SIAM J. Sci. Comput.
Vol. 30, No. 4, pp. 2055–2083

DYNAMICAL SPATIAL WARPING: A NOVEL METHOD FOR THE CONFORMATIONAL SAMPLING OF BIOPHYSICAL STRUCTURE[*]

PETER MINARY[†], MARK E. TUCKERMAN[‡], AND GLENN J. MARTYNA[§]

Abstract. The difficulties encountered in sampling of systems with rough energy landscapes using present methodology significantly limit the impact of simulation on molecular biology, in particular protein folding and design. Here, we present a major methodological development based on a promising new technique, the reference potential spatial warping algorithm (REPSWA) [Z. Zhu et al., *Phys. Rev. Lett.*, 88 (2002), pp. 100201–100204], and present applications to several realistic systems. REPSWA works by introducing a variable transformation in the classical partition function that reduces the volume of phase space associated with a priori known barrier regions while increasing that associated with attractive basins. In this way, the partition function is preserved so that enhanced sampling is achieved without the need for reweighting phase-space averages. Here, a new class of transformations, designed to overcome the barriers induced by intermolecular/nonbonded interactions, whose locations are *not* known a priori, is introduced. The new transformations are designed to work in synergy with transformations originally introduced for overcoming intramolecular barriers. The new transformation adapts to the fluctuating local environment and is able to handle barriers that arise "on the fly." Thus, the new method is referred to as dynamic contact REPSWA (DC-REPSWA). In addition, combining hybrid Monte Carlo (HMC) with DC-REPSWA allows more aggressive sampling to take place. The combined DC-REPSWA-HMC method and its variants are shown to substantially enhance conformational sampling in long molecular chains composed of interacting single beads and beads with branches. The latter topologies characterize the united residue and united side chain representation of protein structures.

Key words. sampling, nonlinear transformation, Monte Carlo

AMS subject classifications. 15A15, 15A09, 15A23

DOI. 10.1137/070686706

1. Introduction.

Monte Carlo (MC) and molecular dynamics (MD) atomistic simulation methods have profoundly increased the understanding of nanoscale phenomena in molecular biology. However, one of the challenges in computational molecular biology is to develop techniques capable of determining the conformational equilibria of systems described by complex, multidimensional, rough energy landscapes as the drive to advance investigations in systems biology moves forward. The term "rough" indicates that energy surfaces have a large number of minima separated by barriers, most of which are high compared to $k_{\mathrm{B}}T$, where T is the temperature and k_{B} is Boltzmann's constant.

Following early work [9], the analytical all-atom energy function of an entire biological molecule includes harmonic potentials that keep bonded atoms and angles between neighboring bonds together, periodic potentials for torsion angles, and Lennard–Jones and Coulomb potentials between nonbonded atoms. Extensive parametrization of such all-atom potential functions leads to modern all-atom force fields

[*] Received by the editors March 28, 2007; accepted for publication (in revised form) September 14, 2007; published electronically May 14, 2008.

http://www.siam.org/journals/sisc/30-4/68670.html

[†] Department of Structural Biology, Stanford University, CA, 94305 (peter.minary@stanford.edu). The research of this author was supported by NIH–GM41455.

[‡] Department of Chemistry and Courant Institute of Mathematical Sciences, New York University, New York, NY 10003 (mark.tuckerman@nyu.edu). The research of this author was supported by NSF CHE-0310107, NSF CHE-0121375, NSF CHE-0704036, and PRF 45485-AC5.

[§] IBM Physical Sciences Division, P.O. Box 218, Yorktown Heights, NY 10598 (martyna@us.ibm.com). The research of this author was supported by funding from NSF and IBM.

[1, 14] that are commonly used in computational chemistry for modeling nonreactive chemical events. If the nonbonded terms in the original energy function [9] are replaced by knowledge-based potentials [24], the resulting energy function can be used for structural prediction or refinement [23], which are among the most significant challenges in computational biology.

As any large conformational rearrangement is due to changes in torsion angles, algorithms that enhance sampling in torsion angle space hold the key to answering some of the most difficult questions of computational molecular biology.

All the aforementioned potential functions can be termed "rough" with respect to both Cartesian and torsion angle coordinates, and the primary difficulty that arises in sampling these rough energy landscapes is the low probability of crossing the barriers that separate important minima or basins of attraction on the surface over simulation times that can be reached using current computational resources.

In order to alleviate these problems, several recently developed methods have been proposed. The so-called puddle skimming [21] approach enhances conformational sampling via a simple linear modification of the energy landscape $V(q)$. As a result, the modified energy surface $V^{\star}(q)$ is flooded up to an a priori chosen energy value, the *boosting energy*, E_B. The Boltzmann distribution on the original energy surface can be obtained after proper reweighting of conformations obtained by sampling on $V^{\star}(q)$. Further improvement has been reached in puddle jumping [20] using several distinct energy surfaces $V^{\star}(q)$ with different *boosting energies*, so that high and low energy regions are sampled with equal accuracy. Based on a concept similar to such biasing potentials, hyperdynamics [27] accelerates the occurrence of a sequence of events that would take place only on a time scale generally inaccessible with conventional MD. In addition, sampling via MD could be further accelerated by particle mass rescaling, which partially removes stiff high frequency motions so that a larger time step can be used. While the above methods involve only simple or no modifications of the potential energy surface $V(q)$, popular methods based on nonlinear modifications were also proposed. For example, in MC minimization [10] or basin-hopping techniques [29], a modification $\tilde{V}(q) = \min\{V(q)\}$ leads to a simple energy surface of interpenetrating staircases and plateaus that leaves the global minimum and the relative importance of local minima unaltered. This simplified landscape $\tilde{V}(q)$ can be sampled more effectively because energy barriers between neighboring energy basins are removed.

Two of the aforementioned methods, puddle skimming and puddle jumping, modify the partition function significantly and can only treat multiple barriers effectively when EB, varied slowly from "off" to "on" within a replica exchange protocol (puddle skimming) or the sequence of *boosting energies* $E_{\mathrm{B}}(1), E_{\mathrm{B}}(2), \ldots, E_{\mathrm{B}}(\mathrm{n})$ are chosen (puddle jumping) based on the topology of the energy surface which is usually not known a priori. Furthermore, as the reweighting protocol of puddle skimming/jumping uses simple averaging of the Boltzmann weight of the difference between the true and boosted potential, it fails for large dimensional systems. The hyperdynamics method assumes that transition state theory can be applied and that the systems traverses phase space via one barrier crossing event at a time; it has the correct phase space weighting. Hyperdynamics has been successfully applied to important systems that admit such a simplified dynamics. However, introducing boosts along many coordinates would lead to failure of the required reweighting procedure. Finally, it is important to mention that the scope of any *basin hoping* approach is only limited to locating local minima, and it is not capable of delivering information about conformational probability distributions.

Most recently, a novel and promising approach to addressing the conformational sampling problem was introduced [31], in which nonlinear variable transformations were employed to reformulate, exactly, the classical statistical mechanical configurational partition function. Note that, unlike the previous linear transformation based sampling methods [21, 20, 27], the reference potential spatial warping algorithm (REPSWA) could analytically eliminate all a priori known built-in torsional barriers present in conventional atom force fields [1, 14]. However, similar to other sampling methods [21, 20], REPSWA transformations [31] were based on explicit knowledge of the locations of the barriers on the potential energy surface, and improving the method requires a scheme for eliminating barriers that arise from intermolecular or solvent contacts.

Here, we present a substantial advance in the REPSWA approach to treating conformation-dependent or contact barriers, i.e., barriers whose locations are not known a priori but arise dynamically in the course of the sampling. *The efficacy of the new dynamic contact REPSWA (DC-REPSWA) lies in the fact that the transformations are, by construction, conformational or contact-dependent and adapt to the changing local environment while still preserving the partition function.* The computational cost of the method remains order N if only short-range nonbonded interactions are considered. Applications to one-dimensional problems, simple and branched chain molecules, and, finally, the folding of a model β-barrel protein are presented. In general, a single DC-REPSWA trajectory is shown to sample the conformational space with orders of magnitude greater efficiency than standard hybrid Monte Carlo (HMC) [2] and an order of magnitude greater efficiency than another popular technique, parallel tempering (PT) [3, 15]. It is stressed that by combining DC-REPSWA with other techniques such as PT [3, 15], puddle jumping [20], and basin-hopping [29], yet more powerful approaches will result, by which the “ab initio” conformational search problem can potentially be solved. Finally, note that further *supplementary information* is available on the method; however, all implementation details are outlined in this paper.

2. One-dimensional example.

2.1. Analytical formulation. In order to illustrate the basic principle behind the REPSWA technique, consider the simple example of a particle, with momentum p, mass m, and coordinate x in a one-dimensional potential $V(x)$ possessing various energy minima separated by large energy barriers in the domain $[a_i, a_f]$. The canonical partition function for this system is

$$Q(\beta) = \frac{1}{h}\int dp \int dx \, \exp\left[-\beta\left[\frac{p^2}{2m} + V(x)\right]\right], \tag{2.1}$$

where h is Planck’s constant and $\beta = 1/k_BT$. The spatial probability distribution function $P(x) = \exp[-\beta V(x)]/Q(\beta)$ could, in principle, be sampled by MD using a Hamiltonian, $H = p^2/2m + V(x)$, coupled to a thermostat or by MC. However, if the barriers of $V(x)$ are high compared to $k_\mathrm{B}T$, barrier crossing will be rare, and the probability distribution will not be adequately sampled without resorting to extremely long simulation lengths.

If we are solely interested in equilibrium properties, then we may exploit the fact that x in (2.1) is just an integration variable and transform to a new variable $u = f(x)$ without altering the partition function. If $f(x)$ is chosen such that a unique inverse

$x = f^{-1}(u) = g(u)$ exists, then substituting the transformation into (2.1) gives

$$Q(\beta) = \frac{1}{h}\int dp \int du \, \exp\left\{-\beta\left[\frac{p^2}{2m} + V_{\text{eff}}(u)\right]\right\}, \tag{2.2}$$

where $V_{\text{eff}}(u) = V(g(u)) - k_{\text{B}}T\ln(g'(u))$ is an effective potential for u that results from performing the transformation. The form of this effective potential suggests that a reference potential $V_{\text{ref}}(x)$ can be exploited to effect the desired spatial warping. Consider choosing a transformation of the form

$$u = f(x) = cx_0 + c\int_{x_0}^{x} dy \, \exp\left[-\beta V_{\text{ref}}(y)\right], \tag{2.3}$$

where c is a constant. Substituting (2.3) into (2.2) gives an effective potential of the form $V_{\text{eff}}(u) = V(g(u)) - V_{\text{ref}}(g(u))$. Thus, if $V_{\text{ref}}(x)$ is chosen to be equal to $V(x)$ in the nonlinear domain $D_n = \{x : x_0 = a_i \le x \le a_f\}$, corresponding to a barrier region of $V(x)$, and 0 for $x \notin D_n$, then the difference potential $V(g(u)) - V_{\text{ref}}(g(u))$ to which u is subject would have no barrier in D_n. Thus, conformational space can be easily and efficiently sampled by thermostatted MD or HMC starting from $\tilde{H} = p^2/2m + V_{\text{eff}}(u)$ as a Hamiltonian. Moreover, conformations obtained in "u-space" can be converted back to physical conformations in "x-space" by inverting (2.3) to give $x = g(u)$. In addition, the reference potential is always chosen in such a way that the integrand of (2.3) is always positive, so that (2.3) describes a strictly monotonically increasing function. Thus, the inverse $g(u)$ exists.

2.2. Numerical implementation.

In order to evaluate the integral in (2.3), $\exp[-\beta V_{\text{ref}}(x)]$ is expanded in terms of orthogonal polynomials; here, we use Legendre polynomials $P_l(x)$ which are orthogonal on $[-1, 1]$ and satisfy the relation

$$(P_i, P_j) = \int_{-1}^{1} P_i(x)P_j(x)dx = \frac{2}{2i+1}\delta_{ij}, \tag{2.4}$$

where δ_{ij} is 1 if $i = j$ and 0 otherwise. Anticipating the application of the REPSWA transformations to dihedral angles in long chain molecules, an expansion

$$\exp\left[-\beta\tilde{V}_{\text{ref}}(x; N_p)\right] = \sum_{l=0}^{N_p} C_l P_l(h(x)) \tag{2.5}$$

is employed, where $h(x)$ is a linear transformation that maps D_n onto the interval $[-1, 1]$ and has the following form:

$$h(x) = \left(x - \frac{a_i + a_f}{2}\right)\left(\frac{2}{a_f - a_i}\right). \tag{2.6}$$

In (2.5) N_p has a finite practical value between 64 and 96, and each coefficient C_l is obtained via the orthogonality relation, (2.4), which gives

$$C_l = \frac{1}{(P_l, P_l)}\left(\exp\left[-\beta V_{\text{ref}}\right], P_l\right). \tag{2.7}$$

It is not necessary to evaluate the above integral over the exponential of the true reference potential analytically to determine C_l. It is sufficient to use an (N_p+1)-point

Gaussian quadrature integration, which is exact for polynomials of degree $2N_p + 1$:

$$C_l(N_p) \equiv \frac{2l+1}{2} \sum_{k=0}^{N_p} w_k \exp\left[-\beta V_{\mathrm{ref}}(x_k)\right] P_l(y_k), \tag{2.8}$$

where $(w_k, y_k : k = 0, \ldots, N_p)$ are the Gauss–Legendre weights and nodes and $(x_k = h^{-1}(y_k) : k = 0, \ldots, N_p)$ are the Gauss–Legendre nodes mapped to D_n via inverting the linear relation $h(x)$. Here, C_0 is adjusted to ensure that $\tilde{V}_{\mathrm{ref}}(x; N_p)$ is strictly real valued. Note that the identity operator is defined exactly on the finite real space and the finite function space, and that the set, the N_p Legendre polynomials on the $N_p + 1$ Gaussian quadrature points, is said to form a discrete variable representation (DVR). The above expansion leads to a transformation $u = \tilde{f}(x)$, which also exactly preserves the partition function. Given the expansion and a practically convenient basis size, the transformation integral of (2.3) over D_n can be easily evaluated by applying the recurrence relation $P'_{n+1}(x) - P'_{n-1}(x) = (2n+1)P_n(\mathrm{x})$ and by using the fact that $P_i(h(x_0))$ will vanish for all i. The transformation is given by

$$\begin{aligned} u_n(x; N_p) &= cx_0 + c\int_{x_0}^{x} dy\ \exp\left[-\beta\tilde{V}_{\mathrm{ref}}(y)\right] = cx_0 + c\sum_{l=0}^{N_p} C_l \int_{x_0}^{x} dy\ P_l(h(y)) \\ &= cx_0 + cC_0P_0(x - x_0) + c\sum_{l=1}^{N_p} C_l \int_{x_0}^{x} dy\ \frac{1}{h'(y)} \frac{P'_{l+1}(h(y)) - P'_{l-1}(h(y))}{2l+1} \\ &= cx_0 + cC_0P_0(x - x_0) \\ &\quad + c\sum_{l=1}^{N_p} \left(\frac{C_l}{2l+1}\right)\left(\frac{a_f - a_i}{2}\right)\left(P_{l+1}(h(x)) - P_{l-1}(h(x))\right). \end{aligned} \tag{2.9}$$

In general, (2.3) is defined on the domain $D_t = \{x : b_i \le x \le b_f\}$, embedding successively connected domains $D_{l_1} = \{x : b_i \le x < a_i\}$, D_n, and $D_{l_2} = \{x : a_f < x \le b_f\}$. Thus, u can be practically evaluated in the extended domain, D_t, by considering all partitions individually:

$$\tilde{u}(x; N_p) = \begin{cases} cx, & x \in D_{l_1}, \\ u_n(x; N_p), & x \in D_n, \\ cx + c((a_f - a_i)(C_0 - 1)), & x \in D_{l_2}. \end{cases} \tag{2.10}$$

In addition, c can be chosen based on a preferred boundary condition. For the case of torsional dihedral transformations $D_t = \{x : 0 < x \le 2\pi\}$, and the boundary condition is $u(2\pi) = 2\pi$, so that $c = 2\pi/(2\pi + (a_f - a_i)(C_0 - 1))$.

It is useful to illustrate the properties of the transformation on simple one-dimensional systems. Figure 2.1(a) depicts the quartic double well potential

$$V(x) = \frac{\epsilon_0}{a^4}\left[x^2 - a^2\right]^2, \tag{2.11}$$

the reference potential $V_{\mathrm{ref}}(x)$ and the nonlinear transformation $u = \tilde{f}(x)$, defined by (2.10). Here we have $D_t = \{x : -\infty \le x \le \infty\}$, $D_n = \{x : -1 \le x \le 1\}$, and the scaling factor, $c = 1$. As expected, u changes very little in the barrier region. In

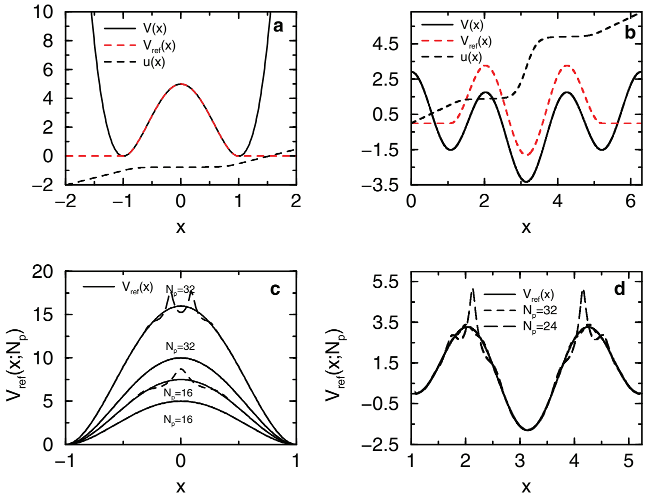

FIG. 2.1. (a) *Potential,* $V(x) = (\epsilon_0/a^4)(x^2 - a^2)^2$*, for the quartic double well with* $\epsilon_0 = 5$ *(solid black) along with the reference potential,* $V_{\rm ref}(x)$ *(dashed red), and* $u(x)$ *(dashed black) defined by* (2.10)*. Here* $b_i = -\infty$*,* $a_i = -1$*,* $a_f = 1$*, and* $b_f = \infty$*.* (b) *Periodic potential,* $V(x) = \sum_{i=1}^{3} W_i \cos(a_i x + d_i)$ *(solid black), with* $\{W_1, W_2, W_3\} = \{-1.0, 0.2, -2.1\}$*,* $\{a_1, a_2, a_3\} = \{1, 2, 3\}$*, and* $d_1 = d_2 = d_3 = \pi$*. The reference potential,* $V_{\rm ref}$ *(dashed red), is shifted, so that* $V_{\rm ref}(a_i) = V_{\rm ref}(a_f) = 0$*.* $u(x)$ *(dashed black) is plotted in the domain defined by* $b_i = 0$*,* $a_i = \pi/3$*,* $a_f = 5\pi/3$*, and* $b_f = 2\pi$*.* (c) *Expanded reference potentials,* $\tilde{V}_{\rm ref}$*, for the quartic double well of* (a) *with different choice of* $\epsilon_0 = 5, 7.5, 10, 16$ *and varying number of Legendre polynomials,* $N_p = 16$ *(dashed) and* $N_p = 32$ *(long dashed). The original reference potential,* $V_{\rm ref}(x)$ *(solid), is plotted, too.* (d) *Expanded reference potentials,* $\tilde{V}_{\rm ref}$*, for the potential introduced in* (b)*.* $N_p = 32$ *(dashed) and* $N_p = 24$ *(long dashed). For comparison, the true reference potential,* $V_{\rm ref}(x)$ *(solid), is depicted as well.*

addition, a small change in u gives rise to a large change in x for high values of ϵ_0, so that a particle moving through the u-space spends little time in the barrier region. Another example is illustrated in Figure 2.1(b), which is a periodic potential of the form

$$V(x) = \sum_{i=1}^{3} W_i \, \cos(a_i \, x + d_i). \tag{2.12}$$

Here, $D_t = \{x : 0 < x \leq 2\pi\}$ and $D_n = \{x : 1/3\pi \leq x \leq 5/3\pi\}$, so that c is determined from the boundary condition $u(2\pi) = 2\pi$. Furthermore, in order to ensure that $V_{\rm ref}(x)$ is continuous over D_t, its definition in D_n is $V_{\rm ref}(x) \leftarrow V_{\rm ref} + C$, where the constant C is chosen such that $V_{\rm ref}(a_i) = V_{\rm ref}(a_f) = 0$. Second, the expanded reference potential $\tilde{V}_{\rm ref}(x; N_p)$ is compared to the true potential or $V_{\rm ref}(x)$ as a function of the number of Legendre polynomials, N_p, and ϵ_0 in Figures 2.1(c) and 2.1(d). It must be emphasized that the partition function remains invariant even at low polynomial orders even though $\tilde{V}_{\rm ref}(x; N_p)$ might not be a good approximation to $V_{\rm ref}(x)$.

Given the numerical implementation of the nonlinear transformation $u = \tilde{f}(x)$, u-space characterized by the effective potential $\tilde{V}_{\rm eff}(g(u))$ can be efficiently explored via various commonly used sampling or optimization methods such as MC, MC with minimization, MD, and HMC algorithms. Most of these methods, however, require the calculation of the local gradients or forces:

$$F_u = -\frac{dV_{\text{eff}}(g(u))}{du} = -\frac{d}{du}\left[V(g(u)) - V_{\text{ref}}(g(u))\right] \tag{2.13}$$
$$= -\frac{d}{d[g(u)]}\left[V(g(u)) - V_{\text{ref}}(g(u))\right]\ \frac{d[g(u)]}{du} = -\left[\frac{dV(x)}{dx} - \frac{dV_{\text{ref}}(x)}{dx}\right]\left[\frac{du(x)}{dx}\right]^{-1}.$$

It can be shown how to obtain this quantity by exploiting the Legendre polynomial basis defined over D_n:

$$\tilde{F}_u^n(x; N_p) = -\left[\frac{d\tilde{u}(x; N_p)}{dx}\right]^{-1}\left[\frac{dV(x)}{dx} - \frac{d\tilde{V}_{\text{ref}}(x; N_p)}{dx}\right] \tag{2.14}$$
$$= \frac{1}{c\ \exp(-\beta\ \tilde{V}_{\text{ref}}(x; N_p))}\left[F_x - \frac{d/dx\ \left[\exp(-\beta\ \tilde{V}_{\text{ref}}(x; N_p))\right]}{\beta\ \exp(-\beta\ \tilde{V}_{\text{ref}}(x; N_p))}\right].$$

Here, we have used the fact that $\tilde{V}_{\text{ref}}(x; N_p) = -1/\beta\ \ln\ \exp(-\beta\ \tilde{V}_{\text{ref}}(x; N_p))$. The practical evaluation of (2.14) can be easily implemented by defining two functions, exp_beta_Vref$(x; N_p) = \exp(-\beta\ \tilde{V}_{\text{ref}}(x; N_p))$ and exp_beta_Vref_prime$(x; N_p) = d/dx\ [\exp(-\beta\ \tilde{V}_{\text{ref}}(x; N_p))]$, using the expansion in (2.5) and its analytical derivative, respectively. Note that the expansion is not valid outside D_n, and the extended definition of the gradient should be used.

$$\tilde{F}_u(x; N_p) = \begin{cases} F_x/c, & x \in D_{l_1}, D_{l_2}, \\ \tilde{F}_u^n(x; N_p), & x \in D_n. \end{cases} \tag{2.15}$$

It is clear from (2.15) that evaluating $\tilde{F}_u(x; N_p)$ assumes the knowledge of $dV(x)/dx$, which implies that every force calculation evaluated at a new position in u-space should be preceded by an $x = \tilde{g}(u)$ transformation executed by inverting the relation defined by (2.10). Since the linear regions involving D_{l_1} and D_{l_2} are trivial, we now show how to invert (2.10) over domain D_n. Assuming that the three variables u, x_0, and u_0 are initially known and $u_0 \equiv \tilde{u}(x_0, N_p)$, u can be approximated as

$$u = u_0 + \left.\frac{d\tilde{u}(x; N_p)}{dx}\right|_{x=x_0}(x - x_0) + \frac{1}{2}\left.\frac{d^2\tilde{u}(x; N_p)}{dx^2}\right|_{x=x_0}(x - x_0)^2 + \cdots$$
$$= u_0 + \exp(-\beta\ \tilde{V}_{\text{ref}}(x_0; N_p))\ (x - x_0) \tag{2.16}$$
$$+ \frac{1}{2}\ \frac{d}{dx}\left.\left[\exp(-\beta\ \tilde{V}_{\text{ref}}(x; N_p))\right]\right|_{x=x_0}(x - x_0)^2 + \cdots.$$

Here, $\exp(-\beta\ \tilde{V}_{\text{ref}}(x; N_p))$ is strictly positive, and the sign of $(x - x_0)$ and $(u - u_0)$ will always coincide since the relation $\tilde{u} = \tilde{u}(x)$ is a strictly monotonically increasing function. Given this information, (2.16) could always be uniquely solved for $(x - x_0)$ or x. Then (x_0, u_0) is replaced by $(x, \tilde{u}(x, N_p))$, and (2.16) is solved again for a new x. This iterative procedure is continued until $|\tilde{u}(x, N_p) - u|$ is smaller than some tolerance, which is practically chosen to be $\approx 1000 * \epsilon_{machine}$. Again, in a practical implementation, only the functions exp_beta_Vref$(x; N_p)$ and exp_beta_Vref_prime$(x; N_p)$ are needed.

3. Dynamic contact REPSWA. The aforementioned methodology is appropriate for treating those parts of the potential energy surface with a priori determined barriers between specific conformations, e.g., those separating different values of a dihedral angle as was shown in [31]. These barriers are built into the potential energy model and are therefore, in a sense, static. Another important set of barriers that must be treated, however, are those that arise only when certain conformations are accessed. These cannot be predicted a priori and hence, cannot be built into the potential energy surface. An example of this type of barrier arises from nonbonded interactions that dominate when a molecule accesses compact or globular conformations. In order to treat this type of barrier, an adaptive transformation scheme is needed that is capable of sensing when such conformations are accessed. These transformations will be more complex than the static transformations discussed above and are referred to as *dynamic contact (DC) transformations.*

The DC-REPSWA transformation method can be illustrated with a generalized example for transforming one dihedral in a system of atoms $(\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3, \{\mathbf{r}_{g_1}, \dots, \mathbf{r}_{g_N}\})$, placed in a bath of monoatomic particles $(\mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_M)$, and the major objective is torsional sampling along the dihedral $(\mathbf{r}_1, \mathbf{r}_2,\ \mathbf{r}_3, \mathbf{r}_{g_i})$ treating $\{\mathbf{r}_{g_1}, \dots, \mathbf{r}_{g_N}\}$ as a rigid group. The transformation algorithm for this system builds on the simple static method [31] by adding to the reference potential part of the intermolecular interaction:

$$V_{\text{ref}}(\tilde{\phi}^{g_1}, \Omega^\dagger) = \left[V^{\text{p}}_{\text{tors}}(\tilde{\phi}^{g_1}) + \sum_{i=1}^{n} V^{\text{s}}_{\text{tors}}(\tilde{\phi}^{g_1} + \delta_i) \right]$$

$$+ \alpha \sum_{i=1}^{N} \sum_{j=1}^{M} V^{\text{reg}}_{\text{inter}}(|\tilde{\mathbf{r}}_{g_i}(\tilde{\phi}^{g_1}, \Omega) - \mathbf{s}_j|) S_{\text{inter}}(|\tilde{\mathbf{r}}_{g_i}(\tilde{\phi}^{g_1}, \Omega) - \mathbf{s}_j|), \tag{3.1}$$

where the first term in brackets involves the primary torsion $\tilde{\phi}^{g_1}$ and possibly coupling to n secondary torsions formed from a subset of atoms in the group. The second term represents the intermolecular contribution to the reference potential scaled by an arbitrary parameter α. $V^{\text{reg}}_{\text{inter}}(r)$ is equal to the intermolecular potential $V_{\text{inter}}(r)$ for $r > r_{\text{core}}$ and is a finite analytic function on the interval $[0, r_{\text{core}}]$. *Replacing* $V_{\text{inter}}(r)$ *on* $[0, r_{\text{core}}]$ *with an analytic function is necessary for the inversion of the transformation.* The choice of r_{core} depends on the intermolecular potential; e.g., for all Lennard–Jones-type pair interaction, $0.9\,\sigma$ was used in this study. $S_{\text{inter}}(x)$ is a switching function that cuts off the intermolecular contribution to the reference potential to include only nearest neighbors. Ω and $\Omega^\dagger$ are sets of other coordinates in the system whose components will be made explicit shortly. The important point is that the new dependence of the reference potential on degrees of freedom other than ϕ leads to an adaptive scheme capable of adjusting to changes in the conformation or in the local environment. The transformation will still scale as order N provided that the range of intermolecular interactions considered is finite, and will be efficient if the range is short.

In order to evaluate the reference potential V_{ref}, a group of pseudoatoms $\{\tilde{\mathbf{r}}_{g_1}, \dots, \tilde{\mathbf{r}}_{g_N}\}$ is constructed from the spherical coordinates $\tilde{\xi} \equiv \{(\tilde{r}^{g_1}, \tilde{\theta}^{g_1}, \tilde{\phi}^{g_1}), \dots, (\tilde{r}^{g_N}, \tilde{\theta}^{g_N}, \tilde{\phi}^{g_N})\}$ along a finite number of points in the whole domain $(0, 2\pi]$ of the azimuthal angle $\tilde{\phi}^{g_1}$ using Gauss–Legendre nodes mapped to the interval. At each node, the interactions between each $\tilde{\mathbf{r}}_{g_i}$ and all the neighboring atoms $\mathbf{s}_j$ within a cutoff distance specified by the switching function in (3.1) are calculated. In this construction, $\tilde{\xi} = \tilde{\xi}(\tilde{\phi}^{g_1}|\dots)$ *is made to be a function of only one dynamic variable,* $\tilde{\phi}^{g_1}$, *the primary dihedral angle, and the rest of the variables are obtained from static information.* In particular, $\tilde{\phi}^{g_i} = \tilde{\phi}^{g_1} + \delta_0^{g_i},\ i = 2, \ldots, N$, and $(\tilde{r}^{g_i}, \tilde{\theta}^{g_i}) = (r_0^{g_i}, \theta_0^{g_i}),\ i = 1, \ldots, N$, where all 0-subscripted variables are evaluated from an ideal geometry. Such a choice relies on the fact that commonly used biomolecular force fields [1, 14] contain stiff bonding and bending interaction potentials so that a small deviation in the molecular bonds and bends does not significantly alter the shape and magnitude of the dynamically changing torsional barriers. However, it is noted here that the above scheme is completely general and is not restricted to the case when all atoms in the group are bonded to $\mathbf{r}_3$. Thus, depending on the nature of topological relations within $(\mathbf{r}_3, \{\mathbf{r}_{g_1}, \ldots, \mathbf{r}_{g_N}\})$, $\tilde{\xi}$ could be a function of as many dynamic variables as needed and at most $3N$. Nevertheless, keeping the computational cost modest requires the use of as many static variables as possible. Based on $\tilde{\xi}$, $\{\tilde{\mathbf{r}}_{g_1}, \ldots, \tilde{\mathbf{r}}_{g_N}\}$ is reconstructed using the local reference coordinate frame defined by the unit vectors $\mathbf{e}_\gamma(\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3),\ \gamma = x, y, z$, and the coordinate components $(\tilde{x}_{g_i}(\tilde{r}^{g_i}, \tilde{\theta}^{g_i}, \tilde{\phi}^{g_i}), \tilde{y}_{g_i}(\tilde{r}^{g_i}, \tilde{\theta}^{g_i}, \tilde{\phi}^{g_i}), \tilde{z}_{g_i}(\tilde{r}^{g_i}, \tilde{\theta}^{g_i}, \phi^{g_i}))$ of $\tilde{\mathbf{r}}_{g_i}$ in this basis. Then the unique definition of $\tilde{\mathbf{r}}_{g_i}$ in R-space becomes

$$\begin{aligned} \tilde{\mathbf{r}}_{g_i} &= \mathbf{r}_3 + \tilde{x}_{g_i}\ \ \mathbf{e}_x(\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3) + \tilde{y}_{g_i}\ \ \mathbf{e}_y(\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3) + \tilde{z}_{g_i}\ \ \mathbf{e}_z(\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3) \\ &= \mathbf{r}_3 + \tilde{r}^{g_i}\ \sin(\tilde{\theta}^{g_i})\ \left[\cos(\tilde{\phi}^{g_i})\ \ \mathbf{e}_x + \sin(\tilde{\phi}^{g_i})\ \ \mathbf{e}_y)\right] + \tilde{r}^{g_i}\ \cos(\tilde{\theta}^{g_i})\ \ \mathbf{e}_z. \end{aligned} \tag{3.2}$$

This definition makes explicit the components of the sets Ω and $\Omega^\dagger$: $\Omega = \{\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3, D(\,\tilde{\xi}\,) \setminus \tilde{\phi}^{g_1} \mid S(\,\tilde{\xi}\,)\}$, where $D(\tilde{\xi})$ and $S(\tilde{\xi})$ refer to the subset of dynamic and static variables of $\tilde{\xi}$, respectively. In addition, $\Omega^\dagger = \{\Omega, \mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_M\}$. As described in section 2.2, the reference potential should satisfy the relation $V_{\text{ref}}(a_i) = V_{\text{ref}}(a_f) = 0$, where a_i and a_f are the boundaries of D_n. In order to effectively ensure the latter, the reference potential is shifted by an amount $(1/2)[V_{\text{ref}}(a_i, \Omega^\dagger) + V_{\text{ref}}(a_f, \Omega^\dagger)]$ before the switching function is applied:

$$V_{\text{ref}}(\tilde{\phi}^{g_1}, \Omega^\dagger)\Big|_{final} = \left(V_{\text{ref}}(\tilde{\phi}^{g_1}, \Omega^\dagger) - \frac{1}{2}[V_{\text{ref}}(a_i, \Omega^\dagger) + V_{\text{ref}}(a_f, \Omega^\dagger)]\right) S(\tilde{\phi}^{g_1}). \tag{3.3}$$

Here, $S(\tilde{\phi}^{g_1})$ is a function that switches off the reference potential at the end points $\tilde{\phi}^{g_1} = \{a_i, a_f\}$. The switching function, chosen based on [30], belongs to the class C_∞ and is applied on switching intervals $[a_i, a_i + 0.01(a_f - a_i)]$ and $[a_f - 0.01(a_f - a_i), a_f]$.

By using $V_{\text{ref}}\left(\phi, \Omega^\dagger\right)$ in a nonlinear transformation $\phi^{g_1} \to \phi_u^{g_1}$ defined by (2.3), $\phi_u^{g_1}$ becomes twofold dependent on $\Omega^\dagger$:

$$\begin{gathered} \phi_u^{g_1}(\Omega^\dagger) = c(\Omega^\dagger) \int_0^{\phi^{g_1}} d\phi\ \exp\left[-\beta\ V_{\text{ref}}\left(\phi, \Omega^\dagger\right)\right]; \\ c(\Omega^\dagger) = \frac{2\pi}{I(\Omega^\dagger)}; \quad I(\Omega^\dagger) = \int_0^{2\pi} d\phi\ \exp\left[-\beta\ V_{\text{ref}}\left(\phi, \Omega^\dagger\right)\right]. \end{gathered} \tag{3.4}$$

In addition, $\partial\phi_u^{g_1}/\partial\Omega^\dagger$ is obtained so that the corresponding force transformations for $\Omega^\dagger$ variables can be realized:

$$\begin{aligned} \frac{\partial\phi_u^{g_1}}{\partial\Omega^\dagger}(\Omega^\dagger) &= \frac{\partial}{\partial\Omega^\dagger}\left[c(\Omega^\dagger) \int_0^{\phi^{g_1}} d\phi\ \exp\left[-\beta\ V_{\text{ref}}\left(\phi, \Omega^\dagger\right)\right]\right] \\ &= -\frac{\phi_u^{g_1}}{I(\Omega^\dagger)} \int_0^{2\pi} d\phi\ \frac{\partial}{\partial\Omega^\dagger}\left[\exp\left[-\beta\ V_{\text{ref}}\left(\phi, \Omega^\dagger\right)\right]\right] \\ &\quad + \frac{2\pi}{I(\Omega^\dagger)} \int_0^{\phi^{g_1}} d\phi\ \frac{\partial}{\partial\Omega^\dagger}\left[\exp\left[-\beta\ V_{\text{ref}}\left(\phi, \Omega^\dagger\right)\right]\right]. \end{aligned} \tag{3.5}$$

In practice, a numerical expansion of (2.5) is employed to evaluate the above derivatives:

$$
\begin{aligned}
\tilde{\phi}_u^{g_1}(\Omega^\dagger, N_p) &= \tilde{c}\left(\Omega^\dagger; N_p\right) \int_0^{\phi^{g_1}} d\phi\ \exp\left[-\beta\, \tilde{V}_{\text{ref}}\left(\phi, \Omega^\dagger, N_p\right)\right] \\
&= \frac{2\pi}{\tilde{I}\left(\Omega^\dagger; N_p\right)} \sum_{l=0}^{N_p} \left[C_l\left(\Omega^\dagger; N_p\right)\right] \int_0^{\phi^{g_1}} d\phi\ \ P_l(h(\phi)), \\
\frac{\partial \tilde{\phi}_u^{g_1}}{\partial \Omega^\dagger}(\Omega^\dagger, N_p) &= \frac{\partial}{\partial \Omega^\dagger}\left[\tilde{c}\left(\Omega^\dagger; N_p\right) \int_0^{\phi^{g_1}} d\phi\ \exp\left[-\beta\, \tilde{V}_{\text{ref}}\left(\phi, \Omega^\dagger, N_p\right)\right]\right] \\
&= -\frac{\tilde{\phi}_u^{g_1}}{\tilde{I}\left(\Omega^\dagger; N_p\right)} \int_0^{2\pi} d\phi\ \frac{\partial}{\partial \Omega^\dagger}\left[\sum_{l=0}^{N_p} C_l\left(\Omega^\dagger; N_p\right)\ P_l(h(\phi))\right] \\
&\quad + \frac{2\pi}{\tilde{I}\left(\Omega^\dagger; N_p\right)} \int_0^{\phi^{g_1}} d\phi\ \frac{\partial}{\partial \Omega^\dagger}\left[\sum_{l=0}^{N_p} C_l\left(\Omega^\dagger; N_p\right)\ P_l(h(\phi))\right] \\
&= -\frac{\tilde{\phi}_u^{g_1}}{\tilde{I}\left(\Omega^\dagger, N_p\right)} \sum_{l=0}^{N_p} \frac{\partial}{\partial \Omega^\dagger}\left[C_l\left(\Omega^\dagger; N_p\right)\right] \int_0^{2\pi} d\phi\ \ P_l(h(\phi)) \\
&\quad + \frac{2\pi}{\tilde{I}\left(\Omega^\dagger; N_p\right)} \sum_{l=0}^{N_p} \frac{\partial}{\partial \Omega^\dagger}\left[C_l\left(\Omega^\dagger; N_p\right)\right] \int_0^{\phi^{g_1}} d\phi\ \ P_l(h(\phi)).
\end{aligned}
\tag{3.6}
$$

Here, all tilde quantities refer to variables represented in the numerical expansion, N_p denotes the number of Legendre polynomials (section 2.2), and the expansion coefficients and their gradients are defined as

$$
\begin{aligned}
C_l\left(\Omega^\dagger; N_p\right) &= \frac{2l+1}{2} \sum_{k=0}^{N_p} w_k\ \exp\left[-\beta\, V_{\text{ref}}\left(x_k, \Omega^\dagger\right)\right]\ P_l(y_k), \\
\frac{\partial}{\partial \Omega^\dagger}\left[C_l\left(\Omega^\dagger; N_p\right)\right] &= \frac{2l+1}{2} \sum_{k=0}^{N_p} w_k\ \frac{\partial}{\partial \Omega^\dagger}\left[\exp\left[-\beta\, V_{\text{ref}}\left(x_k, \Omega^\dagger\right)\right]\right]\ P_l(y_k),
\end{aligned}
\tag{3.7}
$$

where $(w_k, y_k\ :\ k = 0, \ldots, N_p)$ are the Gauss–Legendre weights and nodes and $(x_k = h^{-1}(y_k)\ :\ k = 0, \ldots, N_p)$ are the Gauss–Legendre nodes mapped to $(0, 2\pi]$ via inverting the linear relation $h(x)$, which is defined in section 2.2. Furthermore, the derivatives in (3.7) are available analytically as

$$
\begin{aligned}
&\frac{\partial}{\partial \Omega^\dagger}\left[\exp\left[-\beta\, V_{\text{ref}}\left(x_k, \Omega^\dagger\right)\right]\right] \\
&\quad = -\beta\ \exp\left[-\beta\, V_{\text{ref}}\left(x_k, \Omega^\dagger\right)\right] \frac{\partial}{\partial \Omega^\dagger}\left[\alpha \sum_{i=1}^{N}\sum_{j=1}^{M} T_{VS}(|\tilde{\mathbf{r}}_{g_i}(x_k, \Omega) - \mathbf{s}_j|)\right] \\
&\quad = -\beta\ \exp\left[-\beta\, V_{\text{ref}}\left(x_k, \Omega^\dagger\right)\right] \alpha \sum_{i=1}^{N}\sum_{j=1}^{M} T'_{VS}(|\tilde{\mathbf{r}}_{g_i}(x_k, \Omega) - \mathbf{s}_j|) \\
&\qquad \times \left[\frac{\tilde{\mathbf{r}}_{g_i}(x_k, \Omega) - \mathbf{s}_j}{|\tilde{\mathbf{r}}_{g_i}(x_k, \Omega) - \mathbf{s}_j|}\right] \cdot \left[\frac{\partial \tilde{\mathbf{r}}_{g_i}(x_k, \Omega)}{\partial \Omega^\dagger} - \frac{\partial \mathbf{s}_j}{\partial \Omega^\dagger}\right],
\end{aligned}
\tag{3.8}
$$

where $T_{VS} = V^{\text{reg}}_{\text{inter}}\, S_{\text{inter}}$ is the product of two functions introduced in (3.1) and the dependence on the reconstruction in the local reference frame (3.2) is explicitly taken into account. Moreover,

$$\frac{\partial \tilde{\mathbf{r}}_{g_i}(x_k,\Omega)}{\partial \Omega} = \frac{\mathbf{r}_3}{\partial \Omega} + \tilde{x}_{g_i}\,\frac{\mathbf{e}_x(\mathbf{r}_1,\mathbf{r}_2,\mathbf{r}_3)}{\partial \Omega} + \tilde{y}_{g_i}\,\frac{\mathbf{e}_y(\mathbf{r}_1,\mathbf{r}_2,\mathbf{r}_3)}{\partial \Omega} + \tilde{z}_{g_i}\,\frac{\mathbf{e}_z(\mathbf{r}_1,\mathbf{r}_2,\mathbf{r}_3)}{\partial \Omega},$$

$$(3.9)\quad \frac{\partial \tilde{\mathbf{r}}_{g_i}(x_k,\Omega)}{\partial \overline{\Omega}} = 0, \qquad \overline{\Omega} = \left[\Omega^\dagger \setminus \Omega\right]$$

Equations (3.5)–(3.9) contain the steps for computing the derivatives $\partial\phi^{g_1}_u/\partial\Omega^\dagger$, which are necessary for propagating the force components through the overall dynamic group transformation scheme, which takes positions $(\mathbf{r}_1,\mathbf{r}_2,\mathbf{r}_3,\ \{\mathbf{r}_{g_1},\dots,\mathbf{r}_{g_N}\})$, forces $(\mathbf{f}_1,\mathbf{f}_2,\mathbf{f}_3,\ \{\mathbf{f}_{g_1},\dots,\mathbf{f}_{g_N}\})$, and produces their U-space counterparts $(\mathbf{u}_1,\mathbf{u}_2,\mathbf{u}_3,\ \{\mathbf{u}_{g_1},\ \dots,\mathbf{u}_{g_N}\})$ and $(\mathbf{g}_1,\mathbf{g}_2,\mathbf{g}_3,\ \{\mathbf{g}_{g_1},\dots,\mathbf{g}_{g_N}\})$, respectively. The algorithm proceeds as follows: (0) $\mathbf{r}_1,\mathbf{r}_2,\mathbf{r}_3,\mathbf{r}_{g_1}$ is assumed to be the primary dihedral; ($i\uparrow$) translate/rotate the vectors in the group $(\mathbf{r}_{g_1},\dots,\mathbf{r}_{g_N})$ individually into a coordinate frame in which $\mathbf{r}_3$ is at the origin and $\mathbf{r}_3-\mathbf{r}_2$ lies along the z-axis; ($ii\uparrow$) the group $(\mathbf{r}_{g_1},\dots,\mathbf{r}_{g_N})$ is resolved into spherical coordinates $(r^i,\theta^i,\phi^i) : i \in \{g_1,\dots,g_N\}$; ($g\uparrow$) atoms are grouped; ($iii$) DC-REPSWA transformation is applied on the azimuthal angle, $\phi^{g_1,g} \to \phi^{g_1,g}_u$; ($g\downarrow$) atoms are ungrouped; ($ii\downarrow$) step ($ii\uparrow$) is inverted; ($i\downarrow$) step ($i\uparrow$) is inverted to construct the new pseudo-Cartesian frame $(\mathbf{u}_1,\mathbf{u}_2,\mathbf{u}_3,\{\mathbf{u}_{g_1},\dots,\mathbf{u}_{g_N}\})$ with forces $(\mathbf{g}_1,\mathbf{g}_2,\mathbf{g}_3,\{\mathbf{g}_{g_1},\dots,\mathbf{g}_{g_N}\})$. The above steps are explicitly illustrated in Appendix A. Note that each primitive transformation step $x \to y$ is augmented by a force transformation $f^x \to f^y$, which in general can be illustrated via the following example:

$$y = \Lambda(x), \qquad x = (\mathbf{x}_1,\dots,\mathbf{x}_N), \qquad y = (\mathbf{y}_1,\dots,\mathbf{y}_N),$$

$$(3.10)\qquad f^y = -\frac{\partial V}{\partial y} = -\frac{\partial V}{\partial x}\cdot\frac{\partial x}{\partial y} = f^x \cdot \frac{\partial[\,\Lambda^{-1}(y)]}{\partial y}.$$

Another important aspect of the algorithm is the appearance of the rotation matrix R in steps ($i\uparrow$) and ($i\downarrow$). To obtain R, two unit vectors, $\mathbf{e}_1 = (\mathbf{r}_3-\mathbf{r}_2)/||\mathbf{r}_3-\mathbf{r}_2||$ and $\mathbf{e}_2 = (\mathbf{r}_1-\mathbf{r}_2)/||\mathbf{r}_1-\mathbf{r}_2||$, are constructed first, then $\mathbf{e}_1$ and $\mathbf{e}_2$ are used to construct a local coordinate frame spanned by three vectors, $\mathbf{e}_z = \mathbf{e}_1$, $\mathbf{e}_y = (\mathbf{e}_1\times\mathbf{e}_2)/||\mathbf{e}_1\times\mathbf{e}_2||$, and $\mathbf{e}_x = (\mathbf{e}_y\times\mathbf{e}_z)/||\mathbf{e}_y\times\mathbf{e}_z||$. Given, $\mathbf{e}_x$, $\mathbf{e}_y$, and $\mathbf{e}_z$, $R = (\mathbf{e}'_x,\mathbf{e}'_y,\mathbf{e}'_z)'$. Furthermore, step ($iii$) is surrounded by ($g\uparrow$) and ($g\downarrow$) primitive steps, which are responsible for the *rigid transformation* of the group and propagate the forces accordingly. The scheme of Appendix A displays the grouping ($g\uparrow$), nonlinear transformation (iii), and ungrouping ($g\downarrow$) steps. In (iii), only atoms participating in the primary dihedral $(\mathbf{r}_1,\mathbf{r}_2,\mathbf{r}_3,\mathbf{r}_{g_1})$ and all force components connected to the variable set $\Omega^\dagger$ are affected. Again, the derivation of these force transformation steps strictly follows rules introduced by (3.10). Following (iii), step ($g\downarrow$) translates the rotational effect of the primary transformation to the group $\{\mathbf{r}_{g_2},\dots,\mathbf{r}_{g_N}\}$. Finally, steps ($ii\downarrow$) and ($i\downarrow$) regenerate positions and forces into a Cartesian frame.

Note that the constant c in (3.4) becomes a function of $\Omega^\dagger$ and the effective potential becomes $V_{\text{eff}}(U,\Omega^\dagger) = V(R) - V_{\text{ref}}(\phi^{g_1,g},\Omega^\dagger) + (1/\beta)\ln[c(\Omega^\dagger)]$. Thus, the transformed variables move on $\tilde{V}_{\text{eff}}(U,\Omega^\dagger;N_p) = V(R) - \tilde{V}^{\text{eff}}_{\text{ref}}(\phi^{g_1,g},\Omega^\dagger;N_p)$, where the last term is the applied numerical expansion controlled by the number of Legendre polynomials, N_p, and

$$\begin{aligned}
\tilde{V}_{\rm ref}^{\rm eff}(\phi^{g_1,g},\Omega^\dagger;N_p) &= \tilde{V}_{\rm ref}(\phi^{g_1,g},\Omega^\dagger;N_p) - \frac{1}{\beta}\ln\left[\tilde{c}(\Omega^\dagger;N_p)\right],\\
\tilde{V}_{\rm ref}(\phi^{g_1,g},\Omega^\dagger;N_p) &= -\frac{1}{\beta}\ln\left[\exp\left[-\beta\,\tilde{V}_{\rm ref}\left(\phi^{g_1,g},\Omega^\dagger;N_p\right)\right]\right]\\
&= -\frac{1}{\beta}\ln\left[\sum_{l=0}^{N_p} C_l\left(\Omega^\dagger;N_p\right)\ P_l(h(\phi^{g_1,g}))\right],\\
\tilde{c}(\Omega^\dagger;N_p) &= \frac{2\pi}{\tilde{I}(\Omega^\dagger;N_p)}.
\end{aligned} \tag{3.11}$$

In order to evaluate the forces arising from the effective reference potential, $V_{\rm ref}^{\rm eff}$, the derivatives with respect to all components of $\Omega^\dagger$ must be evaluated:

$$\begin{aligned}
\frac{\partial\tilde{V}_{\rm ref}^{\rm eff}(\phi,\Omega^\dagger;N_p)}{\partial\Omega^\dagger} &= -\frac{1}{\beta\exp\left[-\beta\,\tilde{V}_{\rm ref}\left(\phi,\Omega^\dagger;N_p\right)\right]}\ \frac{\partial}{\partial\Omega^\dagger}\left[\exp\left[-\beta\,\tilde{V}_{\rm ref}\left(\phi,\Omega^\dagger;N_p\right)\right]\right]\\
&\quad -\frac{1}{\beta\,\tilde{c}(\Omega^\dagger;N_p)}\ \frac{\partial\tilde{c}(\Omega^\dagger;N_p)}{\partial\Omega^\dagger}\\
&= -\frac{1}{\beta\exp\left[-\beta\,\tilde{V}_{\rm ref}\left(\phi,\Omega^\dagger;N_p\right)\right]}\ \sum_{l=0}^{N_p}\frac{\partial}{\partial\Omega^\dagger}\left[C_l\left(\Omega^\dagger;N_p\right)\right]\ P_l(h(\phi))\\
&\quad -\frac{2\pi}{\beta\,[\tilde{I}\left(\Omega^\dagger;N_p\right)]^3}\ \sum_{l=0}^{N_p}\frac{\partial}{\partial\Omega^\dagger}\left[C_l\left(\Omega^\dagger;N_p\right)\right]\int_0^{2\pi}d\phi\ \ P_l(h(\phi)).
\end{aligned} \tag{3.12}$$

In addition, the basic derivatives with respect to the torsional degrees of freedom are also evaluated as

$$\begin{aligned}
\frac{\partial\tilde{V}_{\rm ref}^{\rm eff}(\phi,\Omega^\dagger;N_p)}{\partial\phi} &= -\frac{1}{\beta\ \exp(-\beta\,\tilde{V}_{\rm ref}(\phi,\Omega^\dagger;N_p))}\ \frac{\partial}{\partial\phi}\left[\exp(-\beta\,\tilde{V}_{\rm ref}(\phi,\Omega^\dagger;N_p))\right]\\
&= -\frac{1}{\beta\ \exp(-\beta\,\tilde{V}_{\rm ref}(\phi,\Omega^\dagger;N_p))}\ \sum_{l=0}^{N_p} C_l\left(\Omega^\dagger;N_p\right)\ \dot{P}_l(h(\phi))\,\dot{h}(\phi).
\end{aligned} \tag{3.13}$$

In practice, the two contributions to the force transformation, which are connected to step (iii), the nonlinear dihedral transformation, and $V_{\rm ref}^{\rm eff}$, are computed separately, which becomes a crucial implementation issue for handling complex systems. Thus, in a general $R \to U$, $FR \to FU$ transformation, where more than one torsional dihedral is involved, all force components originating from $V_{\rm ref}^{\rm eff}$ must be present in the R Cartesian space before $(R, FR) \to (U, FU)$ is executed, so that the reference forces are already present in the R-space before any nonlinear transformation step is performed.

The key to using DC-REPSWA on any molecular system decomposed into torsional transformation groups is to realize that while the Jacobian in multidimensional spaces can be difficult to determine analytically, there are special cases where the determinant of large matrices can be handled easily, e.g., an upper or lower triangular matrix, for which the Jacobian is simply the product of the diagonal elements. This is, of course, perfectly suited to a torsional decomposition, as the barriers to motion are formed as each group of atoms is added. Start with three base atoms. Upon adding the first group, the first torsion is added; upon adding a second group, a second torsion is added; etc., suggesting that a coordinate system change can be expressed in upper triangular form. In addition, as forces are required in any MD or HMC method, the upper/lower triangular form is already LU decomposed, and force computations can be performed recursively. If the matrix is sparse, as occurs when only torsional barriers are considered, then the computations required to perform DC-REPSWA are of order N. The most important point here is that *all torsional barriers can be removed in order* $\mathcal{O}(N)(!)$. The generalization of the dynamical transformation algorithm to an arbitrary number of dihedrals is demonstrated in Appendix B. Note that attempting to remove all torsional barriers in large molecular systems using metadynamics [8, 7], guiding functions [32], and umbrella sampling [25, 26] would not be possible.

When large molecular chains with intermolecular/nonbonded interactions are treated via DC-REPSWA, rather large motions will occur which result in stability/acceptance problems. The molecule will essentially run into itself, yielding low acceptance probabilities in HMC. In order to overcome the overly aggressive motion induced by DC-REPSWA, it is sufficient to insert the identity transformation for every 10 to 15 groups, i.e., leave a dihedral untransformed. It is then possible to change the starting point of the DC-REPSWA transformation every 100–200 HMC moves so that the identity transformation is used for different torsions at different points in the simulation. For example, suppose torsions 4–14, 16–26, and so forth, are transformed for 100 steps. Subsequently, one might transform torsions 6–16, 18–18, and so forth, for the next 100 steps, using a random number generator to determine the starting point for each set of transformations. In this way, large torquing/clashing motions of the molecule are damped, but local sampling remains sufficient. We refer to this method as staging, dynamic contact REPSWA (SDC-REPSWA). Another important factor in the transformation is the scaling constant α defined in (3.1), which controls the extent to which intermolecular interactions "remain" in U-space. In general, the larger α, the better the sampling; however, $\alpha > 0.9$ could cause contraction in the attractive radius of important energy basins. *Nevertheless, it should be noted that any value of* α *will exactly preserve the partition function.*

4. Results.

The performance of the DC-REPWSA method is studied through the consideration of a series of models of increasing difficulty and topological complexity. First, DC-REPSWA is tested in its ability to reproduce the canonical distribution in a one-contact model system, where the original REPSWA [31] fails due to the absence of intermolecular terms in the reference potential. This is followed by full SDC-REPSWA simulations on molecular topologies with increasing complexity such as chains of interacting single beads and beads with branches. The latter topologies are becoming increasingly important in computational biology since they characterize the united residue and united side chain ("coarse-grained") descriptions of protein structures. The above topological classes are studied using examples of united- and all-atom alkanes, so that equilibrium observables could still be computed even for long chains with many dihedrals. Finally, an application to the folding of an off-lattice protein model is presented.

In all cases, the performance of SDC-REPSWA is benchmarked against both HMC and the most widely used sampling method, PT or replica exchange [15, 13, 4]. For both SDC-REPSWA and the individual replicas in PT, HMC was used as the "driver" algorithm. The basis of the comparison to PT is the number of time steps required to achieve the corresponding sampling efficiency. The current computational overhead of the DC-REPSWA transformation per step compared to a simple MD proposal step is much less than an order of magnitude, and efforts are being made to reduce this factor as much as possible.

4.1. DC/SDC-REPSWA and PT protocols. The DC-REPSWA method applied in the following examples employed a torsional space transformation with $a_i = 0$, $a_f = 2\pi$ boundaries, and $L = 64$ Legendre polynomials, and unless otherwise stated, a reference intermolecular scaling factor, $\alpha = 0.8$ (cf. (3.1)) was used. SDC-REPSWA was used with a staging length of 10. The PT protocol employed 10 replicas, each within a temperature range of 300 K–1000 K and a replica exchange probability in the range 0.05–0.2. In order to obtain the best performance for individual systems, both the temperature ladder and the exchange probability were optimized within the specified ranges. The HMC driver employed an MD proposal kernel with a time step of 0.5 fs (DC/SDC-REPSWA) and 5.0 fs (PT, HMC).

4.2. A simplified model system with one contact. As a test of the dynamic contact transformation method, a simple system with one dihedral is constructed in which a four-atom chain $(\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3, \mathbf{r}_4)$ interacts with a single atom, $\mathbf{r}_5$, via a Lennard–Jones potential specified by $\epsilon = 0.866$ kJ·mol^{-1} and $\sigma = 3.775$ Å. The three base atoms of the chain and the single atom are fixed at positions $(\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3)$ and $\mathbf{r}_5$, respectively, and only $\mathbf{r}_4$ is allowed to move. The latter can be resolved into spherical coordinates (r, θ, ϕ) so that the distribution of the dihedral angle can be computed by performing a two-dimensional quadrature integral, $P(\phi) = 1/\mathcal{N} \int_0^\infty dr \int_0^\pi d\theta \, \exp(-\beta \, V(r, \theta, \phi))$, where $V(r, \theta, \phi)$ denotes the potential of the system and $\mathcal{N}$ is chosen such that $\int_0^{2\pi} P(\phi) \, d\phi = 1$. This distribution is also computed by HMC simulations carried out using the original REPSWA [31], the new DC-REPSWA, and no transformations over the dihedral angle (standard HMC).

In Figure 4.1, the dihedral angle ϕ is plotted versus simulation time for DC-REPSWA, REPSWA, and the HMC methods. Figure 4.1 shows the distribution

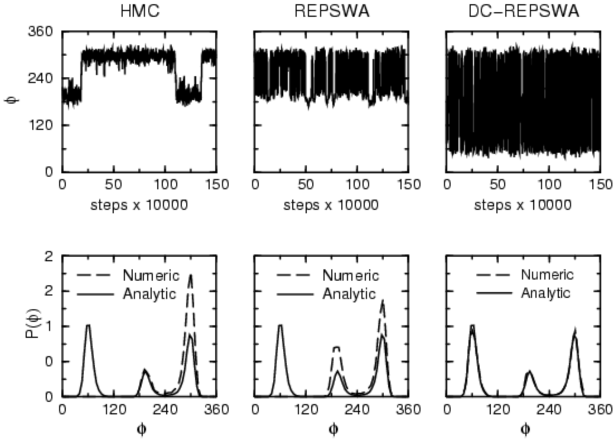

FIG. 4.1. *Top: Instantaneous value of the dihedral angle versus MC steps for the system of section* 4.2 *using HMC, REPSWA, and DC-REPSWA methods. Bottom: Position distribution of the dihedral angle produced by the above methods. All distributions are compared with the exact distribution obtained from multidimensional numerical quadrature integral.*

$P(\phi)$ of the dihedral angle for the three simulation methods along with the analytical result. The interaction of the chain with the single atom significantly alters its conformation space, suggesting that even the original (static) REPSWA transformation is not robust enough for efficient sampling. However, it can be seen that the dynamic transformation leads to a dramatic improvement in conformational sampling efficiency over both the static and no transformation cases and is able to reproduce the analytical distribution function. PT with different numbers of replicas was also applied to the same system, and it was found that at least 10 replicas were needed to reproduce the analytical distribution within the same number of MC iterations.

4.3. A chain of connected beads. As a representative of this topological class, a recent united-atom alkane chain model [16] of chain length 50 was studied. In Figure 4.2, the convergence of the dimensionless end-to-end distance is presented for 3×10^6 step trajectories generated by three different methods, HMC, PT, and SDC-REPSWA. It is evident that HMC does not converge at all, PT is close to convergence on the "time scale" of the trajectory, and REPSWA is converged within roughly 1/6th of the total run time. Thus, the speedup of SDC-REPSWA relative to PT is a factor of 6 or a factor of 60 if the 10 replicas of PT are taken into account. One inset of Figure 4.2 shows a 10 times longer PT trajectory and indicates that on this time scale PT is truly converged. Finally, Figure 4.2 also compares a higher dimensional observable, the end-to-end distributions for SDC-REPSWA, and long PT and HMC trajectories. The comparisons demonstrate that the SDC-REPSWA is a rigorous canonical sampling method with superior performance.

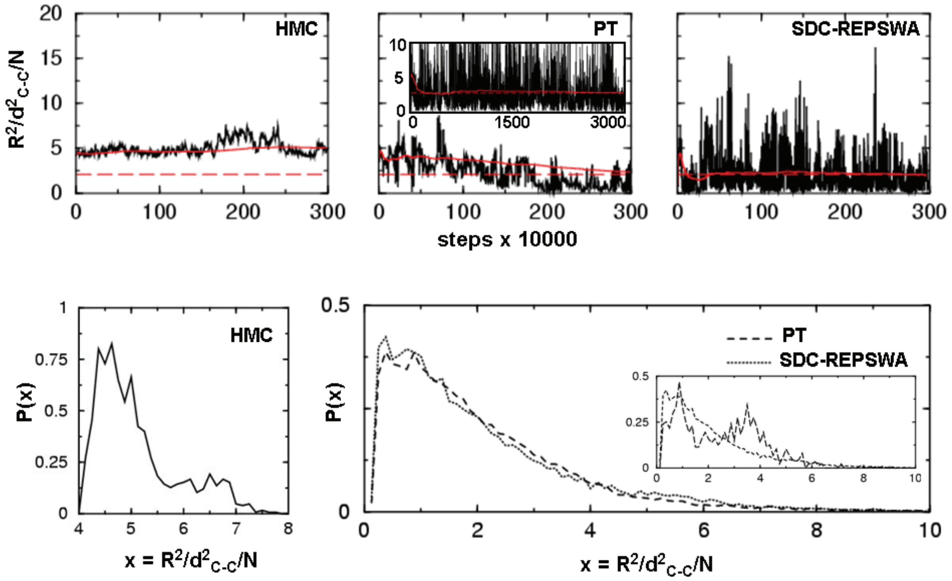

FIG. 4.2. *Comparison is made for HMC, PT with* 10 *replicas, and SDC-REPSWA algorithms in their sampling efficiency of the conformational space of the* C_{50} *united-atom alkane molecule described by the Tra-PPE potential* [16]. *Top: Dimensionless end-to-end distance (black) and its cumulative average (solid red) as a function of the MC steps. The expectation value (dashed red) is indicated for each plot. The inset depicts a* 10 *times longer trajectory produced by applying the PT algorithm. Bottom: Distribution* $(P(x))$ *of the end-to-end distance* x*, calculated from* 30×10^6 *step HMC (solid), PT (dashed), and* 3×10^6 *step SDC-REPSWA (dotted) trajectories. The inset compares the distributions for* 3×10^6 *step PT (dashed) and SDC-REPSWA (dotted) trajectories.*

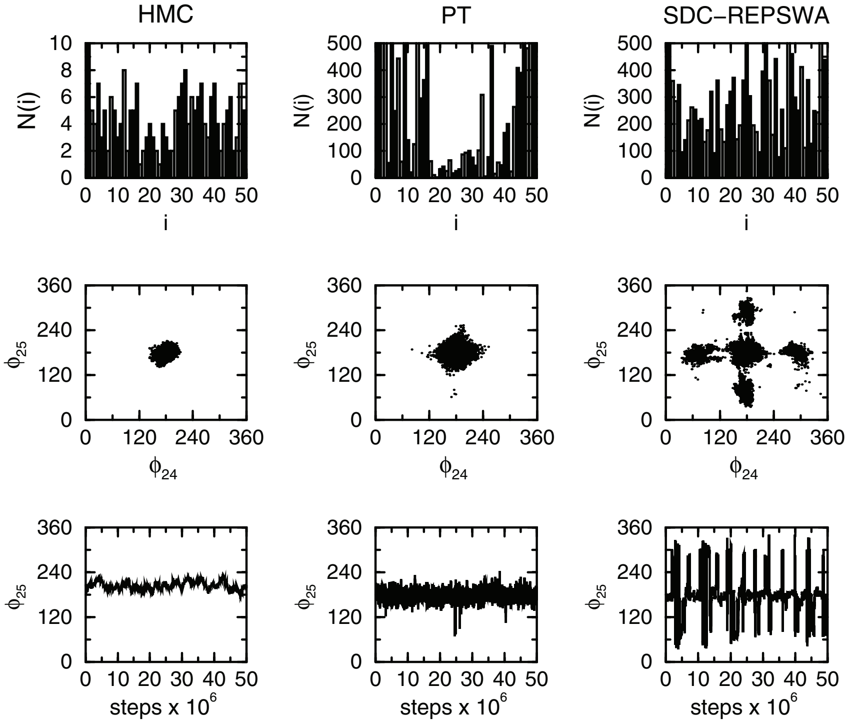

FIG. 4.3. *Comparison is made for HMC, PT with* 10 *replicas, and SDC-REPSWA algorithms in their sampling efficiency of the conformational space of the C_{50} all-atom alkane molecule described by the CHARMM*22 [14] *potential. Top: N(i), number of barrier crossing events for each torsional dihedral i, found in a* 5×10^7 *MC step trajectory. One crossing event of a dihedral angle corresponds to any transition between gauche* ($\phi = 60°/300°$) *and trans* ($\phi = 180°$) *conformers. Middle: Ramachandran plot of the central dihedral angles* (ϕ_{24}, ϕ_{25})*. Bottom: Value of the central dihedral* (ϕ_{25}) *as a function of the MC steps.*

4.4. A chain of beads with branches. Following the above example, the SDC-REPSWA method was employed here on a representative of this topological class, an all-atom alkane chain model [14] of chain length 50. A 5×10^7 length HMC trajectory was generated for the different methods.

Figure 4.3 summarizes the comparison of the three methods. In terms of the efficiency of dihedral sampling, PT constitutes a significant improvement over HMC; however, this enhancement in sampling does not reach the central 20 dihedrals. In the application of SDC-REPSWA, on the other hand, sampling improvement is more uniformly distributed over all dihedrals. The lack of barrier crossing events in the central dihedrals for PT is further demonstrated with the central Ramachandran plots, which refer to the plots of the central dihedrals with respect to each other. In addition, the time evolution of the central dihedrals clearly shows the differences in the performance of the three methods. The quantitative improvement of SDC-REPSWA over PT with 10 replicas can be measured by the 20 central dihedrals. Here, the average numbers of barrier crossing events are 50 and 200 for PT and SDC-REPSWA, respectively, leading to a factor 4 improvement. Again, if the number of PT replicas is considered, this is an overall improvement of 40 for this quantity.

The fluctuations in the dimensionless end-to-end distances were also compared. As shown in Figure 4.4, the cumulative average of this quantity is converged within

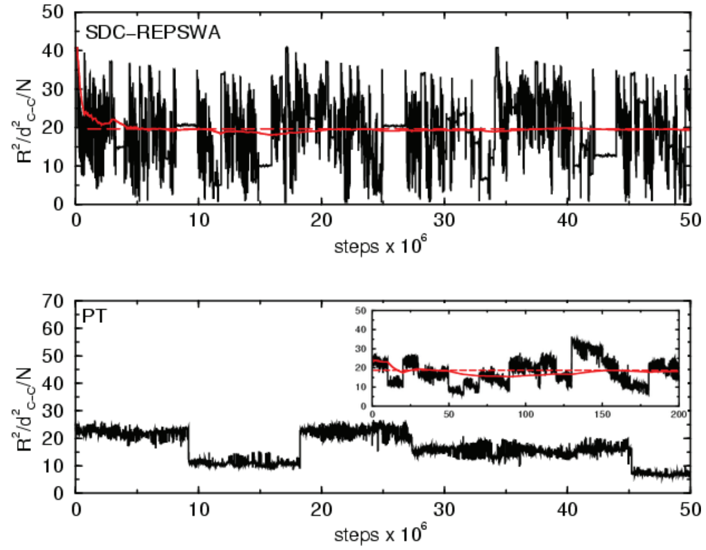

FIG. 4.4. *Comparison is made for PT with* 10 *replicas and SDC-REPSWA algorithms in their sampling efficiency of the conformational space of the* C_{50} *all-atom alkane molecule described by the CHARMM*22 [14] *potential. Top: Dimensionless end-to-end distance (black), its cumulative average (solid red), and the expectation value (dashed red) as a function of the MC steps for SDC-REPSWA. Bottom: Dimensionless end-to-end distance (black) for PT. The inset depicts a* 4 *times longer trajectory and shows the cumulative average (solid red) with the expectation value (dashed red).*

5×10^6 steps for SDC-REPSWA, whereas for PT convergence is incomplete even for 200×10^6 steps. Again, considering the 10 separate systems in PT, the overall speedup in this measure is at least $4 \times 10 = 40$, which is the most substantial speedup for the DC-REPSWA systems considered in this study.

4.5. An off-lattice β-sheet model protein. The last example is a simplified united-residue off-lattice β-sheet protein model [5, 22] with 46 residues composed of three different types: hydrophobic (B), hydrophilic (L), and neutral (N). A particular sequence of "amino acids," in this case, $B_9N_3(LB)_4N_3B_9N_3(LB)_5L$, is known to fold into a β-barrel conformation. Recent studies [18, 28] have shown that this system is characterized by a rough energy landscape. In addition, there is a minimal thermodynamic driving force towards the global (free) energy minium, or "native," structure due to the presence of several attractive energy basins that are populated with equally high probabilities (see [18]: "this system seeks only a general β barrel structure"). Thus, this model protein serves as an excellent benchmark system for sampling/optimization methods as an engine for efficiently locating the dominant energy basins characterized by the β-barrel motif. Note that, in contrast to the previous alkane examples, we do not directly seek equilibrium properties for this system, although these are certainly available. Algorithms with superior performance on the latter problem are expected to be instrumental in ab initio protein structure prediction.

In this example, three different methods, specifically, standard HMC, PT with 16 replicas, and SDC-REPSWA, are compared for their effectiveness in finding the native conformation from an ensemble of random initial configurations, each being at least 10Å root mean-square deviation (RMSD) from the "native" state.

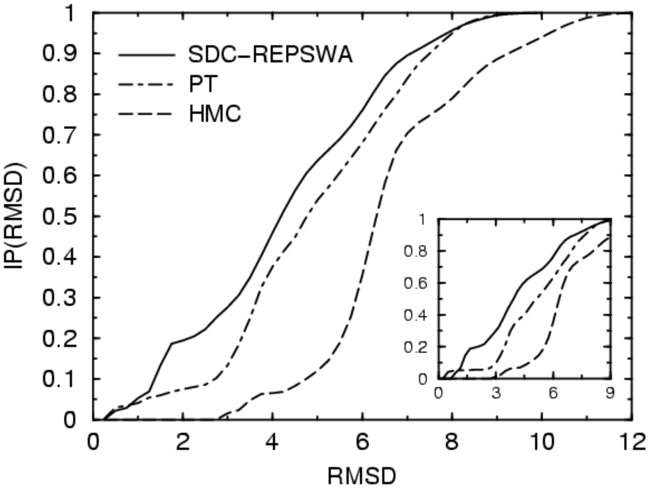

Fig. 4.5. *Comparison is made for HMC (dashed), PT (dot-dashed) with* 10 *replicas, and SDC-REPSWA (solid) algorithms in their sampling efficiency for the simplified united-residue off-lattice β-sheet model protein* [5, 22]. $IP(x) = \int_0^x P(y)dy$ *is the cumulative integral of the RMSD distributions* $P(x)$ *over structural ensembles, which were generated by* 20 *independent trajectories initiated from random coil structures with at least* 10 *ÅRMSD distance from the native state. Here, RMSD represents the average distance of each structure from a pool of native-like* β*-barrel motifs and the distance of each structure from the global energy minimum structure (inset).*

Each trajectory was generated at a temperature of $T = 0.8T_f$, where T_f is the protein folding temperature. The 16 replicas of PT were distributed in a temperature range of 0.5–4.4 T_f, and only trajectories from the $T = 0.8T_f$ replica were considered for statistics. The driver for each method was chosen to be HMC with a collective (MD) proposal time step of $\Delta t = 0.5$ fs. When constructing the reference potential of (3.1), the parameter $\alpha = 0.6$ was used to leave some intermolecular interaction in the U-space. This choice proved to be useful for striking a balance between energetic and entropic barrier effects.

In ab initio protein structure prediction, one of the important properties of different algorithms is their ability to move unfolded or randomly generated structures close to the native state. Here, 20 10^6-step trajectories were generated with each method, and based on these trajectories, *nonequilibrium* structural ensembles were created. Then, the average RMSD distance of each structure present in structural ensembles was calculated from a pool of native-like β-barrel motifs found by both PT and SDC-REPSWA methods. Furthermore, the RMSD distance of each structure present in structural ensembles was calculated from the global energy minimum. Based on this data, RMSD distributions $P(x)$ were obtained for all methods. Figure 4.5 depicts the cumulative integral, $IP(x) = \int_0^x P(y)dy$, of the above distributions. Basically, $IP(x) = P(. \in S_x)$ is the probability that a visited structure is contained in the set $S_x = \{s : 1/k \sum_{i=1}^k RMSD(s, s_k) \leq x\text{Å}\}$, where $RMSD(s, s_k)$ is the RMSD distance of a given structure s from one of the native-like β-barrel structures s_k. According to Figure 4.5, the probabilities that the different methods move the initial coil structure within an average of 4Å RMSD from native-like β-barrel structures are 0.05, 0.30, and 0.5 for HMC, 16-replica PT, and SDC-REPSWA, respectively. The analogue probabilities for 1.5Å are 0.0, 0.07, and 0.2. Within this last range SDC-REPSWA outperforms 16-replica PT by a factor of 3, or a factor of $16 \times 3 = 48$ if the number of

PT replicas is taken into account. This improvement in locating native-like β-barrel structures from random initial coil structures is consistent with the previous results. Figure 4.5 also depicts the case with $k = 1$, $s_1 = s_n$, where s_n is the global minimum structure. Here, the analogue probabilities for the 1.5Å case are 0.0, 0.05, and 0.2, respectively, so that SDC-REPSWA outperforms 16-replica PT by a factor of 4, or a factor of $16 \times 4 = 64$ if the number of PT replicas is taken into account. This latter case marks the improvement in locating the global minimum structure from random initial coil structures.

In addition, it is important to note here that DC-REPSWA need not be viewed entirely as an alternate approach but rather as one that can be *combined* with other methods, such as PT, as a means of enhancing the efficiency of other approaches. The combination of these methods is currently under development.

5. Conclusion. The development of conformational sampling algorithms capable of leading coarse-grained or atomistic simulation into the era of systems biology research will require synergistic adaptation of many methods and concepts. If accomplished, equilibrium features of self-assembly, biological nanomachine activity, and other large-scale biological processes could all potentially be treated in an atomistic or coarse-grained fashion. However, significant challenges must be met, and success is not yet entirely at hand.

Here, we have taken a significant step forward by introducing the SDC-REPSWA approach, the implementation details of which were clearly laid out. The superior performance of the method over PT/replica-exchange was demonstrated for a variety of systems with increasing topological complexity including a model protein. At present, the SDC-REPSWA method is being embedded in our atomistic protein simulation code, PINY_MD [17], to allow realistic folding experiments to be performed in generalized-Born and Poisson–Boltzmann solvent. Success would quickly lead to thorough atomistic applications in proteomics and systems biology.

6. Supplementary information. In this section, further information is provided about the DC-REPSWA-HMC method. In particular, the next subsections provide additional details for the general comparison of DC-REPSWA and other state-of-the-art sampling methods, the rate of convergence for one-dimensional systems, some parameter details of the protein model used in this study.

6.1. DC-REPSWA and other methods. It is important to consider the difference between DC-REPSWA and other methods. First, consider the guiding potential or thermodynamic perturbation [32] technique which uses the identity

$$\begin{aligned} 1 &= \exp[-\beta V_{\mathrm{ref}}(x)] \exp[\beta V_{\mathrm{ref}}(x)], \\ Q &= \frac{1}{h} \int dp_x \int dx \exp[-\beta H_{\mathrm{eff}}(p_x, x)] \exp[-\beta V_{\mathrm{ref}}(x)], \\ H_{\mathrm{eff}}(p_x, x) &= \frac{p_x^2}{2m} + V(x) - V_{\mathrm{ref}}(x). \end{aligned} \tag{6.1}$$

The sampling does not yield the correct probability distribution without an unbiasing correction factor,

$$\langle A(x) \rangle = \frac{\langle A(x) \exp[-\beta V_{\mathrm{ref}}(x)] \rangle_{(H_{\mathrm{eff}})}}{\langle \exp[-\beta V_{\mathrm{ref}}(x)] \rangle_{(H_{\mathrm{eff}})}}, \tag{6.2}$$

a procedure that scales poorly with the number of degrees of freedom in $V_{\rm ref}(x)$. This is quite different from DC-RESPWA, which scales well with the number of degrees of freedom, does not require an unbiasing factor, and involves a variable transformation from x to u to introduce the reference potential. The umbrella sampling technique [25, 26] behaves similarly. Many simulations using a modified potential energy surface

$$\tilde{Q}_k = \frac{1}{h}\int dp_x \int dx \exp[-\beta H_{\rm eff}^{(k)}(p_x, x)],$$

$$H_{\rm eff}^{(k)}(p_x, x) = \frac{p_x^2}{2m} + V(x) + V_{bias}^{(k)}(x) \tag{6.3}$$

are performed. Since $\tilde{Q}_k \neq Q$, the equations

$$\tilde{P}_k(x) \equiv \frac{\exp[-\beta(V(x) + V_{bias}^{(k)}(x))]}{\tilde{Q}_k},$$

$$P(x) \equiv \frac{\exp[-\beta V(x)]}{Q}$$

$$= \frac{\sum_k \tilde{P}_k(x)\frac{\tilde{Q}_k}{Q}}{\sum_j \frac{\tilde{Q}_j}{Q}\exp[\beta V_{bias}^{(j)}(x)]}$$

must be solved self-consistently for the weighting factors $\tilde{Q}_j/Q$. This procedure, again, scales unfavorably with the number of degrees of freedom in $V_{\rm ref}(x)$ and does not involve a change of variables as in DC-REPSWA.

Metadynamics [8, 7], like umbrella sampling and guiding functions, can be applied only to a finite number of degrees of freedom. It uses a basis set expansion to describe a low dimensional free energy surface (4–5 degrees of freedom) whose coefficients are determined on the fly. The number basis set coefficients scale as N^d.

Last, consider the PT method [4, 19]. Many simulations at different inverse temperatures, $\beta_j = 1/[k_B T_j]$ using

$$Q_j = \frac{1}{h}\int dp_x \int dx \exp[-\beta_j H(p_x, x)],$$

$$H(p_x, x) = \frac{p_x^2}{2m} + V(x), \tag{6.4}$$

are run simultaneously. MC exchange moves (switching of inverse temperatures) are attempted periodically. Acceptance depends exponentially on $[\Delta\beta_{jm}\Delta E_{mj}]$, where ΔE_{mj} is the energy difference between system m and system j, and $\Delta\beta_{jm}$ is the associated inverse temperature difference. Since ΔE_{mj} grows with N, this requires increasing fine-grained sampling of β to accept moves, and the method scales like N^2 provided that the system is away from a phase transition, where the scaling is substantially worse [6]. Note, however, that several important modifications to PT have been introduced [12, 11] that help improve the scaling with system size and its overall performance.

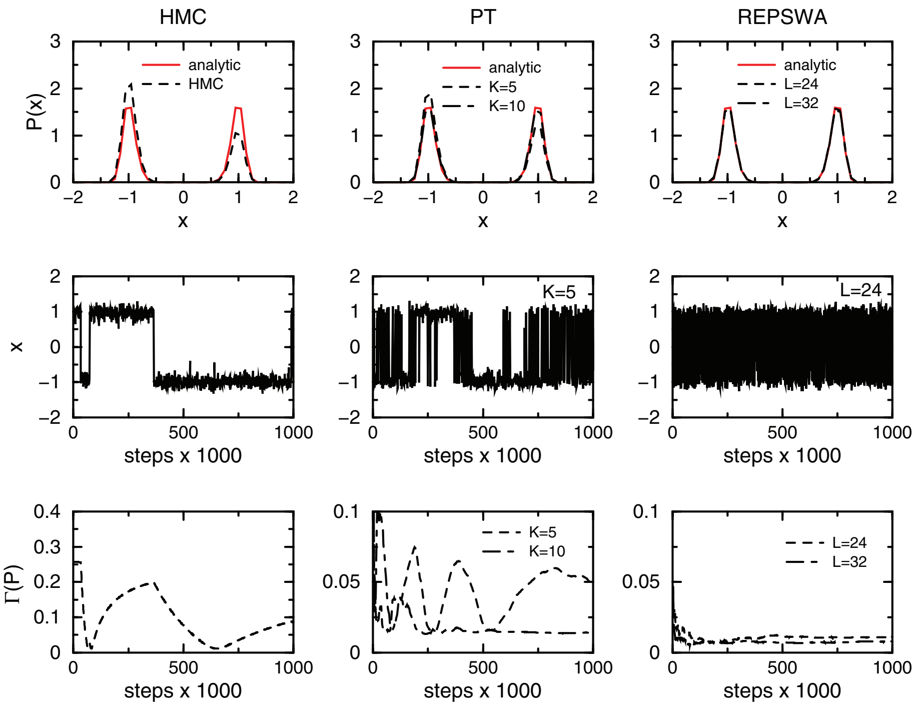

FIG. 6.1. *Comparing the performance of HMC, PT, and REPSWA to sample from a one-dimensional canonical distribution based on the double well potential, $V(x) = \epsilon_0/a^4(x^2 - a^2)^2$ with $\epsilon_0 = 10$. Top: Analytical (solid) is compared with the numerical distributions generated with varying number of replicas (PT), $K = 5$ (dashed), $K = 10$ (long dashed), and Legendre polynomials (REPSWA), $L = 24$ (dashed), $L = 32$ (long dashed). Middle: Position as a function of iteration steps for $K = 5$ replicas (PT) and $L = 24$ Legendre polynomials (REPSWA). Bottom: Convergence to the analytical distribution as a function of the iteration steps. Varying number of replicas (PT) and Legendre polynomials (REPSWA) are considered. For PT, $K = 5$ (dashed) and $K = 10$ (long dashed). For REPSWA, $L = 24$ (dashed) and $L = 32$ (long dashed).*

Certainly, PT and umbrella sampling are useful techniques. Indeed, these can be combined with DC-REPSWA and other algorithms such as metadynamics to form yet more powerful approaches of still greater efficiency. Here, however, it is useful to consider the efficiency of DC-RESPWA alone in comparison to standard HMC and PT so that its benefits can be clearly distinguished from those of supporting techniques.

6.2. The rate of convergence: A one-dimensional study. The efficiency of REPSWA for the quartic double well (2.11) with a barrier height of $\epsilon_0 = 10kT$ is compared to PT and HMC with a varying number of Legendre polynomials used to describe the reference potential. In particular, PT has been performed with different numbers of replicas. In each case 10^5 iterations were generated with an integration step of 0.1. Results presented in Figure 6.1 indicate that while MD is hopelessly inefficient, PT is reasonable, and REPSWA is highly efficient. The convergence $\Gamma_t[P]$ of the probability distribution is given by

$$\Gamma_t[P] = \frac{1}{N_{\text{bins}}} \sum_{i=1}^{N_{\text{bins}}} |P(x_i, t) - P_{\text{true}}(x_i)|\,, \tag{6.5}$$

where $P(x,t)$ is the computed distribution at time t in the simulation, $P_{\mathrm{true}}(x)$ is the true distribution, N_{bins} is the number of bins used to accumulate the distribution, and x_i is the value of x in the ith bin. Fewer polynomials are required than might be expected. The "wiggles" induced by a small basis set simply need to be made smaller than k_BT. However, this is a case where the barrier is quite simple to identify and an analytical treatment that removes it essentially exactly would be expected to prevail.

6.3. Parameters used for the off-lattice $\boldsymbol{\beta}$-sheet model protein. The simplified united-residue off-lattice β-sheet protein model [5, 22] has 46 pseudoatoms representing residues of three different types: hydrophobic (B), hydrophilic (L), and neutral (N). The potential energy contains bonding, bending, torsional, and nonbonded interactions:

$$\begin{aligned} V = {} & \sum_{i=2}^{46} \frac{k_{bond}}{2}(d_i - \sigma)^2 + \sum_{i=3}^{46} \frac{k_{bend}}{2}(\theta_i - \theta_0)^2 \\ & + \sum_{i=4}^{46} [A(1+\cos\phi) + B(1+\cos 3\phi)] \\ & + \sum_{i=1,\, j\geq +3}^{46} V_{XY}(r_{ij}) \qquad X, Y = B, L, \text{ or } N. \end{aligned} \tag{6.6}$$

Here, $k_{bond} = 1000\epsilon_H$ Å^{-2}, $\sigma = 1$ Å; $k_{bend} = 20\epsilon_H\ rad^{-2}$, $\theta_0 = 105°$, and the torsional potentials are of two types: if the dihedral angles involve two or more neutral residues, $A = 0$, $B = 0.2\epsilon_H$ (flexible angles); otherwise $A = B = 1.2\epsilon_H$ (rigid angles). The nonbonded interactions are bead pair specific and are given by $V_{BB} = 4\epsilon[(\sigma/r_{ij})^{12} - (\sigma/r_{ij})^6]$, $V_{LX} = (8/3)\epsilon_H[(\sigma/r_{ij})^{12} + (\sigma/r_{ij})^6]$ for $X = B$ or L, and $V_{NX} = 4\epsilon[(\sigma/r_{ij})^{12}]$ with $X = B, L$, or N.

Appendix A. Here, the dynamic contact REPSWA or DC-REPSWA transformation method is illustrated with an example for a system of atoms ($\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3, \{\mathbf{r}_{g_1}, \ldots, \mathbf{r}_{g_N}\}$) placed in a bath of monoatomic particles ($\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_M$), where all the atoms $\{\mathbf{r}_{g_1}, \ldots, \mathbf{r}_{g_N}\}$ are transformed as a rigid group around the primary dihedral ($\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3, \mathbf{r}_{g_1}$). As a result of these algorithmic steps, the initial positions ($\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3, \{\mathbf{r}_{g_1}, \ldots, \mathbf{r}_{g_N}\}$) and the corresponding forces ($\mathbf{f}_1, \mathbf{f}_2, \mathbf{f}_3, \{\mathbf{f}_{g_1}, \ldots, \mathbf{f}_{g_N}\}$) are transformed into ($\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3, \{\mathbf{u}_{g_1}, \ldots, \mathbf{u}_{g_N}\}$) and ($\mathbf{g}_1, \mathbf{g}_2, \mathbf{g}_3, \{\mathbf{g}_{g_1}, \ldots, \mathbf{g}_{g_N}\}$), respectively, defined in a new pseudo-Cartesian frame. These steps are called the positional $R \rightarrow U$ and force $FR \rightarrow FU$ transformation steps. Note that the transformation also affects the force components of the bath atoms ($\mathbf{fs}_1, \mathbf{fs}_2, \ldots, \mathbf{fs}_M$); *however, neither the bath atoms nor their force components are considered to be transformed, and they still remain in R- and FR-spaces.* As discussed in the text, the nonlinear transformation depends on certain variables Ω and $\Omega^\dagger$, which are explicitly indicated in step (*iii*) (see also main text in section 5).

$(i \uparrow)$

$$\begin{bmatrix} \mathbf{r}_1;\ \mathbf{r}'_1 = \mathbf{r}_1 \\ \mathbf{r}_2;\ \mathbf{r}'_2 = \mathbf{r}_2 \\ \mathbf{r}_3;\ \mathbf{r}'_3 = \mathbf{r}_3 \\ \mathbf{r}_{g_1};\ \mathbf{r}'_{g_1} = \mathbf{r}_{g_1} - \mathbf{r}_3 \\ . \\ . \\ \mathbf{r}_{g_N};\ \mathbf{r}'_{g_N} = \mathbf{r}_{g_N} - \mathbf{r}_3 \\ \mathbf{f}_1;\ \mathbf{f}'_1 = \mathbf{f}_1 \\ \mathbf{f}_2;\ \mathbf{f}'_2 = \mathbf{f}_2 \\ \mathbf{f}_3;\ \mathbf{f}'_3 = \mathbf{f}_3 + \sum_{k=1}^{N} \mathbf{f}_{g_k} \\ \mathbf{f}_{g_1};\ \mathbf{f}'_{g_1} = \mathbf{f}_{g_1} \\ . \\ . \\ \mathbf{f}_{g_N};\ \mathbf{f}'_{g_N} = \mathbf{f}_{g_N} \end{bmatrix} \Rightarrow \begin{bmatrix} \mathbf{r}^i_1 = \mathbf{r}'_1 \\ \mathbf{r}^i_2 = \mathbf{r}'_2 \\ \mathbf{r}^i_3 = \mathbf{r}'_3 \\ \mathbf{r}^i_{g_1} = R(\mathbf{r}'_1, \mathbf{r}'_2, \mathbf{r}'_3)\mathbf{r}'_{g_1} \\ . \\ \mathbf{r}^i_{g_N} = R(\mathbf{r}'_1, \mathbf{r}'_2, \mathbf{r}'_3)\mathbf{r}'_{g_N} \\ \mathbf{f}^i_1 = \mathbf{f}'_1 + \sum_{k=1}^{N} \mathbf{f}'_{g_k} \cdot \partial[\ R^T(\mathbf{r}^i_1, \mathbf{r}^i_2, \mathbf{r}^i_3) \cdot \mathbf{r}^i_{g_k}\ ]/\partial \mathbf{r}^i_1 \\ \mathbf{f}^i_2 = \mathbf{f}'_2 + \sum_{k=1}^{N} \mathbf{f}'_{g_k} \cdot \partial[\ R^T(\mathbf{r}^i_1, \mathbf{r}^i_2, \mathbf{r}^i_3) \cdot \mathbf{r}^i_{g_k}\ ]/\partial \mathbf{r}^i_2 \\ \mathbf{f}^i_3 = \mathbf{f}'_3 + \sum_{k=1}^{N} \mathbf{f}'_{g_k} \cdot \partial[\ R^T(\mathbf{r}^i_1, \mathbf{r}^i_2, \mathbf{r}^i_3) \cdot \mathbf{r}^i_{g_k}\ ]/\partial \mathbf{r}^i_3 \\ \mathbf{f}^i_{g_1} = R^T(\mathbf{r}'_1, \mathbf{r}'_2, \mathbf{r}'_3) \cdot \mathbf{f}'_{g_1} \\ . \\ \mathbf{f}^i_{g_N} = R^T(\mathbf{r}'_1, \mathbf{r}'_2, \mathbf{r}'_3) \cdot \mathbf{f}'_{g_N} \end{bmatrix}$$

$(ii \uparrow)$

$$\begin{bmatrix} \mathbf{r}^{ii}_1 = \mathbf{r}^i_1 \\ \mathbf{r}^{ii}_2 = \mathbf{r}^i_2 \\ \mathbf{r}^{ii}_3 = \mathbf{r}^i_3 \\ \mathbf{f}^{ii}_1 = \mathbf{f}^i_1 \\ \mathbf{f}^{ii}_2 = \mathbf{f}^i_2 \\ \mathbf{f}^{ii}_3 = \mathbf{f}^i_3 \end{bmatrix} \qquad \begin{bmatrix} \mathbf{r}^i_{g_1}(r^i_{g_1,x}, r^i_{g_1,y}, r^i_{g_1,z}) \Rightarrow \mathbf{r}^{ii}_{g_1}(r^{g_1}, \theta^{g_1}, \phi^{g_1}) \\ . \\ \mathbf{r}^i_{g_N}(r^i_{g_N,x}, r^i_{g_N,y}, r^i_{g_N,z}) \Rightarrow \mathbf{r}^{ii}_{g_N}(r^{g_N}, \theta^{g_N}, \phi^{g_N}) \\ \mathbf{f}^i_{g_1}(f^i_{g_1,x}, f^i_{g_1,y}, f^i_{g_1,z}) \Rightarrow \mathbf{f}^{ii}_{g_1}(f^{g_1}_r, f^{g_1}_\theta, f^{g_1}_\phi) \\ . \\ \mathbf{f}^i_{g_N}(f^i_{g_N,x}, f^i_{g_N,y}, f^i_{g_N,z}) \Rightarrow \mathbf{f}^{ii}_{g_N}(f^{g_N}_r, f^{g_N}_\theta, f^{g_N}_\phi) \end{bmatrix}$$

$(g \uparrow) - (iii) - (g \downarrow)$

$$\begin{bmatrix} \mathbf{r}^{ii}_{g_1}(r^{g_1}, \theta^{g_1}, \phi^{g_1}) \\ \mathbf{r}^{ii}_{g_2}(r^{g_2}, \theta^{g_2}, \phi^{g_2}) \\ . \\ \mathbf{r}^{ii}_{g_N}(r^{g_N}, \theta^{g_N}, \phi^{g_N}) \\ \mathbf{f}^{ii}_{g_1}(f^{g_1}_r, f^{g_1}_\theta, f^{g_1}_\phi) \\ \mathbf{f}^{ii}_{g_2}(f^{g_2}_r, f^{g_2}_\theta, f^{g_2}_\phi) \\ . \\ \mathbf{f}^{ii}_{g_N}(f^{g_N}_r, f^{g_N}_\theta, f^{g_N}_\phi) \end{bmatrix} \Rightarrow \begin{bmatrix} \mathbf{r}^{ii,g}_{g_1}(r^{g_1}, \theta^{g_1}, \phi^{g_1,g}) : \phi^{g_1,g} = \phi^{g_1} \\ \mathbf{r}^{ii,g}_{g_2}(r^{g_2}, \theta^{g_2}, \phi^{g_2,g}) : \phi^{g_2,g} = \phi^{g_2} - \phi^{g_1} \\ . \\ \mathbf{r}^{ii,g}_{g_N}(r^{g_N}, \theta^{g_N}, \phi^{g_N,g}) : \phi^{g_N,g} = \phi^{g_N} - \phi^{g_1} \\ \mathbf{f}^{ii,g}_{g_1}(f^{g_1}_r, f^{g_1}_\theta, f^{g_1,g}_\phi) : f^{g_1,g}_\phi = f^{g_1}_\phi + \sum_{k=2}^{N} f^{g_k}_\phi \\ \mathbf{f}^{ii,g}_{g_2}(f^{g_2}_r, f^{g_2}_\theta, f^{g_2,g}_\phi) : f^{g_2,g}_\phi = f^{g_2}_\phi \\ . \\ \mathbf{f}^{ii,g}_{g_N}(f^{g_N}_r, f^{g_N}_\theta, f^{g_N,g}_\phi) : f^{g_N,g}_\phi = f^{g_N}_\phi \end{bmatrix}$$

$$\begin{bmatrix} \begin{bmatrix} u^{ii,g}_{g_1,x} = r^{ii,g}_{g_1,x} = r^{g_1} \\ u^{ii,g}_{g_1,y} = r^{ii,g}_{g_1,y} = \theta^{g_1} \\ u^{ii,g}_{g_1,z} = \phi^{g_1,g}_u(\phi^{g_1,g}) \end{bmatrix} \\ \mathbf{u}^{ii,g}_{g_2} = \mathbf{r}^{ii,g}_{g_2} \\ . \\ \mathbf{u}^{ii,g}_{g_N} = \mathbf{r}^{ii,g}_{g_N} \\ \begin{bmatrix} g^{ii,g}_{g_1,x} = f^{ii,g}_{g_1,x} = f^{g_1}_r \\ g^{ii,g}_{g_1,y} = f^{ii,g}_{g_1,y} = f^{g_1}_\theta \\ g^{ii,g}_{g_1,z} = f^{g_1,g}_{\phi_u} \end{bmatrix} \\ \mathbf{g}^{ii,g}_{g_2} = \mathbf{f}^{ii,g}_{g_2} \\ . \\ \mathbf{g}^{ii,g}_{g_N} = \mathbf{f}^{ii,g}_{g_N} \end{bmatrix} \begin{bmatrix} \begin{bmatrix} \mathbf{u}^{ii}_1 = \mathbf{r}^{ii}_1 \\ \mathbf{u}^{ii}_2 = \mathbf{r}^{ii}_2 \\ \mathbf{u}^{ii}_3 = \mathbf{r}^{ii}_3 \end{bmatrix} (\Omega) \qquad\qquad \Omega^\dagger \\ \mathbf{s}_1 = \mathbf{s}_1 \\ . \\ \mathbf{s}_M = \mathbf{s}_M \\ \qquad f^{g_1,g}_{\phi_u} = f^{g_1,g}_{\phi} \; [\partial\phi^{g_1,g}_u/\partial\phi^{g_1,g}]^{-1} \\ \begin{bmatrix} \mathbf{g}^{ii}_1 = \mathbf{f}^{ii}_1 - f^{g_1,g}_{\phi_u} \; [\partial\phi^{g_1,g}_u/\partial\mathbf{r}_1] \\ \mathbf{g}^{ii}_2 = \mathbf{f}^{ii}_2 - f^{g_1,g}_{\phi_u} \; [\partial\phi^{g_1,g}_u/\partial\mathbf{r}_2] \\ \mathbf{g}^{ii}_3 = \mathbf{f}^{ii}_3 - f^{g_1,g}_{\phi_u} \; [\partial\phi^{g_1,g}_u/\partial\mathbf{r}_3] \end{bmatrix} (\Omega) \\ \mathbf{fs}_1 \leftarrow \mathbf{fs}_1 - f^{g_1,g}_{\phi_u} \; [\partial\phi^{g_1,g}_u/\partial\mathbf{s}_1] \\ . \\ \mathbf{fs}_M \leftarrow \mathbf{fs}_M - f^{g_1,g}_{\phi_u} \; [\partial\phi^{g_1,g}_u/\partial\mathbf{s}_M] \end{bmatrix}$$

$$\begin{bmatrix} \mathbf{u}^{ii,g}_{g_1}(r^{g_1}, \theta^{g_1}, \phi^{g_1,g}_u) \\ \mathbf{u}^{ii,g}_{g_2}(r^{g_2}, \theta^{g_2}, \phi^{g_2,g}) \\ . \\ \mathbf{u}^{ii,g}_{g_N}(r^{g_N}, \theta^{g_N}, \phi^{g_N,g}) \\ \mathbf{g}^{ii,g}_{g_1}(f^{g_1}_r, f^{g_1}_\theta, f^{g_1,g}_{\phi_u}) \\ \mathbf{g}^{ii,g}_{g_2}(f^{g_2}_r, f^{g_2}_\theta, f^{g_2,g}_{\phi}) \\ . \\ \mathbf{g}^{ii,g}_{g_N}(f^{g_N}_r, f^{g_N}_\theta, f^{g_N,g}_{\phi}) \end{bmatrix} \Rightarrow \begin{bmatrix} \mathbf{u}^{ii}_{g_1}(r^{g_1}, \theta^{g_1}, \phi^{g_1}_u) : \phi^{g_1}_u = \phi^{g_1,g} \\ \mathbf{u}^{ii}_{g_2}(r^{g_2}, \theta^{g_2}, \phi^{g_2}_u) : \phi^{g_2}_u = \phi^{g_2,g} + \phi^{g_1,g}_u \\ . \\ \mathbf{u}^{ii}_{g_N}(r^{g_N}, \theta^{g_N}, \phi^{g_N}_u) : \phi^{g_N}_u = \phi^{g_N,g}_u + \phi^{g_1,g}_u \\ \mathbf{g}^{ii}_{g_1}(f^{g_1}_r, f^{g_1}_\theta, f^{g_1}_{\phi_u}) : f^{g_1}_{\phi_u} = f^{g_1,g}_{\phi} - \sum_{k=2}^{N} f^{g_k,g}_{\phi} \\ \mathbf{g}^{ii}_{g_2}(f^{g_2}_r, f^{g_2}_\theta, f^{g_2}_{\phi_u}) : f^{g_2}_{\phi_u} = f^{g_2,g}_{\phi} \\ . \\ \mathbf{g}^{ii}_{g_N}(f^{g_N}_r, f^{g_N}_\theta, f^{g_N}_{\phi_u}) : f^{g_N}_{\phi_u} = f^{g_N,g}_{\phi} \end{bmatrix}$$

$(ii \downarrow)$

$$\begin{bmatrix} \mathbf{u}^i_1 = \mathbf{u}^{ii}_1 \\ \mathbf{u}^i_2 = \mathbf{u}^{ii}_2 \\ \mathbf{u}^i_3 = \mathbf{u}^{ii}_3 \\ \mathbf{g}^i_1 = \mathbf{g}^{ii}_1 \\ \mathbf{g}^i_2 = \mathbf{g}^{ii}_2 \\ \mathbf{g}^i_3 = \mathbf{g}^{ii}_3 \end{bmatrix} \qquad \begin{bmatrix} \mathbf{u}^{ii}_{g_1}(r^{g_1}, \theta^{g_1}, \phi^{g_1}_u) \Rightarrow \mathbf{u}^i_{g_1}(u^i_{g_1,x}, u^i_{g_1,y}, u^i_{g_1,z}) \\ . \\ \mathbf{u}^{ii}_{g_N}(r^{g_N}, \theta^{g_N}, \phi^{g_N}_u) \Rightarrow \mathbf{u}^i_{g_N}(u^i_{g_N,x}, u^i_{g_N,y}, u^i_{g_N,z}) \\ \mathbf{g}^{ii}_{g_1}(f^{g_1}_r, f^{g_1}_\theta, f^{g_1}_{\phi_u}) \Rightarrow \mathbf{g}^i_{g_1}(g^i_{g_1,x}, g^i_{g_1,y}, g^i_{g_1,z}) \\ . \\ \mathbf{g}^{ii}_{g_N}(f^{g_N}_r, f^{g_N}_\theta, f^{g_N}_{\phi_u}) \Rightarrow \mathbf{g}^i_{g_N}(g^i_{g_N,x}, g^i_{g_N,y}, g^i_{g_N,z}) \end{bmatrix}$$

$(i \downarrow)$

$$\left[\begin{array}{l}
\mathbf{u}'_1 = \mathbf{u}^i_1 \\
\mathbf{u}'_2 = \mathbf{u}^i_2 \\
\mathbf{u}'_3 = \mathbf{u}^i_3 \\
\mathbf{u}'_{g_1} = R^T(\mathbf{u}^i_1, \mathbf{u}^i_2, \mathbf{u}^i_3) \cdot \mathbf{u}^i_{g_1} \\
. \\
\mathbf{u}'_{g_N} = R^T(\mathbf{u}^i_1, \mathbf{u}^i_2, \mathbf{u}^i_3) \cdot \mathbf{u}^i_{g_N} \\
\mathbf{g}'_1 = \mathbf{g}^i_1 + \displaystyle\sum_{k=1}^{N} \mathbf{g}^i_{g_k} \cdot \partial[\, R(\mathbf{u}'_1, \mathbf{u}'_2, \mathbf{u}'_3) \cdot \mathbf{u}'_{g_k} \,]/\partial \mathbf{u}'_1 \\
\mathbf{g}'_2 = \mathbf{g}^i_2 + \displaystyle\sum_{k=1}^{N} \mathbf{g}^i_{g_k} \cdot \partial[\, R(\mathbf{u}'_1, \mathbf{u}'_2, \mathbf{u}'_3) \cdot \mathbf{u}'_{g_k} \,]/\partial \mathbf{u}'_2 \\
\mathbf{g}'_3 = \mathbf{g}^i_3 + \displaystyle\sum_{k=1}^{N} \mathbf{g}^i_{g_k} \cdot \partial[\, R(\mathbf{u}'_1, \mathbf{u}'_2, \mathbf{u}'_3) \cdot \mathbf{u}'_{g_k} \,]/\partial \mathbf{u}'_3 \\
\mathbf{g}'_{g_1} = R(\mathbf{u}^i_1, \mathbf{u}^i_2, \mathbf{u}^i_3) \cdot \mathbf{g}^i_{g_1} \\
. \\
\mathbf{g}'_{g_N} = R(\mathbf{u}^i_1, \mathbf{u}^i_2, \mathbf{u}^i_3) \cdot \mathbf{g}^i_{g_N}
\end{array}\right]
\Rightarrow
\left[\begin{array}{l}
\mathbf{u}_1 = \mathbf{u}'_1 \\
\mathbf{u}_2 = \mathbf{u}'_2 \\
\mathbf{u}_3 = \mathbf{u}'_3 \\
\mathbf{u}_{g_1} = \mathbf{u}'_{g_1} + \mathbf{u}'_3 \\
. \\
. \\
\mathbf{u}_{g_N} = \mathbf{u}'_{g_N} + \mathbf{u}'_3 \\
\mathbf{g}_1 = \mathbf{g}'_1 \\
\mathbf{g}_2 = \mathbf{g}'_2 \\
\mathbf{g}_3 = \mathbf{g}'_3 - \displaystyle\sum_{k=1}^{N} \mathbf{g}'_{g_k} \\
\mathbf{g}_{g_1} = \mathbf{g}'_{g_1} \\
. \\
. \\
\mathbf{g}_{g_N} = \mathbf{g}'_{g_N}
\end{array}\right]$$

Appendix B. In order to illustrate the application of the DC-REPSWA transformation to a general molecule with N_t torsional dihedrals, the molecule is topologically decomposed into transformation groups $\Upsilon_t(k), k = 1, \ldots, N_t$, where $\Upsilon_t(k) = \{\mathbf{r}^k_{g_1}, \ldots, \mathbf{r}^k_{g_N}\}$. Furthermore, each transformation group belongs to a dihedral group defined as $\Upsilon_d(k) = \{\mathbf{r}^k_1, \mathbf{r}^k_2, \mathbf{r}^k_3, \{\mathbf{r}^k_{g_1}, \ldots, \mathbf{r}^k_{g_N}\}\}$, and the individual levels denoted by the index k are strictly ordered. Note that the topological decomposition does not have to follow chemical connectivity; however, $\Upsilon_t(i) \cap \Upsilon_t(j) = \emptyset$ is strictly imposed. On the other hand, in most of the cases $\Upsilon_d(i) \cap \Upsilon_d(j) \neq \emptyset$. The action of the transformation steps described in Appendix A are successively carried out on the dihedral groups. *The base atoms of the first level* $\mathbf{r}^1_1, \mathbf{r}^1_2, \mathbf{r}^1_3$ *are the reference atoms, which always remain intact.*

```
$(R, FR) \rightarrow (U, FU)$ [1*]
    $\Upsilon_U = \{\ \}$
    $\Upsilon_R = \{$all atoms in the system$\}$
    for $k \leftarrow N_t$ to 1
        do $\Upsilon_d(k) \leftarrow (i \uparrow)\ [\Upsilon_d(k)]$
            $\Upsilon_d(k) \leftarrow (ii \uparrow)\ [\Upsilon_d(k)]$
            $\Upsilon_U \leftarrow \Upsilon_U \cup \Upsilon_t(k)$
            $\Upsilon_R \leftarrow \Upsilon_R \setminus \Upsilon_t(k)$
            $\Upsilon_d(k) \leftarrow (g \uparrow)\ [\Upsilon_d(k)]$
            $\Upsilon_d(k) \leftarrow (iii)\ [\Upsilon_d(k)]$
                        $\Omega^\dagger \setminus \Omega = \Upsilon_R \setminus \{\mathbf{r}^k_1, \mathbf{r}^k_2, \mathbf{r}^k_3\}$ [2*]
            $\Upsilon_d(k) \leftarrow (g \downarrow)\ [\Upsilon_d(k)]$
    endfor
    for $k \leftarrow 1$ to $N_t$
        do $\Upsilon_d(k) \leftarrow (ii \downarrow)\ [\Upsilon_d(k)]$
            $\Upsilon_d(k) \leftarrow (i \downarrow)\ [\Upsilon_d(k)]$
    endfor
```

```
        $\Upsilon_U = \bigcup_{k=1}^{N_t} \Upsilon_t(k)$
        $\Upsilon_R = \{\textit{all atoms in the system}\} \setminus \bigcup_{k=1}^{N_t} \Upsilon_t(k)$
```

[1*]: forces originating from $V_{\text{ref}}^{\text{eff}}$ (3.13) are already present in R-space

[2*]: force field dependent, e.g., all atom: $\Omega^{\dagger} \setminus \Omega = \Upsilon_R \setminus \{\mathbf{r}_2^k$, atoms bonded to $\mathbf{r}_2^k\}$

```
    $U \rightarrow R$ [1*]
        $\Upsilon_U = \bigcup_{k=1}^{N_t} \Upsilon_t(k)$
        $\Upsilon_R = \{\textit{all atoms in the system}\} \setminus \bigcup_{k=1}^{N_t} \Upsilon_t(k)$
        for $k \leftarrow N_t$ to 1
            do $\Upsilon_d(k) \leftarrow (i \uparrow)\ [\Upsilon_d(k)]$
                $\Upsilon_d(k) \leftarrow (ii \uparrow)\ [\Upsilon_d(k)]$
        endfor
        for $k \leftarrow 1$ to $N_t$
            do $\Upsilon_d(k) \leftarrow (g \uparrow)\ [\Upsilon_d(k)]$
                $\Upsilon_d(k) \leftarrow (iii)^{-1}\ [\Upsilon_d(k)]$ [2*]
                            $\Omega^{\dagger} \setminus \Omega = \Upsilon_R \setminus \{\mathbf{r}_1^k, \mathbf{r}_2^k, \mathbf{r}_3^k\}$  [3*]
                $\Upsilon_d(k) \leftarrow (g \downarrow)\ [\Upsilon_d(k)]$
                $\Upsilon_U \leftarrow \Upsilon_U \setminus \Upsilon_t(k)$
                $\Upsilon_R \leftarrow \Upsilon_R \cup \Upsilon_t(k)$
                $\Upsilon_d(k) \leftarrow (ii \downarrow)\ [\Upsilon_d(k)]$
                $\Upsilon_d(k) \leftarrow (i \downarrow)\ [\Upsilon_d(k)]$
        endfor
        $\Upsilon_U = \{\ \}$
        $\Upsilon_R = \{\textit{all atoms in the system}\}$
```

[1*]: only position transformation no forces

[2*]: inverting one-dimensional torsional transform using (2.16)

[3*]: force field dependent, e.g., all atom: $\Omega^{\dagger} \setminus \Omega = \Upsilon_R \setminus \{\mathbf{r}_2^k$, atoms bonded to $\mathbf{r}_2^k\}$

Appendix C. Further implementation details for SDC-REPSWA. In order to implement the staging dynamic contact or SDC-REPSWA algorithm, each transformation group of Appendix B, $\Upsilon_t(k) = \{\mathbf{r}_{g_1}^k, \ldots, \mathbf{r}_{g_N}^k\}$, is assigned a specific label, and this is kept in a variable array $Trans\_Type = \{Trans\ \_Type(k) : k = 1, \ldots, N_t\}$ indicating the type of transformation performed on that particular group, e.g., $Trans\_Type(k) = \{$"*dynamic*," "*static*," "*identity*,"$\ldots\}$. In this appendix, it is shown how to drive SDC-REPSWA with an HMC scheme.

```
    $HMC - SDC - REPSWA$
        $R \leftarrow R_0,\ FR \leftarrow FORCE(R)$
        $FR \leftarrow FR + FR_{\text{ref}}$
        $(R, U) \rightarrow\ (FR, FU)$
        $P(V) \leftarrow P_{Maxwell-Boltzmann}(T)$
        for $k \leftarrow 1$ to $nstep$
            do $STATE \leftarrow (R, U, FR, FU)$
                $H_{pre}(U) \leftarrow V_{\text{eff}}(U) + K(V)$
                $(U, V) \leftarrow integrate\_UV(U, FU, V)$
                $U \rightarrow R \rightarrow [FR \leftarrow FORCE(R)]$
                $FR \leftarrow FR + FR_{\text{ref}}$
                $FR \rightarrow FU$
            $V \leftarrow integrate\_V(FU, V)$
            $H_{post}(U) \leftarrow V_{\text{eff}}(U) + K(V)$
            $P_{acc} = MIN(1.0, \exp(-(H_{post} - H_{pre})/(k_B\,T)))$
            **if** $P_{acc} > Rand\_Unif[0,1]$
                **if** $(k\ mod\ N_{mod}) == 0$ [*]
                    $Trans\_Type \leftarrow update$
                    $FR \leftarrow FR + FR_{\text{ref}}$
                    $(R, U) \rightarrow (FR, FU)$
                    $P(V) \leftarrow P_{Maxwell-Boltzmann}(T)$
                **endif**
            **else**
                $(R, U, FR, FU) \leftarrow STATE$
                $P(V) \leftarrow P_{Maxwell-Boltzmann}(T)$
            **endif**
    **endfor**
```

[*]:N_{mod} *is the frequency of updating* $Trans\_Type$

```
$(U, V) \leftarrow integrate\_UV(U, FU, V)$
     $U = \{\mathbf{u}_1, \ldots, \mathbf{u}_{N_{atom}}\}$
    $V = \{\mathbf{v}_1, \ldots, \mathbf{v}_{N_{atom}}\}$
    $FU = \{\mathbf{g}_1, \ldots, \mathbf{g}_{N_{atom}}\}$

     **for** $i \leftarrow 1$ **to** $N_{atom}$
         **do** $vx[i] \leftarrow vx[i] + gx[i]/mass[i]\ dt/2$
             $vy[i] \leftarrow vy[i] + gy[i]/mass[i]\ dt/2$
             $vz[i] \leftarrow vz[i] + gz[i]/mass[i]\ dt/2$

             $ux[i] \leftarrow vx[i]\ dt$
             $uy[i] \leftarrow vy[i]\ dt$
             $uz[i] \leftarrow vz[i]\ dt$
    **endfor**

$V \leftarrow integrate\_V(FU, V)$
     $V = \{\mathbf{v}_1, \ldots, \mathbf{v}_{N_{atom}}\}$
    $FU = \{\mathbf{g}_1, \ldots, \mathbf{g}_{N_{atom}}\}$

     **for** $i \leftarrow 1$ **to** $N_{atom}$
         **do** $vx[i] \leftarrow vx[i] + gx[i]/mass[i]\ dt/2$
             $vy[i] \leftarrow vy[i] + gy[i]/mass[i]\ dt/2$
             $vz[i] \leftarrow vz[i] + gz[i]/mass[i]\ dt/2$
    **endfor**
```

Note, that due to the aggressive nature of the transformation, the system could visit regions in U-space, where P_{acc} could become very small. In this case, it is sufficient to leave U-space for a few steps and propagate the system further in R-space using $H(P, R) = K(P) + V(R)$ for a while. In other words, at every step there is a possibility of moving in R-space or in U-space based on $P_{acc}(H(R))$ or $P_{acc}(H(U))$. In order to achieve maximum accuracy, optimization of the latter option should be further investigated. In addition, it is possible to mix U or R propagation by monitoring the relation between $P_{acc}(H(R))$ and $P_{acc}(H(U))$ on the fly. The efficiency of the latter approach is currently being tested.

Acknowledgments. Peter Minary would like to acknowledge Michael Levitt for useful discussions and funding and Jesus Izaguirre for carefully reading this manuscript.

REFERENCES

[1] W. Cornell, P. Cieplak, C. Bayly, I. Gould, K. Merz, D. Ferguson, D. Spellmeyer, T. Fox, J. Caldwell, and P. Kollman, *A second generation force field for the simulation of proteins, nucleic acids and organic molecules*, J. Am. Chem. Soc., 117 (1995), pp. 5179–5197.

[2] S. Duane, A. D. Kennedy, B. J. Pendleton, and D. Roweth, *Hybrid Monte Carlo*, Phys. Lett. B, 195 (1987), pp. 216–222.

[3] C. J. Geyer, *Markov Chain Monte Carlo Maximum Likelihood*, in Computing Science and Statistics: Proceedings of the 23rd Symposium on the Interface, 1991, pp. 156–163.

[4] U. H. E. Hansmann, *Parallel tempering algorithm for conformational studies of biological molecules*, Chem. Phys. Lett., 281 (1997), pp. 140–150.

[5] J. D. Honeycutt and D. Thirumalai, *Metastability of the folded states of globular proteins*, Proc. Natl. Acad. Sci. USA, 87 (1990), pp. 3526–3629.

[6] A. Kone and D. A. Kofke, *Selection of temperature intervals for parallel-tempering simulations*, J. Chem. Phys., 122 (2005), 206101.

[7] A. Laio and M. Parrinello, *Escaping the free-energy minima*, Proc. Natl. Acad. Sci. USA, 99 (2002), pp. 12562–12566.

[8] A. Laio, A. Rodriguez-Fortea, F. L. Gervasio, M. Ceccarelli, and M. Parrinello, *Assessing the accuracy of metadynamics*, J. Phys. Chem. B, 109 (2005), pp. 6714–6721.

[9] M. Levitt and S. Lifson, *Refinement of protein conformations using a macromolecular energy minimization procedure*, J. Mol. Biol., 46 (1969), pp. 269–279.

[10] Z. Li and H. A. Scheraga, *Monte Carlo-minimization approach to the multiple-minima problem in protein folding*, Proc. Natl. Acad. Sci. USA, 84 (1987), pp. 6611–6615.

[11] P. Liu, X. H. Huang, R. H. Zhou, and B. J. Berne, *Hydrophobic aided replica exchange: An efficient algorithm for protein folding in explicit solvent*, J. Phys. Chem. B, 110 (2006), pp. 19018–19022.

[12] P. Liu, B. Kim, R. A. Friesner, and B. J. Berne, *Replica exchange with solute tempering: A method for sampling biological systems in explicit water*, Proc. Natl. Acad. Sci. USA, 102 (2005), pp. 13749–13754.

[13] E. O. M. C. Tesi, E. J. J. van Rensburg, and S. G. Whittington, *Monte Carlo study of the interacting self-avoiding walk model in three dimensions*, J. Stat. Phys., 82 (1996), pp. 155–181.

[14] A. D. MacKerell, D. Bashford, M. Bellott, R. L. Dubrack, J. M. Evanseck, M. J. Field, S. Fischer, J. Gao, H. Guo, S. Ha, D. Joseph-McCarthy, L. Kcuhnir, K. Kuczera, F. T. K. Lau, C. Mattos, S. Michnick, T. Ngo, D. T. Nguyen, B. Prodhom, W. E. Reiher, III, B. Roux, M. Schlenkrich, J. C. Smith, R. Stote, J. Straub, M. Watanabe, J. Wiorkiewicz-Kuczera, D. Yin, and M. Karplus, *All-atom empirical potential for molecular modeling and dynamics studies of proteins*, J. Phys. Chem. B, 102 (1998), pp. 3586–3616.

[15] E. Marinari and G. Parisi, *Simulated tempering: A new Monte Carlo scheme*, Europhys. Lett., 19 (1992), pp. 451–458.

[16] M. G. Martin and J. I. Siepmann, *Transferable potentials for phase equilibria*. 1. *United-atom description of n-alkanes*, J. Phys. Chem. B, 102 (1998), pp. 2569–2577.

[17] T. Me, Y. Da, S. So, and G. Martyna, *Exploiting multiple levels of parallelism in molecular dynamics based calculations via modern techniques and software paradigms on distributed memory computers*, Comp. Phys. Commun., 128 (2000), pp. 333–376.

[18] M. A. Miller and D. J. Wales, *Energy landscape of a model protein*, J. Chem. Phys., 111 (1999), pp. 6610–6616.

[19] A. Mitsutake, Y. Sugita, and Y. Okamoto, *Generalized-ensemble algorithms for molecular simulations of biopolymers*, Biopolymers, 60 (2001), pp. 96–123.

[20] J. A. Rahman and J. C. Tully, *Puddle-jumping: A flexible sampling algorithm for rare event systems*, Chem. Phys., 285 (2002), pp. 277–287.

[21] J. A. Rahman and J. C. Tully, *Puddle-skimming: An efficient sampling of multidimensional configuration space*, J. Chem. Phys., 116 (2002), pp. 8750–8760.

[22] J. M. Sorenson and T. Head-Gordon, *Redesigning the hydrophobic core of a model beta-sheet protein: Destabilizing traps through a threading approach*, Proteins: Struct. Funct. Genet., 37 (1999), pp. 582–591.

[23] C. Summa and M. Levitt, *Near-native structure refinement using in vacuo energy minimization*, Proc. Natl. Acad. Sci. USA, 104 (2007), pp. 3177–3182.

[24] S. Tanaka and H. Scheraga, *Model of protein folding: Inclusion of short-, medium-, and long-range interactions*, Proc. Natl. Acad. Sci. USA, 72 (1975), pp. 3802–3806.

[25] G. M. Torrie and J. P. Valleau, *Monte-Carlo free-energy estimates using non-Boltzmann sampling: Application to the subcritical Lennard-Jones fluid*, Chem. Phys. Lett., 28 (1974), pp. 578–581.

[26] G. M. Torrie and J. P. Valleau, *Non-physical sampling distributions in Monte-Carlo free-energy estimation: Umbrella sampling*, J. Comput. Phys., 23 (1977), pp. 187–199.

[27] A. F. Voter, *Hyperdynamics: Accelerated molecular dynamics of infrequent events*, Phys. Rev. Lett., 78 (1997), pp. 3908–3911.

[28] D. J. Wales, *Energy landscapes and properties of biomolecules*, Phys. Biol., 2 (2005), pp. S86–S93.

[29] D. J. Wales and J. P. K. Doye, *Global optimization by basin-hopping and the lowest energy structures of Lennard-Jones clusters containing up to* 110 *atoms*, J. Phys. Chem. A, 101 (1997), pp. 5111–5116.

[30] M. Watanabe and W. L. Reinhardt, *Direct dynamical calculation of entropy and free energy by adiabatic switching*, Phys. Rev. Lett., 65 (1990), pp. 3301–3304.

[31] Z. Zhu, M. Tuckerman, S. O. Samuelson, and G. Martyna, *Using novel variable transformations to enhance conformational sampling in molecular dynamics*, Phys. Rev. Lett., 88 (2002), 100201.

[32] R. W. Zwanzig, *High-temperature equation of state by a perturbation method*. I. *Nonpolar gases*, J. Chem. Phys., 22 (1954), pp. 1420–1426.