angle updates for all the 24 angles with the RM_1 scheme. For each pair both orders of sorting are used, so that we add altogether six new updates.

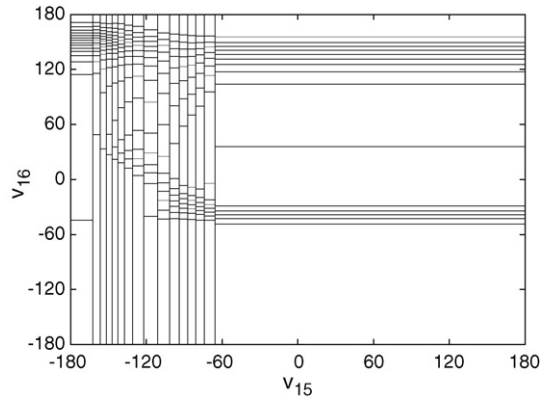

Fig. 9. Areas of equal probabilities (sorting v_{15} then v_{16}).

Table 2
Integrated autocorrelation times for dihedral angle movements in units of 32 sweeps for Metropolis and RM_1 and in units of 26 sweeps for RM_2.

var	400 K (Metro)	300 K (Metro)	300 K (RM_1)	300 K (RM_2)
v_7	5.83 (29)	103 (14)	52.9 (4.3)	24.3 (1.3)
v_8	7.36 (22)	125 (12)	74.2 (6.9)	35.0 (2.7)
v_9	4.39 (13)	32.0 (2.2)	14.2 (1.0)	8.84 (48)
v_{10}	9.08 (88)	124 (12)	80.6 (6.9)	34.3 (2.8)
v_{11}	5.39 (45)	105 (08)	72.4 (5.5)	31.3 (1.9)
v_{15}	6.72 (28)	105 (12)	45.6 (2.7)	27.5 (4.5)
v_{16}	9.28 (28)	133 (09)	75.2 (5.2)	33.9 (2.1)
E	4.89 (21)	50.7 (5.0)	26.0 (1.4)	14.2 (0.7)

For the angles used in the figures the performance of the RM_1 and RM_2 schemes is illustrated in Table 2. Integrated autocorrelation times (computed along the lines of [5]) are compiled. The units are chosen, so that the computer time needed with the different algorithms to achieve the same accuracy is directly proportional to the integrated autocorrelation times of the table. At 300 K we read off that the improvement over the conventional Metropolis algorithm is typically a factor of two for the RM_1 and a factor of four for the RM_2 approach. It stays about the same at lower temperatures [9].

4. Conclusions

High energy physics and biophysics are certainly far apart in their scientific objectives. Nevertheless quite similar computational techniques allow for efficient Metropolis simulations in either field. Cross-fertilization may go in both directions. For instance, generalized ensemble techniques propagated from lattice gauge theory [8] over statistical physics [7] into biophysics [18]. It appears that biased Metropolis techniques propagate in the opposite direction. It remains to be seen whether they will indeed gain widespread acceptance.

Acknowledgments

We would like to thank Michael Mascagni for organizing a most enjoyable workshop. A. Bazavov and B.A. Berg were in part supported by the Department of Energy under contract DE-FG02-97ER41022. H.-X. Zhou was supported in part by the National Institutes of Health grant GM 58187.

References

[1] S. Adler, Overrelaxation algorithms for lattice field theories, Phys. Rev. D 37 (1988) 458–471.
[2] A. Bazavov, B. Berg, Heat bath efficiency with a Metropolis-type updating, Phys. Rev. D 71 (2005) 114506.
[3] A. Bazavov, B. Berg, U. Heller, Biased Metropolis–heat-bath algorithm for fundamental-adjoint SU(2) lattice gauge theory, Phys. Rev. D 72 (2005) 117501.
[4] B. Berg, Metropolis importance sampling for rugged dynamical variables, Phys. Rev. Lett. 90 (2003) 180601.
[5] B. Berg, Markov Chain Monte Carlo Simulations and Their Statistical Analysis, World Scientific, Singapore, 2004.
[6] B. Berg, A. Bazavov, Non-perturbative U(1) gauge theory at finite temperature, Phys. Rev. D 74 (2006) 094502.
[7] B. Berg, T. Celik, New approach to spin-glass simulations, Phys. Rev. Lett. 69 (1992) 2292–2295.
[8] B. Berg, T. Neuhaus, Multicanonical algorithms for first order phase transitions, Phys. Lett. B 267 (1991) 249–253.
[9] B. Berg, H.-X. Zhou, Rugged Metropolis sampling with simultaneous updating of two dynamical variables, Phys. Rev. E 72 (2005) 016712.
[10] F. Brown, T. Woch, Overrelaxed heat-bath and Metropolis algorithms for accelerating pure gauge Monte Carlo calculations, Phys. Rev. Lett. 58 (1987) 2394–2396.
[11] A. Bruce, Universality in the two-dimensional continuous spin model, J. Phys. A 18 (1985) L873–L877.
[12] M. Creutz, Monte Carlo study of quantized SU(2) gauge theory, Phys. Rev. D 21 (1980) 2308–2315.
[13] M. Creutz, Overrelaxation and Monte Carlo simulation, Phys. Rev. D 36 (1987) 515–519.
[14] M. Deem, J. Bader, A configurational bias Monte Carlo method for linear and cyclic peptides, Mol. Phys. 87 (1996) 1245–1260.
[15] F. Eisenmenger, U. Hansmann, S. Hayryan, C.-K. Hu, A modern package for simulation of proteins (SMMP), Comp. Phys. Commun. 138 (2001) 192–212.
[16] G. Favrin, A. Irbäck, F. Sjunnesson, Monte Carlo update for chain molecules: Biased Gaussian steps in torsional space, J. Chem. Phys. 114 (2001) 8154–8158.

[17] J. Gubernatis (ed.), The Monte Carlo method in the physical sciences: celebrating the 50th anniversary of the Metropolis Algorithm, AIP Conference Proceedings, vol. 690, Melville, New York (2003).
[18] U. Hansmann, Y. Okamoto, Prediction of peptide conformation by multicanonical algorithm: new approach to the multiple-minima problem, J. Comp. Chem. 14 (1993) 1333–1338.
[19] W. Hastings, Monte Carlo sampling methods using Markov chains and their applications, Biometrica 57 (1970) 97–109.
[20] T. Hattori, H. Nakajima, Improvement of efficiency in generating random U(1) variables with Boltzmann distribution in Monte Carlo calculations, Nucl. Phys. B (Proc. Suppl.) 26 (1992) 635–637.
[21] N. Metropolis, A. Rosenbluth, N. Rosenbluth, A. Teller, E. Teller, Equation of state calculations by fast computing machines, J. Chem. Phys. 21 (1953) 1087–1092.
[22] A. Milchev, D. Heermann, K. Binder, Finite-size scaling analysis of the ϕ^4 field theory on the square lattice, J. Stat. Phys. 44 (1986) 749–784.
[23] G. Schultz, R. Schirmer, Principle of Protein Structure, Springer, New York, 1979.
[24] M. Sippl, G. Nemethy, H. Scheraga, Intermolecular potential from crystal data. 6. Determination of empirical potentials for O–H $\cdots$O= C hydrogen bonds from packing configurations, J. Phys. Chem. 88 (1984) 6231–6233.
[25] The National Institute of Standards and Technology, The NIST reference on constants, units, and uncertainty. http://physics.nist.gov/cuu/Constants/.
[26] J. Ulmschneider, W. Jorgensen, Monte Carlo backbone sampling for polypeptides with variable bond angles and dihedral angles using concerted rotations and a gaussian bias, J. Chem. Phys. (2003) 4261–4271.
[27] R. Wensley, Monopoles and U(1) lattice gauge theory, PhD dissertation, University of Illinois at Urbana-Champaign, Department of Physics, ILL-TH-89–25 (1989).
[28] K. Wilson, Confinement of quarks, Phys. Rev. D 10 (1974) 2445–2459.