Monte Carlo Backbone Sampling for Nucleic Acids Using Concerted Rotations Including Variable Bond Angles

Jakob P. Ulmschneider and William L. Jorgensen*

Department of Chemistry, Yale University, New Haven, Connecticut 06520-8107

Received: May 21, 2004; In Final Form: August 9, 2004

An efficient concerted rotation algorithm for use in Monte Carlo statistical mechanics simulations of nucleic acids is reported. The corresponding algorithm "concerted rotations with flexible bond angles" (CRA) for sampling polypeptides was found to be superior to local moves that included only flexible dihedral angles by allowing exploration of a larger conformational space as well as facilitating backbone transitions. The performance of the present CRA algorithm for polynucleotides is compared to two alternatives, a simple update of main-chain torsion angles and a previously reported, concerted rotation algorithm with fixed bond angles and a mix of flexible and rigid main-chain dihedral angles. The test system is a 12 base-pair duplex B-form DNA helix, and the performance comparisons are made for the system both in a vacuum and with continuum GB/SA solvation. The results demonstrate the superior efficiency of the CRA method over the alternatives.

Introduction

For efficient sampling of chain molecules such as polyethylene, proteins, or nucleic acids in Monte Carlo statistical mechanics (MC) simulations, it is advantageous to employ local, crankshaft-like moves. This circumvents the problems caused by the use of simpler updates such as the "pivot" method in which a single torsion angle is turned that, while reasonably efficient for short chains, become less effective for larger or dense systems due to the global nature of the changes. Local moves avoid large conformational changes by restricting the modifications to a segment of the chain, while leaving the rest of the molecule unchanged. In addition, they are more computationally efficient, since the energy for only a small part of the whole system has to be re-evaluated. Many different schemes for local moves have been proposed. Configurational bias regrowth procedures remove a local segment of a chain and rebuild it step-by-step from one end.[1−3] The segment can be closed by biasing the regrowth toward this goal.[4,5] A computationally more efficient approach is to reduce the system to the essential torsional degrees of freedom and directly calculate all chain closure solutions. This method has been termed "concerted rotations" or "conrot". Based on theoretical work by Go and Scheraga[6] and after an early basic application by Wakana et al.,[7] this method has been successfully applied to polymers by Dodd et al.[8] and to polypeptides by Knapp.[9,10] Usually a number of additional "driver" torsion angles are randomly perturbed and the remaining six consecutive torsion angles are determined by numerical solution of the closure equations. Various extensions and modifications have been proposed,[11−15] and the conrot scheme has also been combined with configurational-bias[11,12] and internal-bias regrowth procedures.[5] Since the original equations[6] were derived for a protein backbone, simplifications were used that took advantage of the fact that the rotatable bonds along the backbone are separated by rigid bonds in the trans conformation. Dinner overcame this limitation by an extension of the equations to allow arbitrary values of the fixed torsion angles.[13] Simulations are then permitted for systems that contain cis conformations of the peptide bond (e.g., *cis*-proline) or values of dihedral angles that may deviate slightly from planarity (e.g., if the initial coordinates are taken from crystal structures). This also enables application of the conrot approach to other polymers such as nucleic acids, where the sugar δ dihedral angle, which might be treated as fixed, is not trans.

These conrot methods still have drawbacks. In particular, the constraint for fixed bond angles is overly restrictive, and it is inconsistent with normal practices in molecular dynamics (MD) simulations of proteins and nucleic acids. Originally proposed to promote conformational sampling by neglect of uninteresting high-frequency motions, it has been noted that by applying holonomic constraints to bond lengths and angles, MD simulations exhibit dynamics, which is slowed considerably due to the lack of flexibility.[16,17] In particular, energetic fluctuations at room temperature normally lead to substantial bond angle variations, which facilitates conformational transitions. The use of traditional molecular mechanics force fields with conrot-like torsional moves and fixed bond angles results in unwanted artifacts, since the force fields are not parametrized for such rigid models.[17] Modified torsional potentials could be used to compensate for the increased torsional barriers.[17] However, we developed instead a modified conrot algorithm, CRA, that includes flexible bond angles throughout.[18] The CRA algorithm features a Gaussian biasing scheme, similar to the one described by Favrin et al.,[19] and it was found to be superior to the original conrot method for sampling polypeptides.[18] In a recent study, the algorithm was combined with the GB/SA solvation model[20] in folding studies for a series of polypeptides that succeeded in locating the native states from initially extended structures.[21] The CRA algorithm can be applied to any polymer molecule and does not require any intermediate rigid torsional units.

All schemes for MC moves must obey microscopic reversibility by satisfying the detailed balance condition. This leads to an additional drawback of the original conrot methods, caused by the inclusion of the driver torsion angles that are central to the algorithm. To satisfy detailed balance, there has to be both a "forward" move, in which the driver torsions are randomly

* To whom correspondence should be addressed. E-mail: william.jorgensen@yale.edu.

10.1021/jp047796z CCC: $27.50
Published on Web 10/02/2004

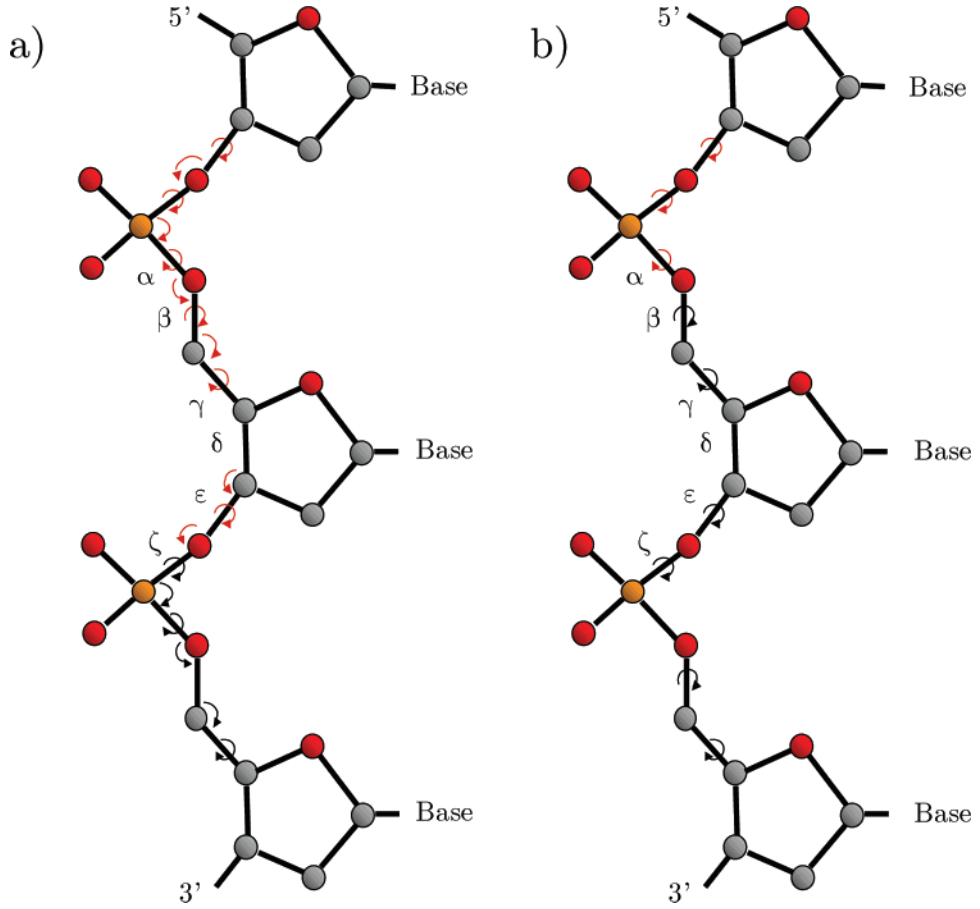

Figure 1. Schematic picture of the local moves implemented for the sugar−phosphate backbone: (a) CRA algorithm; (b) CR algorithm. The involved dof are pictured as red if they are prerotated and black if they are determined by chain closure equations. Both methods are implemented to span a segment of equal size and vary the same number of backbone torsion angles.

perturbed, and a "reverse" move, where they are not. The final conformation has to be picked from the complete set of solutions determined for both forward and reverse moves, with a Jacobian weight assigned to each solution. This procedure is inefficient for a variety of reasons. If a solution of the reverse move is picked, it samples less conformational space by only changing 6 torsion angles. In addition, the changes for the 6 torsion angles determined by the chain closure equations are usually large, for both forward and reverse moves. These changes cannot be made smaller, even if no driver torsions are used. For systems with large bulky side chains, as for nucleic acids, high rejection rates and inefficiency ensue. There have been some attempts to alleviate this problem by restricting the solution set to enforce only minimal changes for the 6 torsion angles.[22] With the CRA algorithm these problems are overcome; the size of the torsion perturbations can be adjusted adequately, and no reverse move is required to ensure microscopic reversibility.

In the present study, the CRA algorithm is extended to nucleic acids. Atomic-level computer simulations of DNA and RNA systems have previously been almost exclusively attempted within the framework of MD calculations.[23] Given the success of the CRA method for polypeptides,[21] it is of interest to apply such methodology in MC simulations of DNA/RNA systems. With Dinner's previous conrot methodology with fixed bond angles,[13] which we will refer to as CR, an RNA loop was studied, and it was concluded that the conrot moves improve efficiency over simple MC sampling of backbone torsion angles. In the following, results from the CRA method are compared to those from corresponding simulations with these two alternatives.

Methods

To sample polynucleotides, some modifications had to be made to the polypeptide version of the CRA algorithm. The flexible main-chain torsion angles are α, β, γ, ϵ, and ζ, while δ is left fixed, as depicted in Figure 1. To sample a substantial segment of the backbone while at the same time moving as few side chains as possible, the local segment includes only 1 base and stretches over 11 consecutive bonds, as shown in Figure 1a. Of the 18 degrees of freedom (dof) sampled (9 torsion angles and 9 bond angles), 12 are covered by the CRA biasing prerotation scheme, and 6 are solved for by the closure equations. In the CRA algorithm, the displacement vector $\delta\phi$ of the dof is picked from the biased distribution in eq 1.

$$p(\delta\phi)d^n(\delta\phi) = \frac{1}{\pi^{n/2}} e^{-\delta\phi^r \mathbf{J} \delta\phi} \det \mathbf{L}\, d^n(\delta\phi) \tag{1}$$

$$\mathbf{J} = a(1 + b\mathbf{I}) \tag{2}$$

$$\mathbf{J} = \mathbf{L}\mathbf{L}^t \tag{3}$$

$$I_{ij} = \left(\frac{\partial \mathbf{a}}{\partial \phi_i}\right) \cdot \left(\frac{\partial \mathbf{a}}{\partial \phi_j}\right) \tag{4}$$

where **a** is the position vector of the last atom in the prerotation step, and $\delta\phi_i$ are the dof. **I** is a conformation-dependent matrix that contains the effect of the constituent dof on the absolute displacement of the last atom of the local segment. The **J** matrix has a unit matrix added to **I** which is necessary to randomize the method; a and b are adjustable parameters corresponding

to the overall size and strength of the biasing, respectively. The **L** matrix is obtained by a Cholesky decomposition of the **J** matrix. All these matrices are conformation dependent. For the algorithm, this implies that the probability to propose a local move from a conformation n to m, $B(n\rightarrow m)$ (obtained through eq 1), will not be the same as the probability of reversing such a move $B(m\rightarrow n)$. For the latter probability, the matrices **I′**, **J′**, and **L′** for the m-conformation have to be calculated, and both transition probabilities must be calculated to ensure detailed balance. The method was implemented as previously described[18] with modifications for the change in the number of dof that make up the biased prerotation step. There are now 6 torsion and 6 bond angles that are prerotated. The dimension of the square **I**, **J**, and **L** matrices is reduced to 12. The scaling factor c that controls the ratio of torsion angle to bond angle perturbations is included by a modification of the **L** matrix (eq 5). The optimal values of the three parameters, a, b, and c (eqs 2 and 5), were redetermined here for polynucleotide simulations. The chain closure procedure is as described previously,[18] with a numerical root search of a two-valued function and the inclusion of a Jacobian factor to achieve detailed balance.

$$L_{ij} \rightarrow \begin{cases} L_{ij} & 1 \le j \le 6 \\ cL_{ij} & 7 \le j \le 12 \end{cases} \tag{5}$$

A detail that is worth recalling is that all atoms that are connected to the backbone rigidly rotate with the backbone moves.[16] For such atoms, for example, a phosphate oxygen, the 3 internal degrees of freedom (r, θ, φ) that determine its position relative to the backbone are kept fixed. However, due to the alteration of the bond angles in the backbone, several additional bond angles involving phosphate oxygens will change. The corresponding energy change for these angles is accounted for in the evaluation of the method. However, it is always the bond angles in the backbone that are sampled, and only these contribute to the geometric change of the last atom in the segment, so only these contribute to the Jacobian. The additional bond angles only contribute to the potential energy.

The equivalent scheme for the CR algorithm of Dinner[13] is illustrated in Figure 1b. These moves involve only the backbone torsion angles except that the sugar δ torsion is fixed. The algorithm was set up exactly as previously described,[13] with a numerical root search of a 4-valued function and a weighting of solutions proportional to their Jacobian factor: $B(m\rightarrow n) \sim J_b$. The only difference is the present use of three driver torsions instead of one to increase the total number of torsion angles to nine in order to enable a direct comparison with the CRA method (Figure 1). Parallel calculations were also carried out with the simple pivot method for the torsional sampling again with fixed bond angles.

The test system is the 12 base-pair duplex B-form DNA with the sequence d(CGCGAATTCGCG), also known as the "Dickerson dodecamer".[24] Initial coordinates were obtained from the crystal structure with PDB code 1BNA. The simulations were run with the MCPRO program[25] modified to include the concerted rotations, as detailed in the original report.[18] The system was modeled in a vacuum and in GB/SA continuum solvent using the OPLS-AA force field[26] with the exception that charges on the phosphate groups were neutralized by removing 0.25 e from each oxygen to approximate the effect of counterion and other electrostatic screening. For the vacuum and continuum simulations, the full potential energy was evaluated with no cutoffs for the nonbonded interactions and with a dielectric constant of 1 for the Coulombic interactions. First, a conjugate gradient minimization was carried out to relax the crystal structure. The system was then equilibrated at 25 °C using the CRA backbone sampling and pivot side-chain moves in a ratio of 1:1 for a total of 80 M configurations. The side-chain moves were handled as usual with the MCPRO program, with sampling of all dof in the nucleotides, including the sugar ring. Since the purpose of this study is the evaluation of the sampling efficiency of conrot algorithms in DNA simulations, the described setup was sufficient.

Next, the optimum parameters were determined for all three algorithms. The principal efficiency indicator is the torsional step size per dihedral angle (eq 6),

$$\delta\phi_{\mathrm{rms}} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\delta\phi_i)^2} \tag{6}$$

which is the root-mean-square displacement of a single torsion angle in an accepted local move that involves the n torsion angles $\phi_1,\cdots,\phi_n$. If the move is rejected, $\delta\phi_{\mathrm{rms}} = 0$. For the simple main-chain pivot move of a torsion angle $\delta\phi_i$, the step size is $\delta\phi_{\mathrm{rms}} = |\delta\phi_i|$ if the move is accepted and $\delta\phi_{\mathrm{rms}} = 0$ otherwise. The average step size over the course of the simulation, $\langle\delta\phi_{\mathrm{rms}}\rangle$, converges after only a few thousand MC moves.

Results

Parameter *b* Optimization. For the CRA method, a good understanding of the effect of the geometrical biasing can be achieved by looking at the behavior of the torsional step size as a function of the biasing parameter b (eq 2). The value of b can range from zero, for unbiased random moves, to a large number for extreme biasing. The biasing behavior is displayed in Figure 2, for constant $a = 1000$, and $c = 20$. Figure 2a shows the absolute Cartesian displacement of all backbone atoms involved in the biased prerotations for a sample backbone segment. The displacements are averaged over 10 000 prerotations, as b is varied over several orders of magnitude. If b is small, the prerotations are random and there is a linear increase in displacement along the segment, as expected. Closing the chain will be difficult. The more b is increased, the more the displacement curve becomes bulge-shaped, with only a small change in position of the last atom.

Figure 2b shows the displacement distribution of the last atom as a function of b, revealing the bias toward small change as b is increased. The torsional step size, the acceptance rate, and the rate of failure to achieve a numerical chain closure are shown in Figure 2c as a function of b. The value of b is varied over 5 orders of magnitude, and averages are taken over 0.1 M steps. For very small values of b, the moves are random, with a low acceptance rate of less than 10%, and there are a high number of moves that result in failure to close the chain. The moves are still usable owing to the large value of a, but they are not efficient. As the value of b rises, there is a rapid increase in step size and acceptance rate, and the failure rate drops to zero. The biasing seems to be most efficient for $b \approx 1-10$, with $\langle\delta\phi_{\mathrm{rms}}\rangle$ more than 2.5 times the value from unbiased sampling. As b is increased even further, the efficiency drops. For large b, the distributions of the dof become highly perturbed and narrow along the direction corresponding to the eigenvector of the **I** matrix with eigenvalue of zero. Since this geometric change is very nonrandom, the probability to reverse such a move is small and the acceptance probability is very low. This is because the eigenvectors corresponding to the eigenvalue zero of the **I** matrix are not the same as the eigenvectors of the **I′** matrix

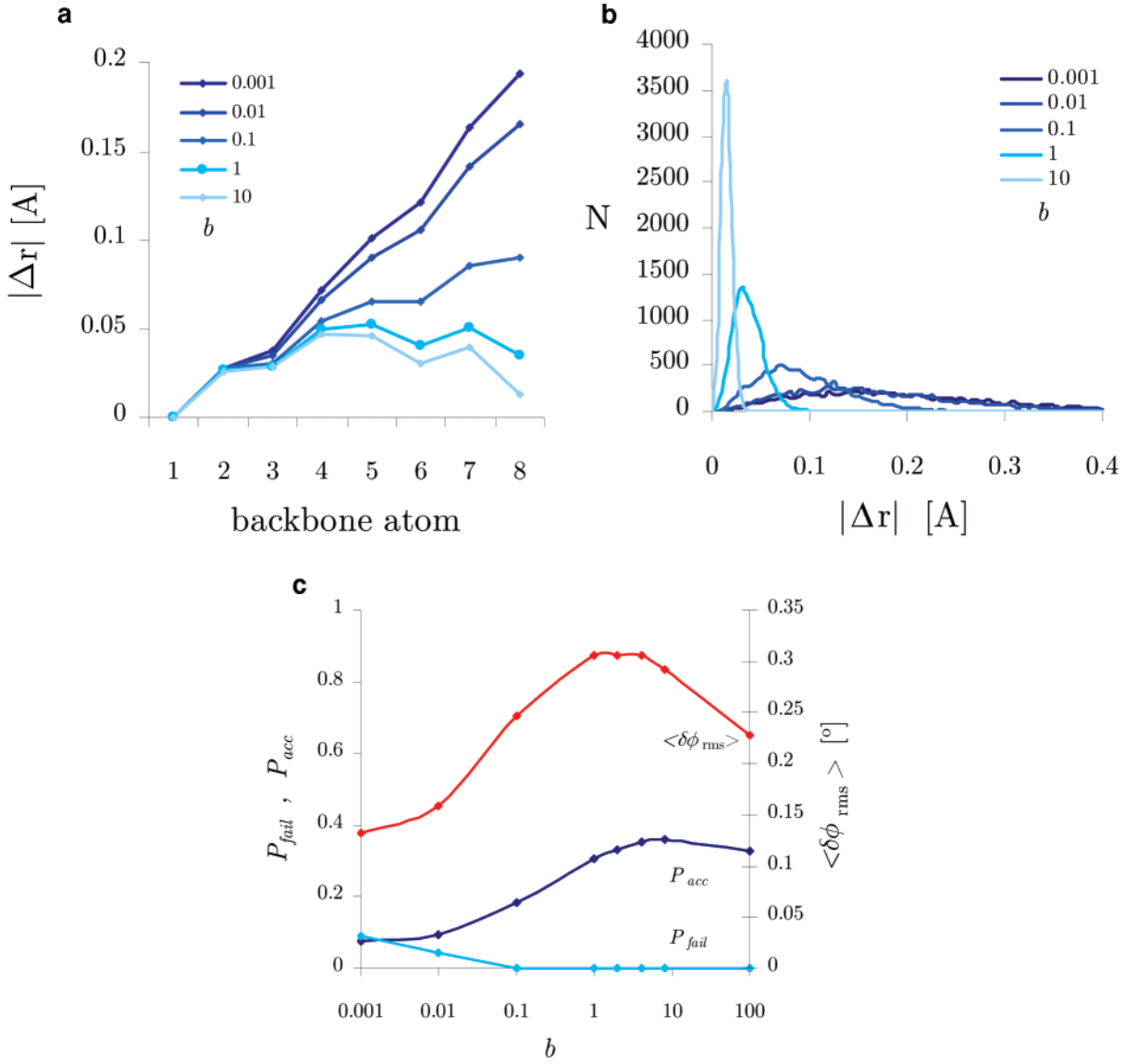

Figure 2. Dependence of the conformational sampling on the value of the Gaussian biasing parameter *b*. (a) The average displacement of the backbone atoms involved in the prerotations of a sample segment for CRA moves with $a = 1000$, $c = 20$. The results were averaged over 10 000 prerotations. (b) The corresponding distribution of the displacement of the last atom of the prerotation segment. (c) Torsional step size (right scale), acceptance rate and chain closure failure rate (left scale) as a function of the biasing rate *b*. Averages are over 10^5 MC configurations.

that determines the reverse move. To propose a reverse move along this direction is then very improbable, with $B(m \rightarrow n) \approx 0$. An optimal *b* must therefore exist between the two extremes, as is clearly visible from Figure 2c. It was also found that the behavior of *b* is more or less independent of *a*, and the biasing rate can be set to its optimum value of $b = 8$, for all values of *a*. This simplifies the present study, since all further optimization only involves the parameter *a*, which corresponds to the parameters used in the other algorithms.

Parameter *a* and $\Delta\phi_{max}$ Optimzation. A series of MC simulations were run for 0.1 M steps to determine the optimal torsional step size for the three methods as a function of the corresponding principal parameter. This is the parameter *a* for the CRA method and $\Delta\phi_{max}$ for both the CR and pivot methods. The optimum value for the biasing strength *b* of the CRA algorithm was set to 8 and the scale of dihedral angles to bond angles *c* was set to 20, as above and for polypeptides.[18] Figure 3 shows the average torsional step size and the average acceptance rate as a function of *a* and $\Delta\phi_{max}$ for the three algorithms. Figure 3a shows the case for the CRA algorithm, as *a* is varied over 3 orders of magnitude. For large *a*, the conformational change and the torsional step size are small, with a high rate of acceptance ($P_{acc} > 0.8$). If *a* is made small, the conformational change is large, but the acceptance rate and the step size become very low ($P_{acc} < 0.1$), since many attempted moves result in failure to close the chain. The optimum local moves are achieved at $a \approx 1000$, with $P_{acc} \approx 0.4$ and $\langle\delta\phi_{rms}\rangle \approx 0.3°$.

The corresponding graph for the CR algorithm is shown in Figure 3b. The behavior of the torsional step size is essentially similar to the CRA case, with small efficiency for both high $\Delta\phi_{max} > 5°$ due to failure to close the chain and low $\Delta\phi_{max} < 0.5°$ due to the small change. The maximum occurs around $\Delta\phi_{max} \approx 1°$ with a much smaller $\langle\delta\phi_{rms}\rangle$, ca. 0.06°, than with the CRA algorithm. The acceptance rate exhibits a different pattern; it only drops from $P_{acc} = 0.55$ to 0.3 as $\Delta\phi_{max}$ is increased by over a factor of 1000. The limited range can be understood from the workings of the CR algorithm: Even if $\Delta\phi_{max}$ is made very small or even zero, there will always be a constant geometrical change from solving the 6 dof of the chain closure algorithm. The somewhat drastic change of these 6 torsion angles is thought to be one of the reasons for the reduced efficiency of the CR method.[22] Thus, the CR moves cannot be

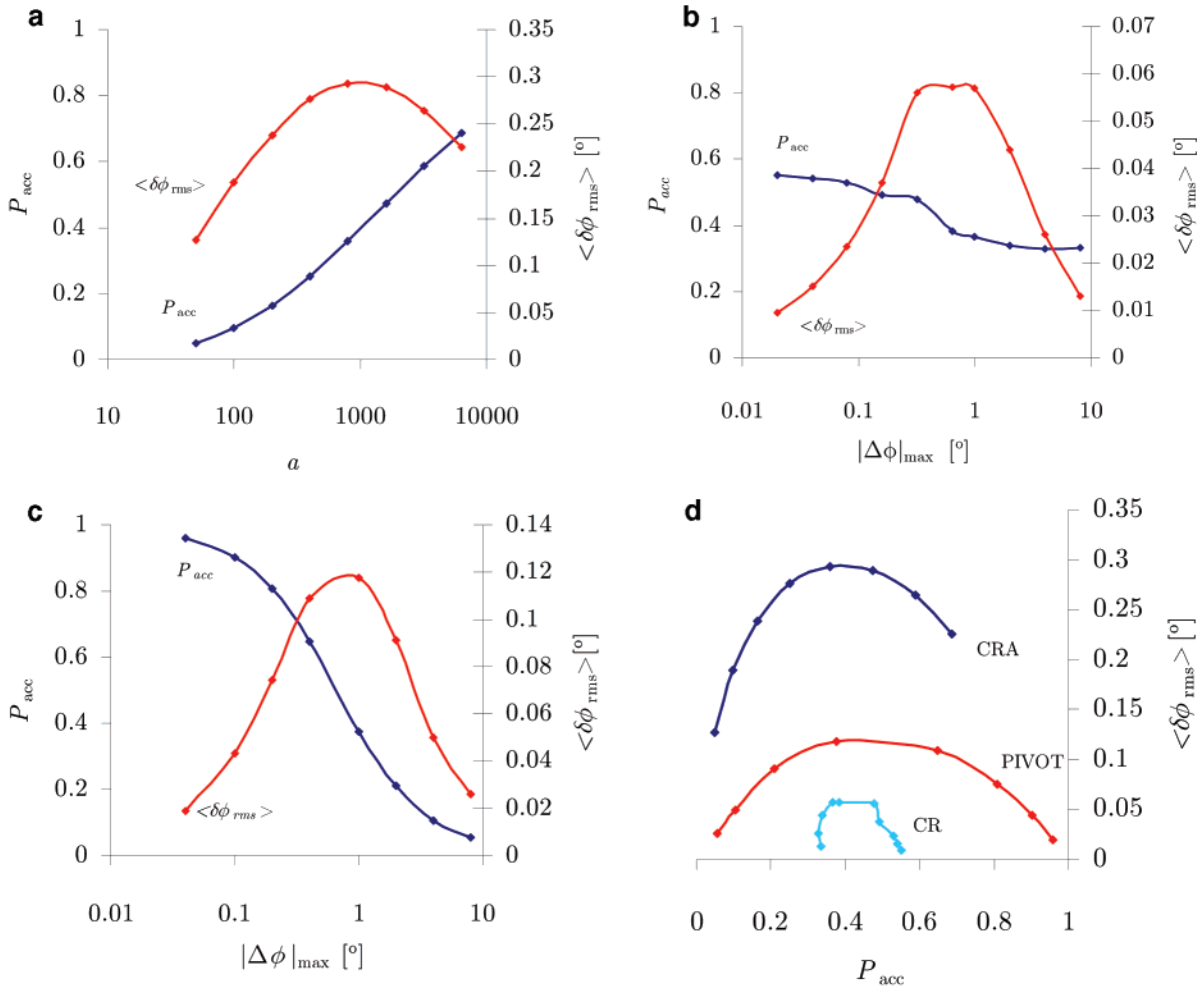

Figure 3. Average torsional step size (right scale) and acceptance rate (left scale) as a function of the parameters for the three algorithms, (a) CRA, (b) CR, and (c) PIVOT. (d) The step size versus the acceptance rate for the three algorithms. Averages are over 10^5 MC configurations.

made arbitrarily "small", and consequently there is an upper limit to the acceptance rate. The lower limit for the case of $\Delta\phi_{max} \rightarrow 180°$ originates from the fact that the acceptance rate includes accepted moves from the reverse move, including moves that accept the original conformation. Even for large prerotations, the reverse moves remain accessible, so the acceptance rate cannot drop beyond this threshold.

The behavior for the pivot algorithm is shown in Figure 3c. Although the pivot method differs significantly from both the CRA and CR algorithms, since the moves are not local, the results as a function of the maximum torsional step size exhibit similar behavior. Maximum efficiency is reached for $\Delta\phi_{max} \approx 1°$ with $\langle\delta\phi_{rms}\rangle \approx 0.11°$. For lower or higher values of $\Delta\phi_{max}$, the moves are less efficient, and the global nature of the changes leads to a fast drop of the average dihedral change for larger $\Delta\phi_{max}$. The acceptance rate shows a smooth drop from nearly 1 for small changes to about 0 for large pivot moves, as expected. However, the pivot method suffers from a systematic sampling problem along the polymer chain; since the updates are global and affect all residues beyond the perturbation point, the acceptance rates varies along the chain for a fixed $\Delta\phi_{max}$. For example, Figure 4 illustrates the average torsional angle change per 10 000 MC moves for the ϵ backbone torsion angles for the residues in one strand of the dodecamer. The closer the moved residue is to the beginning of the chain, the less efficient the pivot algorithm becomes, since large attempted conformational changes at the chain beginning will result in rejection of the new configuration. Equilibration of the system is significantly hampered by this effect.

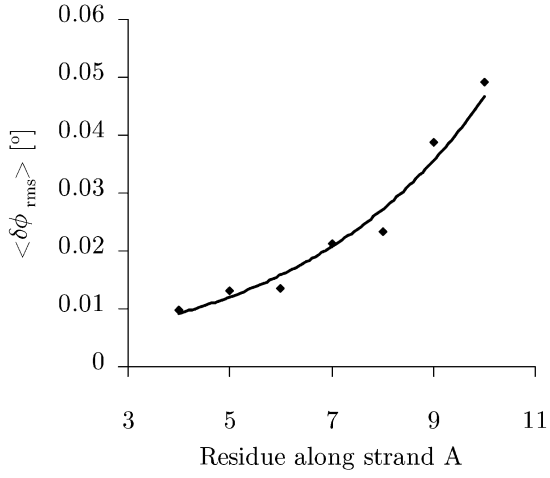

Figure 4. Average torsional step size as a function of residue position along one DNA strand for the PIVOT update.

Performance of the Three Methods. To make a direct comparison of the performance of the three methods, the average torsional step size is plotted versus the acceptance rate for all three algorithms in Figure 3d. The maximum efficiency is reached for an acceptance rate of $P_{acc} \approx 0.4$, for all methods. The superior performance of the CRA algorithm is apparent, with the optimum step size 5 times larger than for the CR

method and 2.5 times larger than for the pivot method. As an alternative approach to the described optimization, $\Delta\phi_{max}$ or a could be optimally adjusted during the simulations using the dynamically optimized Monte Carlo (DOMC) algorithm.[27] Experimenting with this approach did not lead to an improvement of the sampling, and in general, the parameters did not converge[13] due to the different nature of the local moves from more simple MC moves. In addition, since this method violates detailed balance, it could only be used during equilibration of a system.

It is of interest whether the superior per-step performance of the CRA algorithm also yields higher efficiency for longer MC runs. Especially the decay of the torsional autocorrelation function or, equivalently, the statistical inefficiency is of interest. Since much longer runs are needed for the evaluation of these quantities, no attempt was made to determine the optimum parameters of the three algorithms using correlation functions. Instead, only longer simulations using the optimum parameters, as determined above, were run. In the previous studies on polypeptides, it was found that the torsional step size and the statistical inefficiency for the torsion angles were correlated, with the parameters yielding the highest torsional step size also having the lowest correlation time.[18] The present simulation lengths were extended to 20 M configurations. Potential unraveling of the chain ends was avoided by only performing the local moves in the segment involving residues 3−10 on both strands and by not attempting chain-end moves. In addition, side-chain moves were not performed after equilibration in order to focus the statistics on the local and pivot moves for the center of a DNA duplex. The extent of the sampling is reflected in Figure 5.

Key quantities of interest are variations for the five main-chain torsion angles ϵ, ζ, α, β, and γ (δ is not moved), the total energy of the system E, the gyration radius R_G, and the root-mean-square distance (RMSD) of the atomic positions to the initial DNA structure. Figure 6 illustrates the results for the torsion angles in units of radians over the course of the simulations for a representative backbone segment, between the 4th (guanine) and 5th (adenine) bases on the first strand. There is no conformational transition for this segment during the simulations, so the efficiency of the three algorithms to explore the conformational space of this local conformational minimum is addressed. There is a clear difference in the sampling behavior for the three algorithms, with the greatest oscillations from the CRA method and the least efficient exploration achieved by the pivot method. The ranges covered for the sampled angles of this segment are shown in Figure 7. It is evident that for all main-chain angles, the CRA algorithm explores the widest space, while the ranges using the CR method are significantly smaller, and the pivot method performs somewhat worse still. The standard deviations of the sampled range for the five torsion angles are shown in Table 1. The CRA algorithm explores 20−100% larger torsional ranges than the other methods. The remaining sampled backbone torsion angles in the dodecamer show nearly identical behavior to the five described, and the results are not shown.

Autocorrelation Functions. The next step was the calculation of dynamical statistical properties. The torsional autocorrelation function is given by eq 7, where cos ϕ is a main-chain torsion angle and t is the number of MC configurations along the Markov chain (MC "time"). The inner averages are taken over different time origins, and the outer average is taken over all main-chain torsion angles. Since the autocorrelation function tends to be noisy, there is an alternative way of obtaining the correlation time, through the calculation of the statistical inefficiency in eq 8,[28] where $A = \cos \omega_i$ is the observable, $\sigma^2(A)$ is the variance of the complete run, and $\sigma^2(\langle A\rangle_n)$ is the variance of averages taken from consecutive blocks of size n. The statistical inefficiency is related to the correlation time (MC steps or time for MD) by $t_c = s_\omega/2$.[28]

$$f_{\cos\phi}(t) = \left\langle \frac{\langle \cos\phi(t)\cos\phi(0)\rangle - \langle\cos\phi(0)\rangle^2}{\langle\cos\phi(0)\cos\phi(0)\rangle - \langle\cos\phi(0)\rangle^2} \right\rangle \qquad (7)$$

$$s_\omega = \lim_{n\to\infty} \frac{n\sigma^2(\langle A\rangle_n)}{\sigma^2(A)} \qquad (8)$$

Computation of the statistical inefficiencies is complicated for systems such as the duplex DNA because there are many different relaxation processes on multiple time scales. The higher frequency fluctuations are not well separated from the many superimposed lower frequency modes. However, it was possible to determine the fluctuations of several torsion angles that oscillate around a single dominant value during the simulations

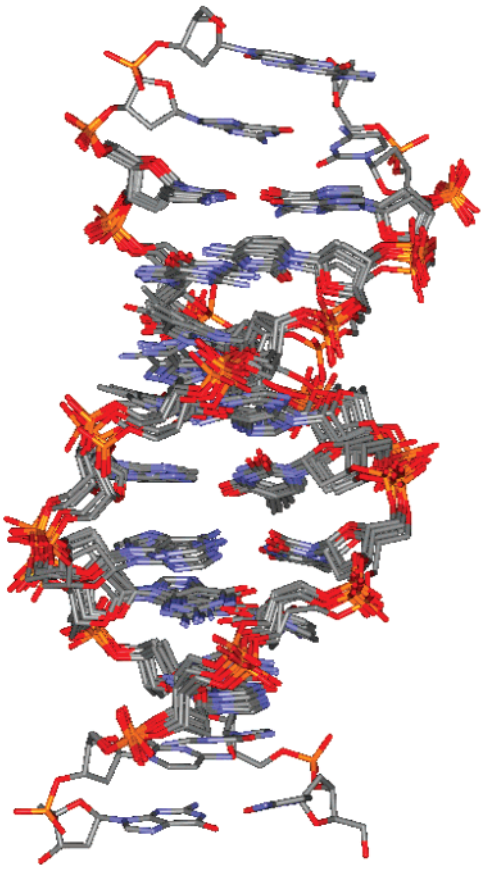

Figure 5. Overlay of representative structures during the course of MC sampling for the DNA duplex, illustrating the regions sampled with the local move methods (here CRA). The chain ends are fixed in order to measure the efficiency for the local moves in the core of the macromolecule.

TABLE 1: Standard Deviations for the Five Backbone Torsion Angles ϵ, ζ, α, β, and γ between the Fourth and the Fifth Nucleotide

	σ_ϕ [rad]				σ_ϕ [rad]		
	CRA	CR	PIVOT		CRA	CR	PIVOT
ϵ	0.214	0.171	0.108	β	0.100	0.051	0.045
ζ	0.118	0.067	0.075	γ	0.110	0.093	0.068
α	0.229	0.165	0.095				

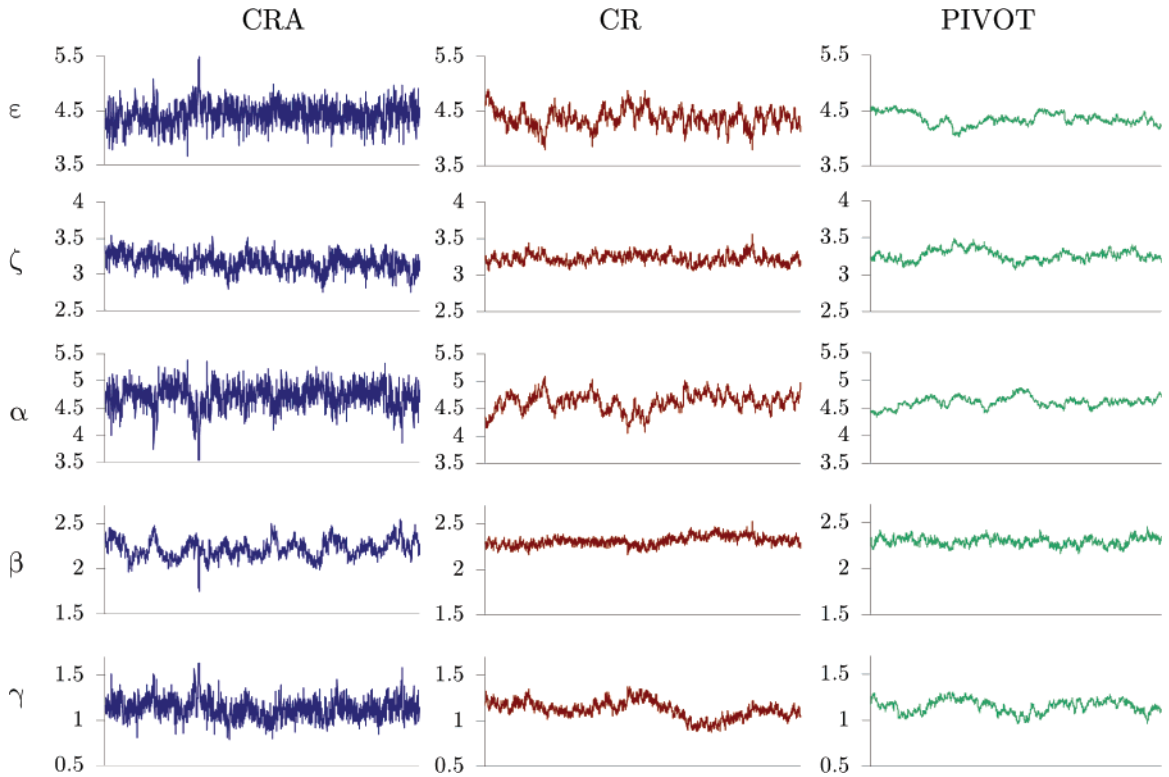

Figure 6. Fluctuation of the main-chain torsion angles (in radians) for a representative nucleotide in the dodecamer, residue 5 (adenosine). The length of each run was 20 M configurations, and the complete internal is shown as a function of MC steps, with samples taken every 0.01 M steps. The δ angle is kept fixed. The three columns provide the CRA, CR, and PIVOT results.

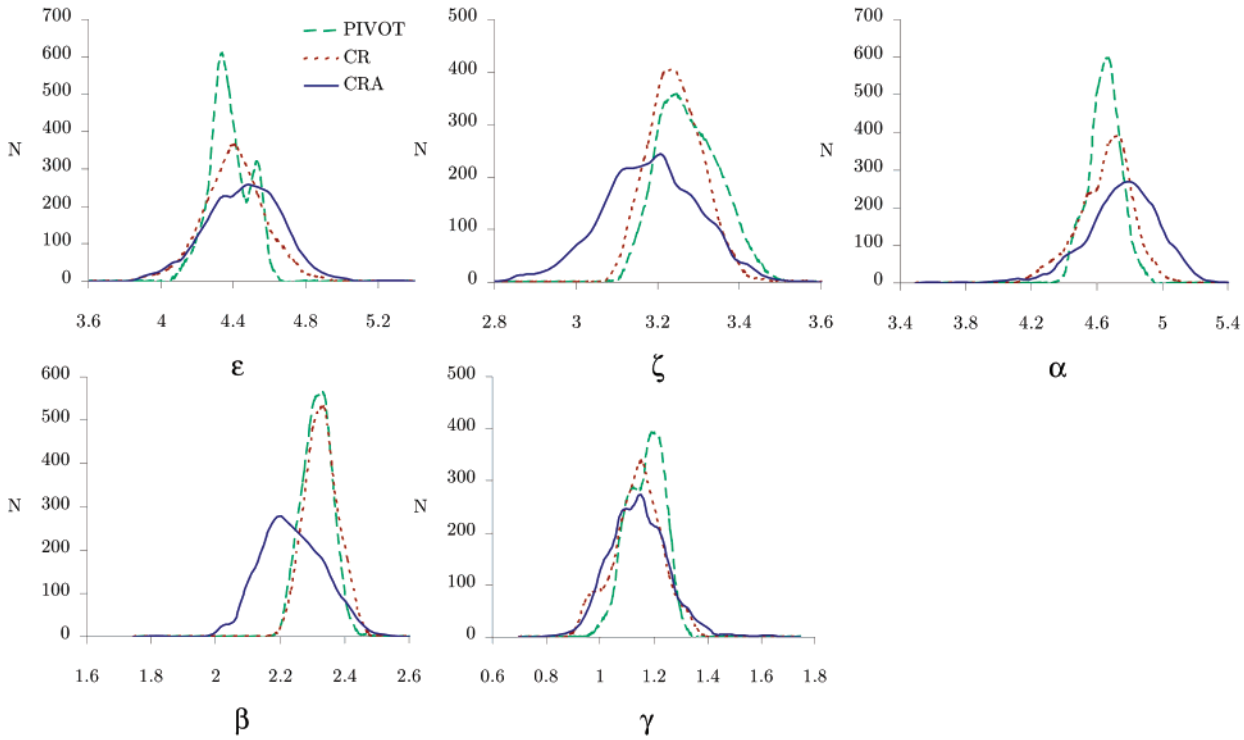

Figure 7. Histograms of the five main-chain torsion angles α, β, γ, ϵ, and ζ (in radians) for the representative nucleotide 5 of the duplex dodecamer (adenosine). Sample points were taken every 0.01 M steps, for a total of 20 M configurations. The distributions are for the CRA (solid line), CR (dotted line), and PIVOT (dashed line) methods.

and to estimate their correlation times. The resultant autocorrelation function is the average of the individual functions for the single torsions, as shown in Figure 8. The CRA method yields faster decorrelation than either the CR or pivot methods. In particular, the pivot method is much more inefficient than the local move algorithms for the main chain. The related statistical inefficiencies are then reported in Table 2 both in MC steps and in CPU time to take into account the different computational demands of the algorithms. Though the error bars are substantial, it remains clear that the sampling is better with the CRA method than the CR method; the statistical inefficiency based on the MC steps is reduced by 34%. Compared to pivot

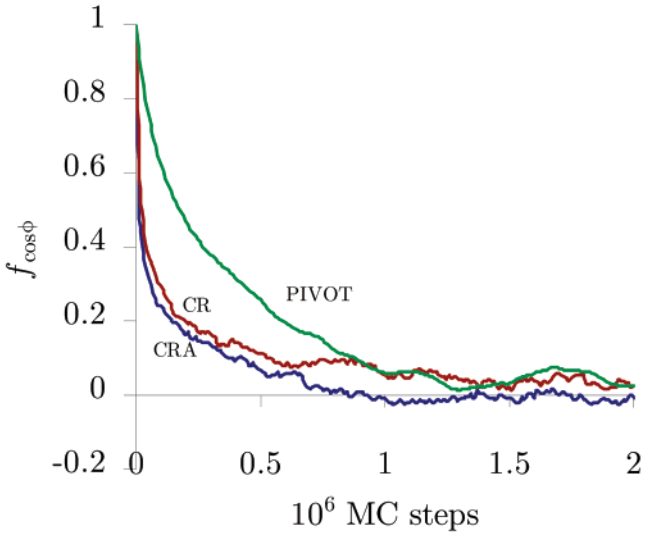

Figure 8. Autocorrelation function for the torsion angles in the center of the chain using the three algorithms with the parameters that maximize the average torsional step size.

TABLE 2: CPU Times (2.4 GHz Pentium IV) and Statistical Inefficiencies for the Five Main-Chain Torsion Angles from MC Simulations in a Vacuum

method	CPU time [s/ 10^3 MC steps]	$s_{\cos\phi}$ [10^3 MC steps]	$s_{\cos\phi}$ [10^3 s]	$\langle\sigma_\phi^{CRA}/\sigma_\phi\rangle$
CRA	24.1	150.9 ± 57.5	3.63 ± 1.38	1
CR	24.4	231.8 ± 141.9	5.65 ± 3.46	1.64 ± 0.31
PIVOT	34.8	493.8 ± 188.4	17.17 ± 6.55	1.50 ± 0.51

moves, the efficiency increase for CRA is 70%. For CPU demands, the CRA and CR methods are equivalent, while the pivot method requires 44% more computer time for the same number of MC steps. The latter increase is associated with the need to recompute more interactions on a typical pivot attempt, since more of the chain is moved. This amplifies the relative inefficiency of the pivot method as reflected in the fourth column of Table 2. Finally, as a measure of the relative sizes of the conformational space sampled in the simulation time, the average ratios of the standard deviations of the fluctuations of the individual main-chain torsion angles were calculated. The ratio is a roughly constant at 1.5−1.6 for all torsion angles when comparing CRA to the CR and PIVOT methods.

RMSD, Energy, and Radius of Gyration. Monitoring of additional observables was also used to gauge the relative efficiencies of the two local move methods. Figure 9 shows the RMSD of the instantaneous structure to the equilibrated starting structure as a function of MC steps for the two algorithms. The values are smaller than the usual 1−2 Å RMSD results that are often found in fully flexible MD simulations,[23] since the RMSD is calculated here using all atoms including the fixed residues at the chain termini (Figure 5). The comparison reveals that the CRA algorithm is sampling on average farther from the starting structure and is covering a significantly wider range of structures than the CR method. The average RMSD from the CRA calculations is about twice that from the CR method (Table 3).

Figures 10 and 11 show the fluctuations for the potential energy E and the radius of gyration of the system during the CR and CRA runs. Both observables appear converged; however, the CRA algorithm samples wider ranges with factors of ca. 1.6 for the energy and 2 for R_G (Table 3). For the energy, this is due to two reasons: first, the unfreezing of the harmonic

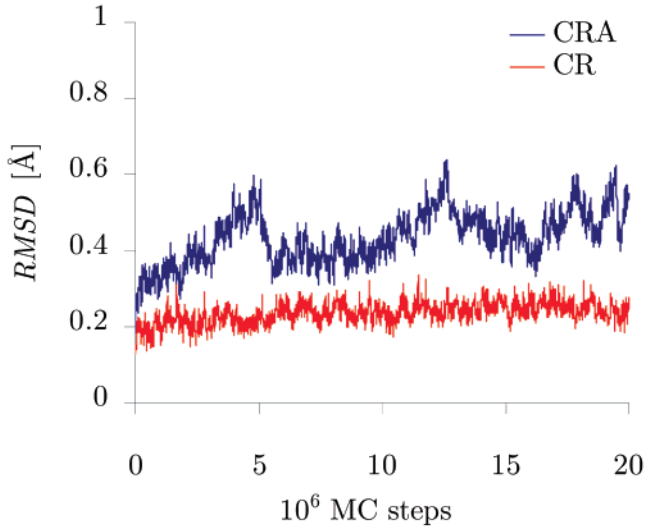

Figure 9. Plots of the RMSD to the equilibrated starting structure for the CR and CRA algorithms over 20 M step trajectories. The RMSD is calculated using all atoms of the duplex DNA including the atoms at the chain termini that remain fixed.

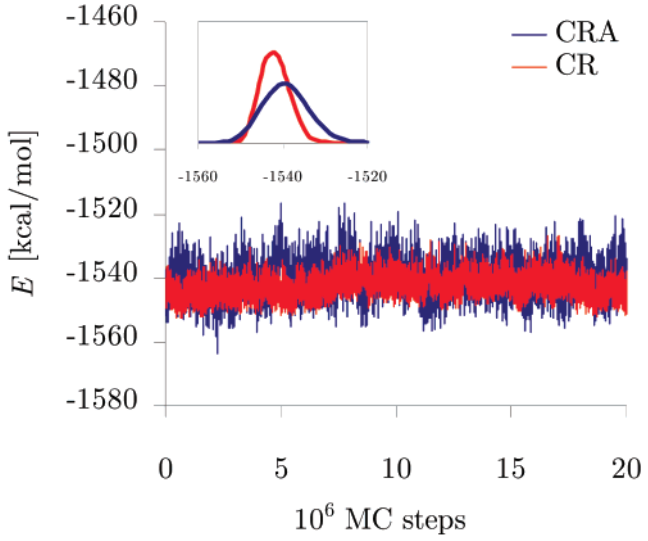

Figure 10. Plots of the total potential energy E of the DNA duplex using the CR and CRA algorithms over 20 M step trajectories. Inset: Distributions of E from two algorithms.

TABLE 3: Total Energy E, Radius of Gyration R_G, and RMSD to the Equilibrated Starting Structure and Their Standard Deviations

	CR	CRA		CR	CRA	σ_{CRA}/σ_{CR}
E [kcal/mol]	−1542.84	−1540.37	σ_E	3.38	5.41	1.60
R_G [Å]	13.341	13.372	σ_{R_G}	0.011	0.021	1.96
RMSD [Å]	0.235	0.434	σ_{RMSD}	0.029	0.069	2.38

bond angle terms contributes directly to the fluctuation of the total energy, and second, the increased flexibility allows for larger variation of accessible conformations and associated potential energies. The average energies with CRA and CR are very close (Table 3); though angle bending is not performed with CR, the angle bending energy is constant at the value for the equilibrated structure, which is essentially the same as the average angle bending energy in the CRA simulation. Figure 11 shows the fluctuation of R_G, which is a measure of the overall size of the macromolecule. Since the double helix is constrained at the chain ends, R_G is damped, but the difference in accessible range for the two methods is clearly visible.

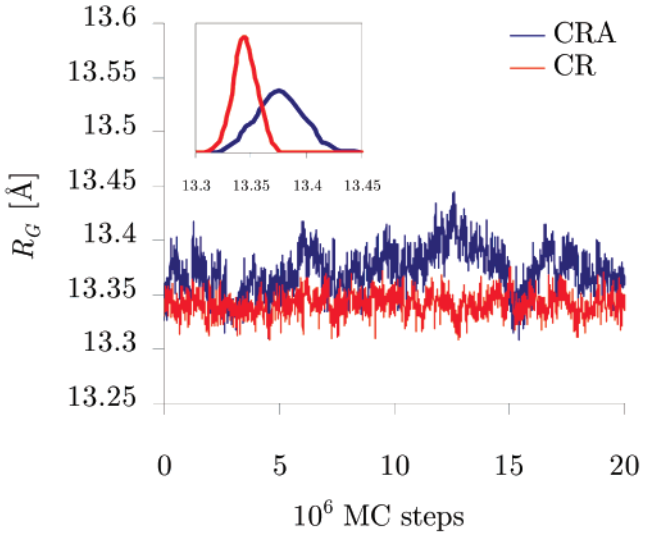

Figure 11. Plots of the radius of gyration R_G for the DNA duplex using for the CR and CRA algorithms over 20 M step trajectories. Inset: Distributions of R_G from the two algorithms.

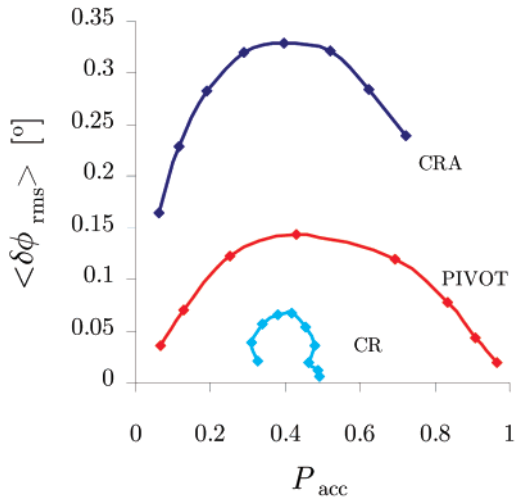

Figure 12. Average torsional step size versus acceptance rate for the three algorithms including the GB/SA solvation model. Averages are taken over 10^5 MC steps.

Results with GB/SA Solvation. To investigate the influence of solvation on the results, all simulations were repeated with the inclusion of the generalized Born (GB/SA) solvent model. This model has been widely applied because it is computationally efficient and physically reasonable. The utilized GB/SA method was the analytical model developed by Qiu, Still, and co-workers,[29] validated for simulations using the OPLS-AA force field.[21,30] First, the DNA duplex was re-equilibrated including the GB/SA model with a run of 20 M configurations. The parameters for the optimal torsional step size of the three algorithms were then redetermined using a series of simulations covering 0.1 M configurations, as described above for the vacuum simulations. Figure 12 shows the resultant average torsional step size plotted versus the acceptance rate for all three algorithms. Again, the maximum efficiency is reached for an acceptance rate of $P_{acc} \approx 0.4$ for all methods. The average torsional step size is increased for all methods, with the optimal value of $\langle \delta\phi_{rms} \rangle$ larger by 13−22%. This indicates a somewhat less rugged energy landscape for the solvated system. Calculation of the torsional statistical inefficiencies was repeated with runs of 20 M configurations, and the results are shown in Table 4. The results are similar to the vacuum case (Table 2), and the relative sampling efficiency of the algorithms is unaltered. This indicates that solvation does not favor any one algorithm over the other and that the relative performance of the sampling schemes is independent of whether the system is GB/SA solvated. The current implementation, which uses no cutoffs for energy evaluations, has received little optimization for speed, and the required computer time doubles for the CRA calculations upon adding GB/SA solvation.

TABLE 4: CPU Times (2.4 GHz Pentium IV) and Statistical Inefficiencies for the Five Main-Chain Torsion Angles from MC Simulations Including GB/SA Solvation

	CPU time [s/ 10^3 MC steps]	$s_{\cos\phi}$ [10^3 MC steps]	$s_{\cos\phi}$ [10^3 s]	$\langle \sigma_\phi^{CRA}/\sigma_\phi \rangle$
CRA	57.0	173.8 ± 88.3	9.90 ± 5.0	1
CR	57.4	239.0 ± 75.6	13.71 ± 4.3	1.53 ± 0.24
PIVOT	133.5	334.0 ± 138.4	44.59 ± 18.5	1.39 ± 0.33

Conclusion

An enhanced concerted rotation algorithm that includes flexible dihedral and bond angles (CRA) has been extended for Monte Carlo simulations of polynucleotides. The use of concerted rotation algorithms for this class of polymers has only recently been proposed[13] and required a generalization of the original equations[6] to systems in which the fixed internal coordinates are nonplanar in their ideal geometries. Previous applications of concerted rotation algorithms have revealed inefficiency associated with the lack of conformational flexibility by sampling only the main-chain dihedral angles.[16,17] This problem is overcome by including flexible bond angles in the CRA method, which leads to significantly improved sampling for polypeptides compared to the original concerted rotation approach.[18,21] The results of this study show a similar increase in efficiency for the more complicated case of nucleic acids. The efficiency of the previous conrot algorithms is adversely affected by the necessity to consider all solutions to the chain closure equations for generating a new configuration. Even for small prerotations, the change in the remaining dihedral angles is usually large, leading to a low probability of acceptance for polymers with bulky side chains. There has been a recent attempt to overcome this problem by choosing only solutions that are close to the original conformation.[22] An additional inefficiency is caused by having to consider solutions of the reverse move, and if such a move is accepted, it involves only 6 dihedrals. The CRA methods circumvent these complications and yield significantly enhanced sampling by all measures. The least acceptable alternative is the simple pivot sampling method; this procedure is both much more demanding for computer time and inefficient than either local move technique. The relative performance of the three algorithms was found to be unaffected by the inclusion of GB/SA solvation. Future CRA-based studies can feature applications including biomolecular structure prediction[21] and technical comparisons with additional MC sampling methods[1−5] and molecular dynamics.

Acknowledgment. Gratitude is expressed to Dr. Julian Tirado-Rives for assistance and discussions and to the National Science Foundation (CHE0130996) and the National Institutes of Health (GM032136) for support of this work.

References and Notes

(1) Escobedo, F. A.; de Pablo, J. J. *J. Chem. Phys.* **1995**, *102*, 2636−2652.

(2) Vendruscolo, M. *J. Chem. Phys.* **1997**, *106*, 2970−2976.

(3) Chen, Z.; Escobedo, F. A. *J. Chem. Phys.* **2000**, *113*, 11382−11392.
(4) Wick, C. D.; Siepmann, J. I. *Macromolecules* **2000**, *33*, 7207−7218.
(5) Uhlherr, A. *Macromolecules* **2000**, *33*, 1351−1360.
(6) Go, N.; Scheraga, H. A. *Macromolecules* **1970**, *3*, 178.
(7) Wakana, H.; Wako, H.; Saito, N. *Int. J. Pept. Protein Res.* **1984**, *23*, 315−323.
(8) Dodd, L. R.; Boone, T. D.; Theodorou, D. N. *Mol. Phys.* **1993**, *78*, 961−996.
(9) Knapp, E. W. *J. Comput. Chem.* **1992**, *13*, 793−798.
(10) Knapp, E. W.; Irgensdefregger, A. *J. Comput. Chem.* **1993**, *14*, 19−29.
(11) Wu, M. G.; Deem, M. W. *Mol. Phys.* **1999**, *97*, 559−580.
(12) Deem, M. W.; Bader, J. S. *Mol. Phys.* **1996**, *87*, 1245−1260.
(13) Dinner, A. R. *J. Comput. Chem.* **2000**, *21*, 1132−1144.
(14) Hoffmann, D.; Knapp, E. W. *Eur. Biophys. J.* **1996**, *24*, 387−403.
(15) Hoffmann, D.; Knapp, E. W. *J. Phys. Chem. B* **1997**, *101*, 6734−6740.
(16) Bruccoleri, R. E.; Karplus, M. *Macromolecules* **1985**, *18*, 2767−2773.
(17) Sartori, F.; Melchers, B.; Bottcher, H.; Knapp, E. W. *J. Chem. Phys.* **1998**, *108*, 8264−8276.
(18) Ulmschneider, J. P.; Jorgensen, W. L. *J. Chem. Phys.* **2003**, *118*, 4261−4271.
(19) Favrin, G.; Irback, A.; Sjunnesson, F. *J. Chem. Phys.* **2001**, *114*, 8154−8158.
(20) Bashford, D.; Case, D. A. *Annu. Rev. Phys. Chem.* **2000**, *51*, 129−152.
(21) Ulmschneider, J. P.; Jorgensen, W. L. *J. Am. Chem. Soc.* **2004**, *126*, 1849−1857.
(22) Mezei, M. *J. Chem. Phys.* **2003**, *118*, 3874−3879.
(23) Beveridge, D. L.; McConnell, K. J. *Curr. Opin. Struct. Biol.* **2000**, *10*, 182−196.
(24) Drew, H. R.; Wing, R. M.; Takano, T.; Broka, C.; Tanaka, S.; Itakura, K.; Dickerson, R. E. *Proc. Natl. Acad. Sci. U.S.A.* **1981**, *78*, 2179−2183.
(25) Jorgensen, W. L.; Tirado-Rives, J. *MCPRO*, version 1.68; Yale University: New Haven, CT, 2002.
(26) Jorgensen, W. L.; Maxwell, D. S.; Tirado-Rives, J. *J. Am. Chem. Soc.* **1996**, *118*, 11225−11236.
(27) Bouzida, D.; Kumar, S.; Swendsen, R. H. *Phys. Rev. A* **1992**, *45*, 8894−8901.
(28) Allen, M. P.; Tildesley, D. J. *Computer Simulation of Liquids*; Oxford University Press: New York, 1989.
(29) Qiu, D.; Shenkin, P. S.; Hollinger, F. P.; Still, W. C. *J. Phys. Chem. A* **1997**, *101*, 3005−3014.
(30) Jorgensen, W. L.; Ulmschneider, J. P.; Tirado-Rives, J. *J. Phys. Chem. B*, in press.