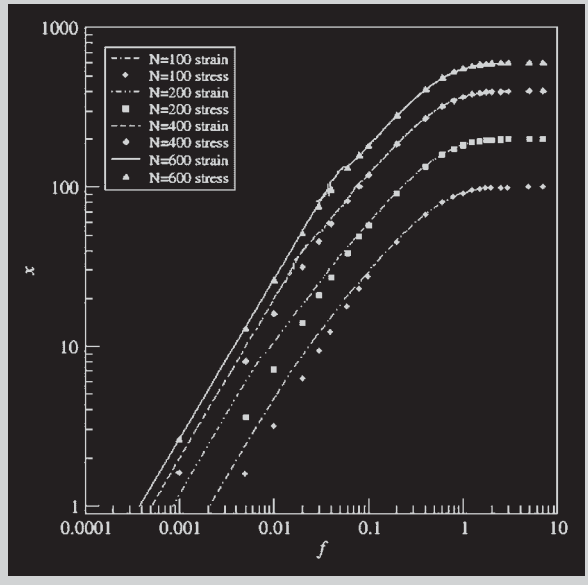
**Summary:** We describe an approach to use multiple-histogram methods in combination with static, biased Monte Carlo simulations. To illustrate this, we computed the force-extension curve of an athermal polymer from multiple histograms constructed in a series of static Rosenbluth Monte Carlo simulations. From the complete histogram of the distribution function of the end-to-end vectors of the polymer chain, we can efficiently compute the polymer force-extension curve.




Comparison of the stress-strain curves for the stress ensemble (symbols) and the strain ensemble (lines). Results obtained for $N = 100$, 200, 400, and 600. For small $x$, $f(x) = -F'(x)$ was computed by aproximating $F(x)$ by a second degree polynomial and then taking the derivative. For large $x$, $f(x) = -F'(x)$ was computed numerically.

# Multiple Histogram Method and Static Monte Carlo Sampling

*Márcia A. Inda,*[a] *Daan Frenkel**

FOM Institute for Atomic and Molecular Physics, Kruislaan 407, 1098 SJ, Amsterdam, The Netherlands
Fax: 00+31 20 6684106; E-mail: Frenkel@amolf.nl

Received: May 13, 2003; Revised: August 22, 2003; Accepted: September 8, 2003; DOI: 10.1002/mats.200350040

**Keywords:** Monte Carlo simulation; multiple histogram method; polyethylene (PE); stress/strain; single chain

## Introduction

There are two distinct Monte Carlo (MC) methods to simulate the behavior of individual chain molecules: one is the so-called ''static'' approach in which every MC step involves the generation of a chain conformation from scratch. To achieve proper Boltzmann sampling, different chain conformations are given a different weight. The second, ''dynamic'' scheme, generates a Markov chain of states such that every state is visited with a probability proportional to its Boltzmann weight. Consequently, every state that is generated in a dynamic MC scheme, carries the same weight. For the study of many-body systems, dynamic MC sampling is usually the method of choice. However, in certain cases, e.g., in the generation of conformations of a single polymer or model protein, static MC may be more efficient. Sometimes the aim of a simulation is to obtain a histogram that measures the probability distribution of all possible states of the system, as a function of some order parameter. In such cases, use of the dynamic MC approach is straightforward, but the use of static MC schemes is more subtle. The aim of the present paper is to illustrate how multi-histogram methods can be fruitfully combined with biased, static MC sampling. As an example, we consider the computation of the force-extension curve of a simple model polymer. Although the specific example that we consider has been selected because of its simplicity, the general problem of computing force-extension curves is of considerable current interest, as more experimental results on the stretching of synthetic polymers[1–4] and

[a] Current address: Department of Mathematics, Utrecht University, Po Box 80010, 3508 TA, Utrecht, The Netherlands; E-mail: Inda@math.uu.nl.

bio-macromolecules (RNA, DNA,[5–9] and proteins[8,10–15]) become available.

Two basic strategies exist to study the deformation behavior of a single polymer chain: the *stress ensemble*, in which one measures the elongation of a chain subjected to a stretching force, and the *strain ensemble*, in which one measures the internal retractive force of a chain at fixed elongation. Various Metropolis Monte Carlo[16] approaches have been used to simulate self-avoiding polymer chains using both the stress ensemble[17–23] and the strain ensemble.[21,24–26] These Monte Carlo simulations include athermal chains and chains in good/bad solvents using different polymer models.

While the existing computational schemes are quite adequate to simulate the force-extension curves of simple polymers, histogram methods have the intrinsic advantage that they allow us to obtain much more information than that of the simple stress-strain curve. Of course, given a complete histogram of the probability distribution of polymer end-to-end vectors, we can deduce the force-extension curve. But, in addition, the same (or similar) histograms may provide information about the free-energy landscape of the chain molecule. To compute such a histogram, one usually computes sub-histograms under different conditions and then combines these sub-histograms into the desired total histogram. The *multiple histogram method* (MHM)[27] is a powerful method to construct improved histograms from a set of related (less accurate) histograms. The MHM was devised to combine unbiased data, and it has been used to combine histograms generated by dynamic sampling methods.[28] Grassberger and Hsu[29] have applied a multiple histogram approach to the particular case of histograms obtained by the pruned-enriched Rosenbluth sampling method (PERM).[30] In this paper, we show how to extend the MHM so it can be used with any set of weighted histograms. This extension makes it possible to combine MHM calculations with static sampling techniques, such as the Rosenbluth method[31] and the PERM, in a consistent way. In our example, we use a simple athermal lattice model for a polyethylene-like polymer. By applying the biased MHM to a set of weighted histograms generated using the plain Rosenbluth sampling method, we generate free energy profiles and study the deformation behavior of our model polymer chain in the stress and strain ensembles. We compare our results with the scaling theory of de Gennes[32] and Pincus.[33] Due to the nature of the biased MHM, the results of a single simulation can also be used to predict the force-extension curves at different temperatures. However, we do not explore this possibility in the present paper.

The remainder of this paper is organized as follows. In the Section *Chain Elasticity and Free-Energy Profiles*, we introduce the stress and strain ensembles and we show how histograms can be used to study the elastic properties of chain molecules in both ensembles. In the Section *Multiple Histogram Method for Biased Sampling*, we show how to apply the multiple histogram method to a set of weighted histograms. In the Section *Application*, we describe our case study, an athermal polyethylene chain model. The practical implementation is discussed in the Section *Implementation Issues*. And in the Section *Results and Discussion*, we discuss the results obtained for our athermal polyethylene chain and compare them with the relevant theoretical predictions.

## Chain Elasticity and Free-Energy Profiles

We consider a polymer model for which the potential energy of a single conformation, $\alpha$, is given by $U(\alpha)$. Let $x = x(\alpha)$ be the $x$ component of the end-to-end vector of a chain configuration. (From here onwards, we use $x$ to denote an specific value and $x(\alpha)$ to denote the function.) The histogram of the distribution function of $x$ is defined as

$$p(x) = \frac{\int \delta[x - x(\alpha)] e^{-\beta U(\alpha)} d\alpha}{\int e^{-\beta U(\alpha)} d\alpha} \tag{1}$$

Here, $\delta$ is the Dirac delta; $\beta = 1/(k_b T)$; $k_b$ is the Boltzmann constant; and $T$ is the absolute temperature. A Landau free energy $F(x)$ as a function of $x$ can be defined as

$$F(x) = -k_b T \ln(p(x)) \tag{2}$$

In the strain ensemble, one wants to measure the retractive force, $f(x)$, acting on the ends of a chain for which the $x$ component of the end-to-end distance is fixed. This force corresponds to the derivative of the Landau free energy, $F(x)$

$$f(x) = -\frac{d}{dx} F(x) = k_b T \frac{d}{dx} \ln(p(x)) \tag{3}$$

In the constant-stress ensemble, the elongation of the chain is allowed to fluctuate, while the applied force $\mathbf{f}$ is kept constant. If a force of magnitude $f$ is applied along the x-direction, then the average elongation is given by

$$\langle x \rangle_f = \frac{\int x(\alpha) e^{-\beta U_f(\alpha)} d\alpha}{\int e^{-\beta U_f(\alpha)} d\alpha} \tag{4}$$

where

$$U_f(\alpha) = U(\alpha) - f\, x(\alpha) \tag{5}$$

is the total potential energy of the system.

As the chain elongation fluctuates, we can define the probability distribution of chain extensions $p_f(x)$:

$$p_f(x) = \frac{1}{Z_f} \int \delta[x - x(\alpha)] e^{-\beta U_f(\alpha)} d\alpha \tag{6}$$

where

$$Z_f = \int e^{-\beta U_f(\alpha)} d\alpha \tag{7}$$

As we expect

$$\int x p_f(x)dx = \int x\left(\frac{1}{Z_f}\int \delta[x-x(\alpha)]e^{-\beta U_f(\alpha)}d\alpha\right)dx$$
$$= \frac{1}{Z_f}\int\left(\int x\delta[x-x(\alpha)]dx\right)e^{-\beta U_f(\alpha)}d\alpha$$
$$= \frac{1}{Z_f}\int x(\alpha)e^{-\beta U_f(\alpha)}d\alpha = \langle x\rangle_f \quad (8)$$

Knowledge of any distribution function $p_i(x)$ corresponding to a force $f_i$ can be used to compute any other histogram $p_j(x)$ corresponding to a force $f_j$:

$$p_j(x) = \frac{1}{Z_j}\int \delta[x-x(\alpha)]e^{-\beta U_j(\alpha)}d\alpha$$
$$= \frac{1}{Z_j}\frac{Z_i}{Z_i}\int \delta[x-x(\alpha)]e^{-\beta U_j(\alpha)}e^{\beta f_i x(\alpha)-\beta f_i x(\alpha)}d\alpha$$
$$= \frac{e^{-\beta(f_i x-f_j x)}Z_i}{Z_j}\int \delta[x-x(\alpha)]\frac{e^{-\beta U_i(\alpha)}}{Z_i}d\alpha$$
$$= e^{-\beta(f_i x-f_j x)}p_i(x)\frac{Z_i}{Z_j} \quad (9)$$

In theory, an accurate estimate of $p_i$ and of the normalization constants $Z_i$ and $Z_j$ can be used to study the deformation behavior of a polymer chain in both the stress and strain ensembles: in the constant-strain ensemble, by computing $f(x)$, cf. Equation (3) and in the constant-stress ensemble, by computing $\langle x\rangle_{fi}$, cf. Equation (8) and (9). In practice, obtaining accurate estimates is not straightforward.

It is not necessary to have estimates of both $Z_i$ and $Z_j$ to compute the ratio $Z_j/Z_i$. Since

$$\frac{Z_j}{Z_i} = \frac{1}{Z_i}\int e^{-\beta(f_i x(\alpha)-f_j x(\alpha))}e^{-\beta U_i(\alpha)}d\alpha$$
$$= \langle e^{-\beta(f_i x(\alpha)-f_j x(\alpha))}\rangle_i \quad (10)$$

we can use an estimate of $\left\langle e^{-\beta(f_i x(\alpha)-f_j x(\alpha))}\right\rangle_{f_i}$ instead.

## Multiple Histogram Method for Biased Sampling

### *Multiple Histogram Method*

Suppose that we have obtained the estimates $H_i(x)$ and $\hat{Z}_i$ corresponding to the histograms $p_i(x)$ and the normalization constants $Z_i$ of a set of chain systems subjected to the potential energy $U_i$, for $i = 0, 1, \ldots, N$. We want to combine those estimates in order to obtain an improved estimate for a certain $p_j(x)$ with $j \in [0, 1, \ldots, N]$. By Equation (9),

$$p_j(x) = e^{-\beta(f_i x-f_j x)}p_i(x)\frac{Z_i}{Z_j} \approx e^{-\beta(f_i x-f_j x)}H_i(x)\frac{\hat{Z}_i}{\hat{Z}_j} \quad (11)$$

Since in practice Equation (11) gives very poor results, we construct our estimator by making a linear combination of the $p_i$'s:[27]

$$p_j(x) \approx p_j^{est}(x) = \sum_{i=0}^{N} a_i(x)e^{-\beta(f_i x-f_j x)}H_i(x)\frac{\hat{Z}_i}{\hat{Z}_j} \quad (12)$$

Here, the $a_i$'s are weights subjected to the condition that $\sum_i a_i = 1$. As in the unbiased method, we can use the Lagrange multipliers method to choose the $a_i$'s in such a way that the variance, $Var(p_j^{est}(x))$, is minimal. In this case,

$$a_i(x) = \frac{e^{2\beta(f_i x-f_j x)}}{Var\left(H_i(x)\frac{\hat{Z}_i}{\hat{Z}_j}\right)\sum_{k=0}^{N}\frac{e^{2\beta(f_k x-f_j x)}}{Var\left(H_k(x)\frac{\hat{Z}_k}{\hat{Z}_j}\right)}} \quad (13)$$

if we assume that $Var(H_i(x)\frac{\hat{Z}_i}{\hat{Z}_j})$ are independent. From here on, our method differs from the original approach. The histograms obtained by a biased sampling method are weighted histograms, and therefore, we cannot use the usual statistical estimators to estimate $Var(H_i(x)\hat{Z}_i/\hat{Z}_j)$. Instead, we need estimators that account for the bias. But, first of all, we need to derive expressions for the estimates $H_i(x)$ and $\hat{Z}_i$.

Alternatively, if we want to estimate $p_l(x)$ for a force $f_l$, that is not in the set of simulated histograms, we need to compute estimates for $\left\langle e^{-\beta(f_i x(\alpha)-f_i x(\alpha))}\right\rangle_i$, because estimates of $Z_l$ will not be available. Expressions for these estimates can be derived in the same way as for the estimates $H_i(x)$.

### *Estimators of $H_i(x)$ and $\hat{Z}_i$*

A biased sampling method generates the same sample space of the system in study, but with a different density distribution function. To account for this difference, we need to introduce a weighting function, which is proportional to the ratio between the desired probability and the sampling probability.

The problem can be described as follows. Given a set of values $X_{i,1} = x(\alpha_{i,1}), X_{i,2} = x(\alpha_{i,2}), \ldots, X_{i,M_i} = x(\alpha_{i,M_i})$ corresponding to a set of $M_i$ chain configurations $\alpha_{i,1}$, $\alpha_{i,2}, \ldots, \alpha_{i,M_i}$ generated according to a certain biased probability $\sigma_i$ (in our case the Rosenbluth probability), we want to estimate $p_i(x)$ with respect to another probability $\pi_i$ (in our case the Boltzmann probability) defined by

$$\pi_i(\alpha) = \frac{\tilde{\pi}_i(\alpha)}{\int \tilde{\pi}_i(\alpha)d\alpha} = \frac{1}{Z_i}\tilde{\pi}_i(\alpha) \quad (14)$$

where $\tilde{\pi}_i$ is a given unnormalized probability.

Now we define the weights:

$$w_i(\alpha) = c_i \frac{\tilde{\pi}_i(\alpha)}{\sigma_i(\alpha)} = c_i Z_i \frac{\pi_i(\alpha)}{\sigma_i(\alpha)} \tag{15}$$

where $c_i$ is an arbitrary constant.

The expected value of $w_i$ with respect to $\sigma_i$, $E_{\sigma_i}(w_i)$, is

$$E_{\sigma_i}(w_i) = \int c_i Z_i \frac{\pi_i(\alpha)}{\sigma_i(\alpha)} \sigma_i(\alpha) d\alpha = c_i Z_i \tag{16}$$

Consequently, if for each configuration, $\alpha_{i,j}$, we compute the weight $W_{i,j} = w_i(\alpha_{i,j})$, we can estimate $Z_i$ using

$$\hat{Z}_i = \frac{1}{c_i} \overline{W_i} = \frac{1}{c_i M_i} \sum_{k=1}^{M_i} W_{i,k} \tag{17}$$

A histogram, $p_i$, can be estimated from a sample generated according to the probability, $\sigma_i$, as follows. First note that

$$\begin{aligned} p_i(x) &= \frac{\int \delta[x - x(\alpha)]\, \tilde{\pi}_i(\alpha) d\alpha}{\int \tilde{\pi}_i(\alpha) d\alpha} = \frac{\int \delta[x - x(\alpha)] \frac{\tilde{\pi}_i(\alpha)}{\sigma_i(\alpha)} \sigma_i(\alpha) d\alpha}{\int \frac{\tilde{\pi}_i(\alpha)}{\sigma_i(\alpha)} \sigma_i(\alpha) d\alpha} \\ &= \frac{\frac{1}{c_i} \int \delta[x - x(\alpha)] w_i(\alpha) \sigma_i(\alpha) d\alpha}{\frac{1}{c_i} \int w_i(\alpha) \sigma_i(\alpha) d\alpha} \\ &= \frac{E_{\sigma_i}(\delta[x - x(\alpha)] w_i(\alpha))}{E_{\sigma_i}(w_i(\alpha))} \end{aligned} \tag{18}$$

To define an estimator for $E_{\sigma_i}(\delta[x - x(\alpha)] w_i(\alpha))$, we have to replace $\delta$ by its integral over the interval $[x, x + \Delta x)$, with $\Delta x$ small enough:

$$\begin{aligned} &\int \delta[x - x(\alpha)] w_i(\alpha) \sigma_i(\alpha) d\alpha \\ &\quad \approx \frac{1}{\Delta x} \int_x^{x+\Delta x} \int \delta[x' - x(\alpha)] w_i(\alpha) \sigma_i(\alpha) d\alpha dx' \\ &\quad = \frac{1}{\Delta x} \int \Lambda_{x,\Delta x}(x(\alpha)) w_i(\alpha) \sigma_i(\alpha) d\alpha, \end{aligned}$$

where $\Lambda_{x,\Delta x}$ is defined by

$$\Lambda_{x,\Delta x}(y) = \begin{cases} 1, & \text{if } x \le y < x + \Delta x \\ 0, & \text{otherwise} \end{cases} \tag{19}$$

Note that $\Lambda_{x,\Delta x}(y) = h[(x + \Delta x) - y] - h[x - y]$, where $h$ is the step function. The estimator for $p_i$(x) is thus

$$H_i(x) = \frac{\overline{\Lambda_{x,\Delta x}(X_i) W_i}}{\Delta x \overline{W_i}} = \frac{\sum_{k=1}^{M_i} \Lambda_{x,\Delta x}(X_{i,k}) W_{i,k}}{\Delta x \sum_{k=1}^{M_i} W_{i,k}} \tag{20}$$

Such estimators are not unbiased, i.e., their expected values are not equal to the respective $p_i$, but their error is of the order of $1/M_i$, whereas their standard deviations are of the order of $1/\sqrt{M_i}$. Thus, for large $M_i$, the bias is negligible compared to the standard error of the estimator and we can use it without problem.[34]

### *Estimating $Var(H_i(x)\hat{Z}_i/\hat{Z}_j)$ Using the $\delta$-Method*

Now that we derived explicit formulas for the $H_i(x)$'s and the $Z_i$'s, we can find an estimator for $Var(H_i(x)\hat{Z}_i/\hat{Z}_j)$, thus completing the theoretical part. First we observe that

$$H_i(x) \frac{\hat{Z}_i}{\hat{Z}_j} = \frac{\overline{\Lambda_{x,\Delta x}(X_i) W_i}}{\Delta x \overline{W_i}} \frac{\frac{1}{c_i}\overline{W_i}}{\frac{1}{c_j}\overline{W_j}} = \frac{c_j}{c_i \Delta x} \frac{\overline{\Lambda_{x,\Delta x}(X_i) W_i}}{\Delta x \overline{W_j}} \tag{21}$$

Now,

$$Var\left( \frac{c_j}{c_i \Delta x} \frac{\overline{\Lambda_{x,\Delta x}(X_i) W_i}}{\overline{W_j}} \right) = \frac{c_j^2}{c_i^2 \Delta x^2} Var\left( \frac{\overline{\Lambda_{x,\Delta x}(X_i) W_i}}{\overline{W_j}} \right)$$

The variance of a ratio can be computed approximately using the $\delta$-method (mentioned in ref.,[35] chapters 4 and 7) i.e., approximating the ratio by its first-order Taylor expansion:

$$\begin{aligned} Var\left( \frac{\overline{\Lambda_{x,\Delta x}(X_i) W_i}}{\overline{W_j}} \right) &\approx \frac{1}{E(\overline{W_j})^2} \left( \frac{E(\overline{\Lambda_{x,\Delta x}(X_i) W_i})^2}{E(\overline{W_j})^2} Var(\overline{W_j}) + Var(\overline{\Lambda_{x,\Delta x}(X_i) W_i}) - 2 \frac{E(\overline{\Lambda_{x,\Delta x}(X_i) W_i})}{E(\overline{W_j})} Cov(\overline{\Lambda_{x,\Delta x}(X_i) W_i}, \overline{W_j}) \right) \\ &\approx \frac{1}{\overline{W_j}^2 M_i} \left( \frac{\overline{\Lambda_{x,\Delta x}(X_i) W_i}^2}{\overline{W_j}^2} S^2_{W_j} + S^2_{\Lambda_{x,\Delta x}(X_i) W_i} - 2 \frac{\overline{\Lambda_{x,\Delta x}(X_i) W_i}}{\overline{W_j}} S_{\Lambda_{x,\Delta x}(X_i) W_i, W_j} \right) \end{aligned} \tag{22}$$

where $S^2_{W_j}$, $S^2_{\Lambda_{x,\Delta x}(X_i) W_i}$ and $S_{\Lambda_{x,\Delta x}(X_i) W_i, W_j}$ are the usual variance and covariance estimators. Note that, if $j = 1$, then $Cov(\overline{\Lambda_{x,\Delta x}(X_i) W_i}, \overline{W_j}) = 0$.

## Application

To illustrate our method, we simulated athermal, self-avoiding polyethylene chains. In our model, a polyethylene chain is the trace of a self-avoiding walk restricted to a tetrahedral grid and the following geometric conditions: Given an overall arbitrary orientation, defined by the position of the first 3 particles, each new bond can be in one of the three possible states: *trans*, *gauche+*, and

*gauche*-, with $\phi$ angles of 0, 120, and $-120^\circ$, respectively; here, $\phi$ is the angle between the bond and its projection on the plane formed by the two previous bonds. The self-avoidance condition implies the following potential energy:

$$U(\alpha) = \begin{cases} \infty \text{ if } \mathbf{r}_j = \mathbf{r}_k,\ \forall j \neq k \text{ with } j,k \in \{0,\ldots,N-1\} \\ 0 \quad \text{otherwise} \end{cases} \tag{23}$$

where $\alpha$ is a chain configuration, $\mathbf{r}_j$ is the position of its $j$-th particle, and $N$ is the number of particles in the chain. We want to use the Rosenbluth sampling scheme to make a sample of chains submitted to an external force, **f**, that is being applied in the $x$ direction. In this case, the density distribution function of the system is

$$\pi_f = \frac{e^{-\beta U_f(\alpha)}}{Z_f}$$

We construct each chain step by step by placing one particle after another. The position of the next particle is chosen respecting the chain geometry and according to a probability $\sigma_f \neq \pi_f$ in the following way:

1. Place particle 0 in the origin of the coordinates system.
2. **for** particles $j = 1$ **to** $N-1$ **do**
   - Look for sites neighbor to particle $j-1$ that are free.
   - **if** there are free sites **then**
     choose one of them with probability

$$\frac{e^{\beta f(\mathrm{x_j}-\mathrm{x_{j-1}})}}{\sum_{k=1}^{K_j} e^{\beta f(\mathrm{x_k^{(j)}}-\mathrm{x_{j-1}})}}$$

     where $K_j$ is the number of free sites when placing the $j$-th particle and $\mathrm{x_k^{(j)}}$ is the x component corresponding to a free site $k$.
   - **or else**
     Reinitialize the construction (go to step 1).

The probability $\sigma_f(\alpha)$ of generating a certain self-avoiding configuration $\alpha$ by the Rosenbluth sampling is

$$\begin{aligned} \sigma_f(\alpha) &= \prod_{j=1}^{N-1} \frac{e^{\beta f(\mathrm{x_j}-\mathrm{x_{j-1}})}}{\sum_{k=1}^{K_j} e^{\beta f(\mathrm{x_k^{(j)}}-\mathrm{x_{j-1}})}} = \frac{\prod_{j=1}^{N-1} e^{\beta f(\mathrm{x_j}-\mathrm{x_{j-1}})}}{\prod_{j=1}^{N-1}\sum_{k=1}^{K^j} e^{\beta f(\mathrm{x_k^{(j)}}-\mathrm{x_{j-1}})}} \\ &= \frac{e^{\beta f \sum_{j=1}^{N-1}(\mathrm{x_j}-\mathrm{x_{j-1}})}}{\prod_{j=1}^{N-1}\sum_{k=1}^{K_j} e^{\beta f(\mathrm{x_k^{(j)}}-\mathrm{x_{j-1}})}} \\ &= \frac{e^{\beta f x(\alpha)}}{\prod_{j=1}^{N-1}\sum_{k=1}^{K_j} e^{\beta f(\mathrm{x_k^{(j)}}-\mathrm{x_{j-1}})}} \end{aligned} \tag{24}$$

which gives the following weighting function:

$$w_f(\alpha) = \frac{1}{c_f}\prod_{j=1}^{N-1}\sum_{k=1}^{K_j} e^{\beta f(\mathrm{x_k^{(j)}}-\mathrm{x_{j-1}})} \propto \frac{\pi_f(\alpha)}{\sigma_f(\alpha)} \tag{25}$$

The prefactor, $1/c_f$, is needed to prevent overflow in the computation of weights, mainly in cases where the number of particles is large.

## Implementation Issues

Our implementation has two separate parts. The first part uses the Rosenbluth sampling to construct a sample of $M_i$ chains of length $N$ subjected to a chosen potential energy, $U_i$, corresponding to a external force $\mathbf{f_i}$. The program performs the simulation for the longest desired length and also stores the results for the desired intermediate chain lengths. The second part gathers the data generated in various instances of the first part (for different $\mathbf{f_i}$) and constructs the desired histograms.

### *Chain Construction*

A simple way to build a tetrahedral grid is by using a cubic grid as basis and restricting the random walk to a tetrahedral grid (a description is given by Cifra and Bleha[36]).

To minimize the time of searching for self-intersections, it is easiest to use a lattice with linear dimensions larger than the total chain length. However, this approach is wasteful in computer storage. We, therefore, employ a hashing data structure, where we ''project'' the coordinates of all particles into a smaller box. The positions of the particles are stored using a cell index method mentioned in ref.,[37] chapter 5.3. This is done as follows: a cubic region of edge of length $L = k \cdot l$ is divided into $k^3$ smaller cubic cells of edge of length $l$. A cell numbered $n_{\mathrm{cell}}$ stores all particles with coordinates $(x, y, z)$ such that

$$n_{\mathrm{cell}} = i_x \cdot k^2 + i_y \cdot k + i_z \tag{26}$$

where

$$i_a = \left\lfloor \frac{a}{l} \right\rfloor \bmod k, \quad \text{for } a = x, y, z \tag{27}$$

Consequently, particles with very different real coordinates may be stored in the same cell. Hence, within one cell, it is necessary to distinguish particles that have different coordinates from those that would physically overlap. Because we do keep track of the real $\{x, y, z\}$-coordinates of all particles, the test for overlap is straightforward. The advantage of the current scheme is that the number of particles within one cell can be kept small, whereas the memory requirements for storage are modest. In our experience, for chains sizes up to 1 000, the choice of $k = 16$ (which is much smaller than the typical lattice size that would be needed when using periodic boundary conditions) hardly ever leads to more than 1 or 2 particles in a cell. It is straightforward to extend this scheme to the off-lattice case.

### *Multiple Histogram*

To achieve better results, we have made some modifications to the described method. First, we have made use of the symmetry of the problem: a potential energy $U_f = U_0 + f \cdot x$ is symmetric to $U_f = U_0 - f \cdot x$; by using this property, we choose only positive forces and get results for both positive and negative forces.

Furthermore, we have added an acceptance parameter. We discard $X$ entries with less than a percentage of the maximum number of entries in a bin. In this way, we can filter out the least reliable entries, which could lead to a wrong computation of weights $a_i(X)$.

The computations involved in combining the various histograms contain the factor

$$C_{i,j}(x) = \frac{c_i}{c_j} e^{-\beta(f_i - f_j)x} \tag{28}$$

which can easily lead to overflow. To prevent this (to a certain extent), for each entry of the histogram, we choose a force, $f_m$, the smallest force for which that entry has enough hits, and factor $C_{j,m}$ out of the computation. Resulting in the following expression for $p_j^{est}$

$$p_j^{est}(x) = C_{j,m}(x) \frac{1}{\Delta x} \sum_{i=0}^{N} a_i^*(x) C_{m,i}(x) e^{-\beta(f_m - f_i)x} \frac{\overline{\Lambda_{x,\Delta x}(X_i) W_i}}{\Delta x \overline{W}_j} \tag{29}$$

where

$$a_i^*(x) = \frac{C_{i,m}^2(x) / Var\left(\frac{\overline{\Lambda_{x,\Delta x}(X_i) W_i}}{\overline{W}_j}\right)}{\sum_{k=0}^{N} \left[ C_{k,m}^2(x) / Var\left(\frac{\overline{\Lambda_{x,\Delta x}(X_k) W_k}}{\overline{W}_j}\right)\right]} \tag{30}$$

## Results and Discussion

In the present work, we simulated chains with up to 600 particles. For each chain size $N$, we made a set of simulation runs for various external forces varying from 0 to 7 in $k_bT$ units. For each run, we generated a total of $10^7$ sample chains. Histogram bins containing less than 0.1% of the maximum number of entries in a bin where ignored.

As shown in Figure 1, the histograms obtained by the simple Rosenbluth sampling are greatly improved with the application of the multiple histogram method (without the need of extra simulation time). Apart from an overall reduction of the noise, this improvement is more apparent in the wings of the histograms, that are not adequately sampled by the simple Rosenbluth sampling. Even though the results of the combined Rosenbluth-MHM are much better, the histograms are still noisy (mainly for longer chains). With this noise, it becomes hard to compute the rectractive force, $f(x) = -F'(x)$, numerically. To circumvent this problem, we approximate $F(x)$ for small $x$ by a second degree polynomial and compute the derivative of the outer of $F(x)$ numerically by fitting straight lines to consecutive segments of points. The gradient of each straight line was taken as the estimate of the derivative of $F(x)$ at the mid-point of the segment.

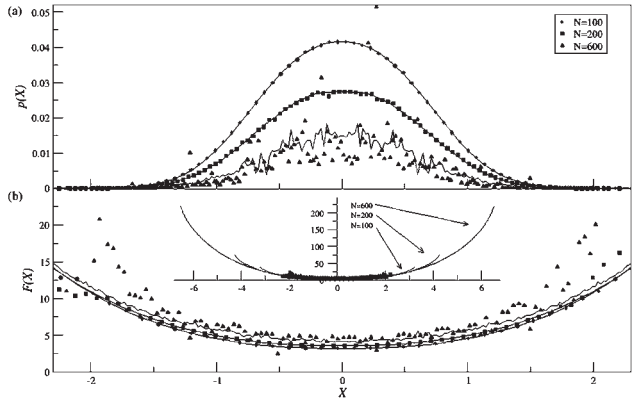



Figure 1. Comparison of the plain Rosenbluth sampling (symbols) and the combined Rosenbluth-MHM (solid lines). Results obtained for $N = 100$, 200, and 600. The histograms, $p(X)$ (Figure 1a), and the free energies, $F(X)$ (Figure 1b), were computed as functions of the reduced strain, $X = x/\sqrt{\langle R_0^2 \rangle}$, where $\langle R_0^2 \rangle$, the average end-to-end distance of an unperturbed chain, was estimated during the simulation. The inset in Figure 1b shows that the plain Rosenbluth sampling does not give results for large $X$.

Figure 2 shows the stress-stain curves obtained using the combined Rosenbluth-MH method in the constant-stress ensemble. The stress ensemble results agree with scaling theory,[32,33] mainly for small forces, where the prefactor $1/3$ matches very well. In the finite extensibility regime, the curve reaches a plateau which is indicative of the maximal elongation of the chain. In the constant-strain ensemble (Figure 3), the behavior of the strong force and finite extensibility regimes are similar to that of the respective regimes in the stress ensemble. However, in the small force regime, the difference in the fluctuation properties of the stress and strain ensembles shows up as considerable finite-size effects.[38,39] At larger extensions, where the histograms become much more strongly peaked, fluctuations, and thereby finite-size effects, are much less important and both sets of simulations converge towards the predicted theoretical curve as $N \to \infty$.

## Conclusions

In this paper, we have shown how to adapt the multiple histogram method to be used together with the Rosenbluth sampling. We have tested our approach by studying the elastic properties of self-avoiding chains in a tetrahedral lattice both for the stress ensemble and for the strain ensemble.

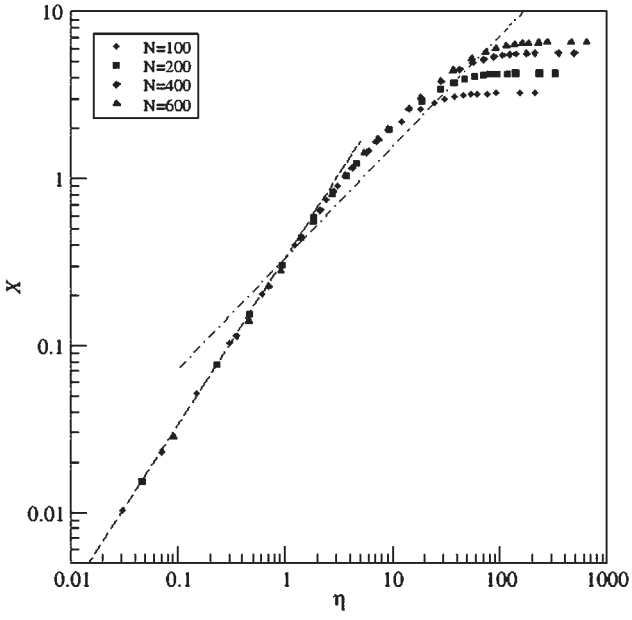



Figure 2. Stress-strain curves computed in the constant-stress ensemble. The strain, $X$, is expressed in reduced units $X = x/\sqrt{\langle R_0^2 \rangle}$. The reduced stress, $\eta$, is defined as $\eta = \sqrt{\langle R_0^2 \rangle} f/(k_B T)$. Results obtained for $N = 100$, 200, 400, and 600. The dashed line corresponds to the small forces regime ($X = \frac{1}{3}\eta$) and the dot-dashed line to the strong force regime ($X = \frac{1}{3}\eta^{2/3}$).[32,33]

The combined Rosenbluth-MHM is an efficient method to study the elastic properties of single chains, because one simulation set can provide results both in the stress and stain ensembles, for chains of all sizes up to a maximum size. Furthermore, it is also possible to obtain results for a range of temperatures using the same techniques described here.

Our test case showed that the histograms computed using the combined Rosenbluth-MHM are much better than ones computed using the simple Rosenbluth sampling, mainly in the case of strongly stretched chains where the simple Rosenbluth does not produce results at all.

We find that the stress-strain curves for the stress ensemble agree well with the predictions of scaling theory.[32,33] For the constant-strain ensemble, the agreement with theory is also good, except at small chains at small extensions, where finite-size effect take place.

*Acknowledgement:* We thank Dr. *Mark Miller* and Dr. *Rob H. Bisseling* for the useful discussions and computational advice. This work is part of the Softlink research program of the ''Stichting voor Fundamenteel Onderzoek der Materie (FOM)'', which is financially supported by the ''*Nederlandse organisatie voor Wetenschappelijk Onderzoek* (NWO)''.

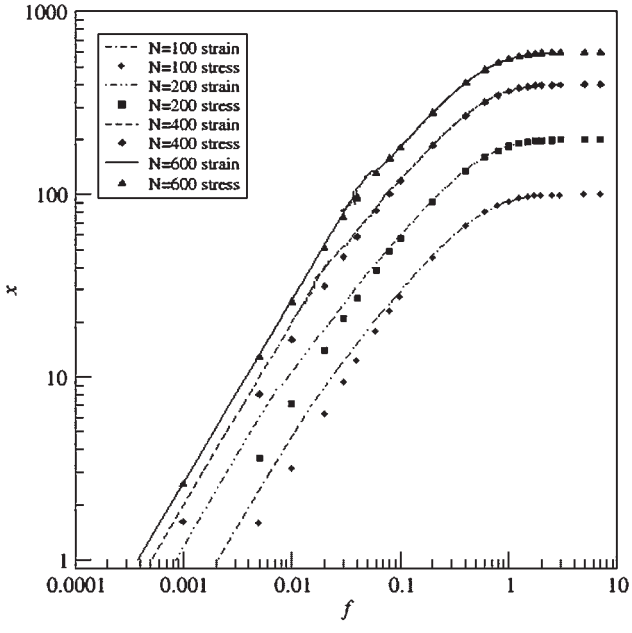



Figure 3. Comparison of the stress-strain curves for the stress ensemble (symbols) and the strain ensemble (lines). Results obtained for $N = 100$, 200, 400, and 600. For small $x$, $f(x) = -F'(x)$ was computed by aproximating $F(x)$ by a second degree polynomial and then taking the derivative. For large $x$, $f(x) = -F'(x)$ was computed numerically.

[1] J. E. Bemis, B. B. Akhremitchev, G. C. Walker, *Langmuir* **1999**, *15*, 2799.
[2] H. Li, W. Zhang, X. Zhang, J. Shen, B. Liu, C. Gao, G. Zou, *Macromol. Rapid Commun.* **1998**, *19*, 609.
[3] C. Ortiz, G. Hadziioannou, *Macromolecules* **1999**, *32*, 780.
[4] W. Zhang, S. Zou, C. Wang, X. Zhang, *J. Phys. Chem. B* **2000**, *104*, 10258.
[5] C. Bustamante, Z. Bryant, S. B. Smith, *Nature* **2003**, *421*, 423.
[6] P. Cluzel, A. Lebrun, C. Heller, R. Lavery, J.-L. Viovy, D. Chatenay, F. Caron, *Science* **1996**, *271*, 792.
[7] S. B. Smith, L. Finzi, C. Bustamante, *Science* **1992**, *258*, 1122.
[8] T. R. Strick, M.-N. Dessinges, G. Chavin, N. H. Dekker, J.-F. Allemand, D. Bensimon, V. Croquete, *Rep. Prog. Phys.* **2003**, *66*, 1.
[9] M. C. Williams, I. Rouzina, *Curr. Opin. Struct. Biol.* **2002**, *12*, 330.
[10] M. S. Z. Kellermayer, S. B. Smith, H. L. Granzier, C. Bustamante, *Science* **1997**, *276*, 1112.
[11] M. C. Leake, D. Wilson, B. Bullard, R. M. Simmons, *FEBS Lett.* **2003**, *535*, 55.
[12] A. F. Oberhauser, P. E. Marszalek, H. P. Erickson, J. M. Fernandez, *Nature* **1998**, *393*, 181.
[13] M. Rief, M. Gautel, F. Oesterhelt, J. M. Fernandez, H. E. Gaub, *Science* **1997**, *276*, 1109.
[14] A. Ptak, S. Takeda, C. Nakamura, J. Miyake, *J. Appl. Phys.* **2001**, *90*, 3095.
[15] L. Tskhovrebova, J. Trinick, J. A. Sleep, R. M. Simmons, *Nature* **1997**, *387*, 308.

[16] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, E. Teller, *J. Chem. Phys.* **1953**, 1087.
[17] P. Cifra, T. Bleha, *Macromol. Theory Simul.* **1995**, *4*, 233.
[18] P.-Y. Lai, *Phys. Rev. E* **1998**, *55*, 6222.
[19] P.-Y. Lai, Y.-J. Sheng, H.-K. Tsao, *Physica A* **1998**, *254*, 280.
[20] Y.-J. Sheng, P.-Y. Lai, *Phys. Rev. E* **1997**, *56*, 1900.
[21] J. T. Titantah, C. Pierleoni, J.-P. Ryckaert, *J. Chem. Phys.* **2002**, *117*, 9028.
[22] I. Webman, J. L. Lebowitz, M. H. Kalos, *Phys. Rev. A* **1981**, *23*, 316.
[23] M. Wittkop, J.-U. Sommer, S. Kreitmeier, D. Göritz, *Phys. Rev. E* **1994**, *49*, 5472.
[24] S. Kreitmeier, M. Wittkop, D. Göritz, *Phys. Rev. E* **1999**, *59*, 1982.
[25] M. Wittkop, S. Kreitmeier, D. Göritz, *J. Chem. Soc., Faraday Trans.* **1996**, *92*, 1375.
[26] M. Wittkop, S. Kreitmeier, D. Göritz, *Phys. Rev. E* **1996**, *53*, 838.
[27] A. M. Ferrenberg, R. H. Swendsen, *Phys. Rev. Lett.* **1989**, *63*, 1195.
[28] D. K. Klimov, D. Thirumalai, *J. Phys. Chem. B* **2001**, *105*, 6648.
[29] P. Grassberger, H.-P. Hsu, *Phys. Rev. E* **2002**, *65*, 031807.
[30] P. Grassberger, *Phys. Rev. E* **1997**, *56*, 3682.
[31] M. N. Rosenbluth, A. W. Rosenbluth, *J. Chem. Phys.* **1955**, *23*, 356.
[32] P.-G. de Gennes, ''*Scaling Concepts in Polymer Physics*'', Cornell University Press, Ithaca 1979.
[33] P. Pincus, *Macromolecules* **1976**, *9*, 386.
[34] F. L. McCrackin, *J. Res. Natl. Bur. Stand. B Math. Sci.* **1972**, *76B*, 193.
[35] J. A. Rice, ''*Mathematical Statistics and Data Analisys*'', 2nd edition, Duxbury Press, Belmont, CA 1995, p. 602.
[36] P. Cifra, T. Bleha, *Macromol. Theory Simul.* **1995**, *4*, 405.
[37] M. P. Allen, D. J. Tildesley, ''*Computer Simulations of Liquids*'', Oxford Science Publications, New York 1987.
[38] D. E. Macarov, Z. Wang, J. B. Thompson, H. G. Hansma, *J. Chem. Phys.* **2002**, *116*, 7760.
[39] J. T. Titantah, C. Pierleoni, J.-P. Ryckaert, *Phys. Rev. E* **1999**, *60*, 7010.