

# Robust Biased Brownian Dynamics for Rate Constant Calculation

Gang Zou\* and Robert D. Skeel†

\*Renaissance Technologies, East Setauket, New York; and †Department of Computer Science and Beckman Institute, University of Illinois, Urbana, Illinois

**ABSTRACT** A reaction probability is required to calculate the rate constant of a diffusion-dominated reaction. Due to the complicated geometry and potentially high dimension of the reaction probability problem, it is usually solved by a Brownian dynamics simulation, also known as a random walk or path integral method, instead of solving the equivalent partial differential equation by a discretization method. Building on earlier work, this article completes the development of a robust importance sampling algorithm for Brownian dynamics—i.e., biased Brownian dynamics with weight control—to overcome the high energy and entropy barriers in biomolecular association reactions. The biased Brownian dynamics steers sampling by a bias force, and the weight control algorithm controls sampling by a target weight. This algorithm is optimal if the bias force and the target weight are constructed from the solution of the reaction probability problem. In reality, an approximate reaction probability has to be used to construct the bias force and the target weight. Thus, the performance of the algorithm depends on the quality of the approximation. Given here is a method to calculate a good approximation, which is based on the selection of a reaction coordinate and the variational formulation of the reaction probability problem. The numerically approximated reaction probability is shown by computer experiments to give a factor-of-two speedup over the use of a purely heuristic approximation. Also, the fully developed method is compared to unbiased Brownian dynamics. The tests for human superoxide dismutase, *Escherichia coli* superoxide dismutase, and antisweetener antibody NC6.8, show speedups of 17, 35, and 39, respectively. The test for reactions between two model proteins with orientations shows speedups of 2578 for one set of configurations and 3341 for another set of configurations.

## INTRODUCTION

This article considers an elliptic partial differential equation for reaction probability in high dimensions and constructs path integral, or random walk, methods for solving it. The methods are oriented toward the calculation of rate constants for diffusion-limited reactions. The challenge of the random walk method is variance reduction. An importance sampling method for random walk methods, biased Brownian dynamics, is proposed in an earlier article (Zou et al., 2000). That method makes use of weighted averages, but unfortunately the possibility of large weights sometimes cancels the benefit of importance sampling. To avoid this problem, a method of weight control has been developed (Zou, 2002) and submitted to *SIAM J. Sci. Comput.* for publication (G. Zou and R. D. Skeel, “Robust Variance Reduction for Random Walk Methods,” <http://bionum.cs.uiuc.edu/p.html>). The main contribution of this present work is to give an effective and systematic technique to approximate the reaction probability, from which the bias force in biased Brownian dynamics and the target weight in weight control are constructed. We demonstrate the effectiveness of the algorithm with four numerical tests.

The application considered here is the rate constant computation for diffusion-limited reactions, for which the encounter of the reactants is the time-limiting stage of the reaction. Biochemical reactions such as enzyme-substrate

and antibody-antigen association fit well in this category. Brownian dynamics (BD) simulation is widely used in this area (Antosiewicz et al., 1995; Brune and Kim, 1994; Getzoff et al., 1992; Guddat et al., 1994; Kozack et al., 1995; Northrup et al., 1993; Wade et al., 1994). Northrup, Allison, and McCammon (NAM) developed a method for rate constant calculation for diffusion-limited reactions in 1984. The NAM method connects the reaction rate with a reaction probability, which is the solution of the Smoluchowski equation, an elliptic partial differential equation (PDE), in a finite domain (Zhou, 1990). Solving the PDE by a discretization method is difficult due to the possible high dimension and the complicated geometry of configuration space. Instead, the random walk method is used, which involves the calculation of the expectation of an exit value for a stochastic differential equation (SDE). A BD simulation code for biomolecules, University of Houston Brownian Dynamics (UHBD), that calculates biomolecular rates of association with the NAM method, is distributed by McCammon and co-workers (Davis et al., 1991). Our rate constant computation is based mainly on the NAM method and UHBD.

Standard BD samples only a few reacted trajectories, especially for problems with high dimensions or with high energy barriers. Thus it gives a large variance for the random exit value. To reduce the variance, an enhanced sampling method—weighted ensemble Brownian dynamics (WEBD)—was developed by Huber and Kim (1996). WEBD maintains an ensemble of particles (in configuration space) and associates a probability weight with each particle. All particles carry out Brownian motion without influencing each other. Periodically, WEBD splits and merges particles

Submitted March 15, 2003, and accepted for publication June 4, 2003.

Address reprint requests to Robert D. Skeel, 1304 W. Springfield Ave., 2316 DCL, University of Illinois, Urbana, IL 61801. Tel.: 217-333-2727; E-mail: [skeel@cs.uiuc.edu](mailto:skeel@cs.uiuc.edu).

© 2003 by the Biophysical Society

0006-3495/03/10/2147/11 \$2.00

according to their position and weight. This procedure enables better sampling in regions inaccessible to standard BD. Thus, particles are sampled more frequently near reaction regions and the variance is reduced. WEBD was originally proposed for use with the flux-over-population method (Hänggi et al., 1990, p. 258), but it has also been combined with the NAM method (Rojnuckarin et al., 2000).

However, WEBD is a fairly complicated algorithm, and biased BD (Zou et al., 2000) is offered as a simpler, and delightfully parallel, alternative. Biased BD does importance sampling by 1), introducing a bias force in addition to the existing force; and 2), associating a weight with the particle. The expectation of the weighted exit value—the product of the exit value and the weight at the exit time—gives an unbiased estimate of the reaction probability of standard BD. The variance of the weighted exit value is related to the bias force. It turns out there is an optimal bias force that makes the variance of the weighted exit value 0, but in practice a rough approximation must be used.

One difficulty that biased BD faces is that the particle weight can grow without bound due to the imperfect choice of the bias force and the approximation in the numerical integration. The large weight fluctuations result in a large variance of the weighted exit value. Under the optimal bias force, the particle has an ideal weight, which is not trajectory-dependent, but a function only of the current configuration. This is the basis of a weight control algorithm (Zou, 2002), which uses a target weight and a tolerance range to force the particle weight into a range. The weight control algorithm operates as follows. At the beginning of each time step, the particle weight is checked. If the weight exceeds the upper limit, the particle splits into two, each with one-half the weight, and they are simulated independently. The final result is the sum of the two weighted exit values. Conversely, if the weight falls below the lower limit, the weight is doubled with 50% probability and zeroed out (and the trajectory abandoned) otherwise. Thus, the expectation of the exit value is unchanged.

In practice, an approximate reaction probability function is used to construct the bias force and the target weight. Hence, the quality of the approximation determines the performance of the algorithm. Given in the section “Construction of bias force” of this article is a method to calculate a good approximation, based on the selection of a reaction coordinate and the variational formulation of the reaction probability problem.

The benefit of using a partly numerical rather than a purely heuristic estimate of reaction probability is investigated for a model of *Escherichia coli* superoxide dismutase (SOD) 1eso. The speedup is 1.8 without weight control and 2.0 with it. Also, the fully developed method is compared to unbiased Brownian dynamics for this and two other systems. A *Homo sapiens* SOD 1spd test shows a speedup of 17, the SOD 1eso test shows a speedup of 35, and an antisweetener antibody NC6.8 test shows a speedup of 39.

Results are also obtained for a more difficult test problem of a reaction between two model proteins with orientations. The speedup due to a partly numerical estimate of reaction probability is 2.3 without weight control and 1.3 with it. The speedup for the fully developed method over unbiased Brownian dynamics is a factor of 2578 for one set of configurations and a factor of 3341 for another set.

## Discussion

The partly numerical estimate of the reaction probability gives a worthwhile speedup and may serve as a safeguard against a poor heuristic estimate. The overall method is shown by numerical experiments to be a great success and as fast as WEBD. The relative simplicity of biased BD makes implementation easier and facilitates analysis and improvement of the method. In any case, these dramatic reductions in the cost of calculating rate constants make it practical to employ more detailed models of the biomolecular system and thus to close the gap between experiment and computation.

## RATE CONSTANTS OF DIFFUSION-LIMITED REACTIONS

### Equations of motion

Consider two types of molecules, enzyme and substrate, diffusing in solvent. Assume dilute concentrations, and, therefore, consider the movement between a pair of molecules. Model the enzyme by a set of atoms that moves like a rigid body, and neglect its rotational movement. Each atom has a partial charge and van der Waals parameter. Model the substrate by a set of spherical subunits with partial charges and van der Waals parameters and with subunits connected by constraints or bonded forces. Neglect the rotational movement of each subunit. Electrostatic and excluded volume forces act between subunits of the substrate, and between the substrate and the enzyme.

Let  $\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_N$  be the coordinates of the subunits of the substrate relative to the geometry center of all atoms of the enzyme. These coordinates describe a configuration of the system, represented by a column vector  $\mathbf{R}$  of dimension  $3N$ . Specify a set of reacted configurations  $\Omega_{rc}$  such that if the particle diffuses to the reaction surface  $\partial\Omega_{rc}$ , a reaction happens and motion terminates. Assume, as in UHBD, that reaction happens if, and only if, a certain number of distance criteria are satisfied, say,  $m$  out of  $n$  distances are less than a distance  $\xi_{rc}$ .

Define a potential energy function  $U(\mathbf{R})$  and a  $3N \times 3N$  symmetric positive definite diffusion tensor  $D(\mathbf{R})$ . The movement of the substrate relative to the enzyme is described by a stochastic differential equation

$$d\mathbf{R} = \left( (\nabla^T D(\mathbf{R}))^T + D(\mathbf{R}) \frac{\mathbf{F}(\mathbf{R})}{k_B T} \right) dt + \sqrt{2} D_{1/2}(\mathbf{R}) d\mathbf{W}(t), \quad (1)$$

where  $D_{1/2}$  satisfies  $D_{1/2}D_{1/2}^T = D$ ,  $\nabla$  is the column vector of  $3N$  partial derivative operators,  $\mathbf{F}(\mathbf{R}) = -\nabla U(\mathbf{R})$  are the forces,  $\mathbf{W}(t)$  is a  $3N$ -dimensional canonical Wiener process,  $k_B$  is the Boltzmann constant, and  $T$  is the temperature. A canonical Wiener process  $\mathbf{W}(t)$  has a Gaussian distribution with mean zero and covariance  $EW_i(s)W_j(t) = \min\{s, t\}\delta_{ij}$ . A typical choice for the tensor  $D(\mathbf{R})$  is the Rotne-Prager tensor, whose diagonal blocks are

$$D_{ii} = \left( \frac{k_B T}{6\pi\eta a_i} + \frac{k_B T}{6\pi\eta a_{\text{mol}}} \right) I, \quad (2)$$

and whose off-diagonal blocks are

$$D_{ij} = \frac{k_B T}{8\pi\eta r_{ij}} \left[ \left( I + \frac{\mathbf{r}_{ij}\mathbf{r}_{ij}^T}{r_{ij}^2} \right) + \frac{a_i^2 + a_j^2}{r_{ij}^2} \left( \frac{1}{3}I - \frac{\mathbf{r}_{ij}\mathbf{r}_{ij}^T}{r_{ij}^2} \right) \right], \quad (3)$$

where  $\eta$  is the viscosity of the solvent,  $a_i$  is the radius of the  $i^{\text{th}}$  subunit,  $a_{\text{mol}}$  is the radius of the enzyme, and  $I$  is the  $3 \times 3$  identity matrix.

### Rate constant calculation

Select a center  $\mathbf{r}_c$  for the substrate. In UHBD,  $\mathbf{r}_c$  is the geometry center of all subunits. The rate constant  $k$  is approximated by the formula (Northrup et al., 1984)

$$\frac{1}{k} = \frac{1}{\bar{\beta}(b)} \int_b^q dr \frac{\exp(U_{\text{ext}}(r)/(k_B T))}{4\pi r^2 d(r)} + \int_q^{+\infty} dr \frac{\exp(U_{\text{ext}}(r)/(k_B T))}{4\pi r^2 d(r)}. \quad (4)$$

where  $b < q$  are two values for  $|\mathbf{r}_c|$ ,  $d(|\mathbf{r}_c|)$  is the diffusion coefficient of the center  $\mathbf{r}_c$ , and  $\bar{\beta}(b)$  is defined shortly. Assume that the  $3N$ -dimensional set defined by  $|\mathbf{r}_c| \leq b$  contains  $\Omega_{\text{rc}}$ , and let  $\Omega_q$  be the  $3N$ -dimensional set defined by  $|\mathbf{r}_c| \leq q$ . For a high-dimensional problem, UHBD computes  $d(|\mathbf{r}_c|)$  during the BD simulation by collecting the mobility information of the center  $\mathbf{r}_c$ . The term  $U_{\text{ext}}(|\mathbf{r}_c|)$  is usually computed by treating the enzyme as a Debye-Hückel sphere with its net charge at the center and treating the substrate as a point charge.

The value  $\bar{\beta}(b)$  is defined in terms of a reaction probability  $\beta(\mathbf{R})$ , which satisfies the PDE and boundary condition (Zhou, 1990),

$$\left( \nabla_{\mathbf{R}}^T D(\mathbf{R}) \nabla_{\mathbf{R}} + \frac{\mathbf{F}(\mathbf{R})^T}{k_B T} D(\mathbf{R}) \nabla_{\mathbf{R}} \right) \beta(\mathbf{R}) = 0, \quad \mathbf{R} \in \Omega, \\ \beta(\mathbf{R}) = f(\mathbf{R}), \quad \mathbf{R} \in \partial\Omega, \quad (5)$$

where  $\nabla_{\mathbf{R}}$  acts on all that follows it, the domain  $\Omega = \Omega_q \setminus \Omega_{\text{rc}}$ , and the function  $f(\mathbf{R})$  is defined to be 0 on the escape surface  $\partial\Omega_q$  and 1 on the reaction surface  $\partial\Omega_{\text{rc}}$ .

In terms of SDEs, the reaction probability  $\beta(\mathbf{R})$  is (see Appendix) the path integral

$$\beta(\mathbf{R}) = E_{0,\mathbf{R}} f(\mathbf{R}(\tau_{\Omega})),$$

where  $\mathbf{R}(t)$  is a trajectory of the Brownian particle governed by Eq. 1 with initial condition  $\mathbf{R}(0) = \mathbf{R}$ ,  $\tau_{\Omega}$  is the first exit time from domain  $\Omega$ , and  $E_{0,\mathbf{R}}$  is the expectation with respect to the probability law for the random trajectory  $\mathbf{R}(t)$  starting at  $\mathbf{R}(0) = \mathbf{R}$ .

The average reaction probability  $\bar{\beta}(b)$  on the  $b$ -surface for high-dimensional problems is taken as the average of  $\beta(\mathbf{R})$  with respect to some distribution  $p_b(\mathbf{R})$  on the  $b$ -surface, in which the center  $\mathbf{r}_c$  is uniformly distributed on the  $|\mathbf{r}_c| = b$  spherical surface and conformations are Boltzmann-distributed.

In practice, Eq. 1 is approximated by the scheme

$$\mathbf{R}^{n+1} = \mathbf{R}^n + \left( (\nabla^T D(\mathbf{R}^n))^T + D(\mathbf{R}^n) \frac{\mathbf{F}(\mathbf{R}^n)}{k_B T} \right) \Delta t \\ + \sqrt{2} D_{1/2}(\mathbf{R}^n) \sqrt{\Delta t} \mathbf{Z}^{n+1}, \quad (6)$$

where  $\Delta t$  is the time step and  $\mathbf{Z}^{n+1}$  is a vector of  $3N$  independent standard Gaussian random numbers (of mean 0 and variance 1).

### BIASED BROWNIAN DYNAMICS WITH WEIGHT CONTROL

The expectation of  $f(\mathbf{R}(\tau_{\Omega}))$  can be estimated with smaller relative statistical error if reactions are more numerous. This motivates the addition of a bias force  $\mathbf{F}_b(\mathbf{R})$ :

$$\mathbf{R}^{n+1} = \mathbf{R}^n + \left( (\nabla^T D(\mathbf{R}^n))^T + D(\mathbf{R}^n) \frac{\mathbf{F}(\mathbf{R}^n) + \mathbf{F}_b(\mathbf{R}^n)}{k_B T} \right) \Delta t \\ + \sqrt{2} D_{1/2}(\mathbf{R}^n) \sqrt{\Delta t} \mathbf{Z}^{n+1}. \quad (7)$$

The bias thus introduced is compensated for perfectly if the exit value is multiplied by the appropriate weight: the weight associated with  $\mathbf{R}^n$  is expressed as  $\exp u^n$  where  $u^0 = 0$  and

$$u^{n+1} = u^n - \frac{1}{4} \frac{\mathbf{F}_b(\mathbf{R}^n)^T}{k_B T} D(\mathbf{R}^n) \frac{\mathbf{F}_b(\mathbf{R}^n)}{k_B T} \Delta t \\ - \frac{\sqrt{2}}{2} \frac{\mathbf{F}_b(\mathbf{R}^n)^T}{k_B T} D_{1/2}(\mathbf{R}^n) \sqrt{\Delta t} \mathbf{Z}^{n+1}. \quad (8)$$

In the continuum limit  $\Delta t \rightarrow 0$ ,  $E_{0,\mathbf{R}} \exp u(\tau_{\Omega}) f(\mathbf{R}(\tau_{\Omega})) = \beta(\mathbf{R})$ , where  $u(t)$  satisfies the  $\Delta t \rightarrow 0$  limit of Eq. 8 (Milstein, 1988).

The purpose of the bias force is to reduce the variance of the estimate. Remarkably, in the continuum limit  $\Delta t \rightarrow 0$ , there is an optimal bias force,

$$\mathbf{F}_b(\mathbf{R}) = 2k_B T \nabla_{\mathbf{R}} \ln \beta(\mathbf{R}), \quad (9)$$

that reduces the variance to zero (Milstein, 1988). In practice, the bias force is constructed from an estimate of  $\beta(\mathbf{R})$ .

In the continuum limit  $\Delta t \rightarrow 0$ , the use of the optimal bias force produces an ideal weight

$$\exp u(t) = \beta(\mathbf{R}(0)) / \beta(\mathbf{R}(t)), \quad (10)$$

that does not depend on the history of the trajectory. An estimate  $\tilde{\beta}(\mathbf{R})$  of  $\beta(\mathbf{R})$  can be used to define a target weight and an acceptable range of weights, and this can be used to control the weight as described in the Introduction. The numerical tests reported here use the range  $u^*(\mathbf{R}^n) - 2 \leq u^n \leq u^*(\mathbf{R}^n) + 1$ .

The actual calculation is that of an average  $\bar{\beta}(b)$  on the  $b$ -surface, not that of a value  $\beta(\mathbf{R}^0)$  at a single point. A better target weight is given by

$$\exp u^*(\mathbf{R}) = \tilde{\beta}_b / \tilde{\beta}(\mathbf{R}), \quad (11)$$

where  $\tilde{\beta}_b$  approximates  $\bar{\beta}(b)$ .

## CONSTRUCTION OF BIAS FORCE

Biased BD with weight control requires a bias force and a target weight, which can be constructed from an approximate solution  $\tilde{\beta}(\mathbf{R})$  using Eqs. 9 and 10. In particular, the bias force is

$$\mathbf{F}_b(\mathbf{R}) = 2k_B T \nabla_{\mathbf{R}} \ln \tilde{\beta}(\mathbf{R}), \quad (12)$$

which is computed by one-sided finite differences applied to  $\tilde{\beta}(\mathbf{R})$ .

The method proposed here for computing  $\tilde{\beta}(\mathbf{R})$  uses a reaction coordinate  $\xi = \xi(\mathbf{R})$  that is easy to evaluate for every configuration  $\mathbf{R}$  and that correlates well with  $\beta$ , in particular, 1), there is a value  $\xi_{rc}$  for which the set of configurations with  $\xi(\mathbf{R}) < \xi_{rc}$  is exactly  $\Omega_{rc}$ ; and 2), there is a value  $\xi_q$  for which the set of configurations with  $\xi(\mathbf{R}) \leq \xi_q$  encloses  $\Omega_q$  and equals  $\Omega_q$  approximately. See Fig. 1 for an illustration. The reaction probability  $\beta(\mathbf{R})$  is approximated by  $\hat{\beta}(\xi(\mathbf{R}))$ , where  $\hat{\beta}(\xi)$  is to be determined.

The choice of reaction coordinate is discussed first. Then a two-point boundary value problem is obtained for  $\hat{\beta}(\xi)$  by minimizing a functional. Finally, a Monte Carlo method is described for computing a density function  $\rho(\xi)$ , which is required to compute  $\hat{\beta}(\xi)$ .

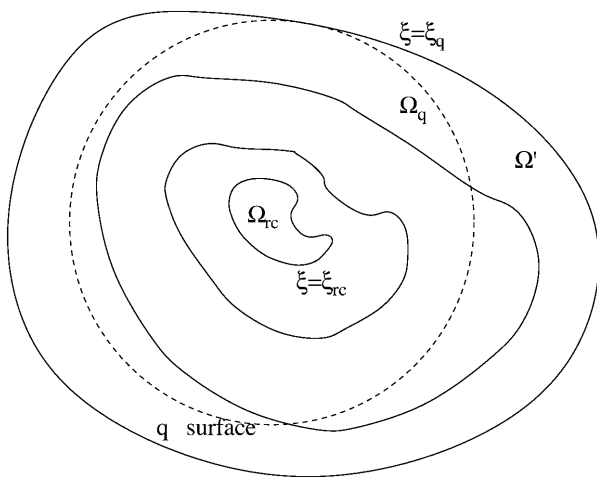


FIGURE 1 Reaction coordinate.

A single reaction coordinate  $\xi = \xi(\mathbf{R})$  is theoretically sufficient for representing  $\beta(\mathbf{R})$ . The closer the constant  $\xi$  hyper-surfaces are to the constant  $\beta$  hyper-surfaces, the better is the choice of reaction coordinate. The reaction coordinate should attempt to measure closeness to reaction in the same way that  $\beta$  does, particularly near the binding site, which is most important for sampling. That is also why  $\xi(\mathbf{R}) = \xi_{rc}$  is required to be the exact boundary of  $\Omega_{rc}$ .

As stated earlier, the reaction condition in the test problems is that  $m$  out of  $n$  distances be less than the reaction distance  $\xi_{rc}$ . Hence, the reaction coordinate  $\xi(\mathbf{R})$  is defined as the  $m^{\text{th}}$  smallest of  $n$  distances.

The value  $\xi_q$  is defined as the maximum of  $\xi(\mathbf{R})$  on the  $q$ -surface. The  $\xi(\mathbf{R}) = \xi_q$  surface is not the same as the  $q$ -surface, but is close to it. It is enough that  $\hat{\beta}(\xi(\mathbf{R}))$  satisfies the boundary condition on the  $q$ -surface only approximately, because this does not affect  $\hat{\beta}(\xi(\mathbf{R}))$  much for the more interesting configurations with small reaction coordinates.

To determine  $\hat{\beta}(\xi)$ , we use the variational formulation of Eq. 5, namely, that  $\beta(\mathbf{R})$  minimize the functional

$$I_0[\gamma] = \int_{\Omega} d\mathbf{R} e^{-U(\mathbf{R})/(k_B T)} (\nabla \gamma(\mathbf{R}))^T D(\mathbf{R}) \nabla \gamma(\mathbf{R})$$

for all functions  $\gamma(\mathbf{R})$  satisfying the same boundary condition as  $\beta(\mathbf{R})$ , which is 0 at the  $q$ -surface  $\partial\Omega_q$ , and 1 at the reaction site  $\partial\Omega_{rc}$ . We seek a function  $\hat{\beta}(\xi)$  that minimizes the much more restricted and slightly different functional

$$I[\hat{\gamma}] = \int_{\Omega'} d\mathbf{R} e^{-U(\mathbf{R})/(k_B T)} (\nabla_{\mathbf{R}} \hat{\gamma}(\xi(\mathbf{R})))^T D(\mathbf{R}) \nabla_{\mathbf{R}} \hat{\gamma}(\xi(\mathbf{R})), \quad (13)$$

subject to the boundary condition  $\hat{\gamma}(\xi_{rc}) = 1$  and  $\hat{\gamma}(\xi_q) = 0$  where  $\Omega'$  is the set of configurations with  $\xi_{rc} \leq \xi(\mathbf{R}) \leq \xi_q$ . Noting that  $\nabla_{\mathbf{R}} \hat{\gamma}(\xi(\mathbf{R})) = (d\hat{\gamma}/d\xi) \nabla \xi(\mathbf{R})$  and inserting  $\int_{\xi_{rc}}^{\xi_q} d\xi' \delta(\xi(\mathbf{R}) - \xi') = 1$  into the right-hand side of Eq. 13, where  $\delta(x)$  is the one-dimensional Dirac delta function, we obtain

$$\begin{aligned} I[\hat{\gamma}] &= \int_{\Omega'} d\mathbf{R} \int_{\xi_{rc}}^{\xi_q} d\xi' \delta(\xi(\mathbf{R}) - \xi') e^{-U(\mathbf{R})/(k_B T)} \\ &\quad \times (\nabla_{\mathbf{R}} \hat{\gamma}(\xi(\mathbf{R})))^T D(\mathbf{R}) \nabla_{\mathbf{R}} \hat{\gamma}(\xi(\mathbf{R})) \\ &= \int_{\xi_{rc}}^{\xi_q} d\xi' \rho(\xi') \left( \frac{d\hat{\gamma}}{d\xi'} \right)^2, \end{aligned} \quad (14)$$

where the density function is

$$\rho(\xi') = \int_{\Omega'} d\mathbf{R} \delta(\xi(\mathbf{R}) - \xi') e^{-U(\mathbf{R})/(k_B T)} (\nabla \xi(\mathbf{R}))^T D(\mathbf{R}) \nabla \xi(\mathbf{R}). \quad (15)$$

To minimize the functional in Eq. 14,  $\hat{\beta}(\xi)$  must satisfy  $(d/d\xi)\rho(\xi)(d/d\xi)\hat{\beta}(\xi) = 0$ , with boundary condition  $\hat{\beta}(\xi_{rc}) = 1$ ,  $\hat{\beta}(\xi_q) = 0$ . The solution is

$$\hat{\beta}(\xi) = \frac{\int_{\xi}^{\xi_q} \frac{d\xi'}{\rho(\xi')}}{\int_{\xi_{rc}}^{\xi_q} \frac{d\xi'}{\rho(\xi')}}. \quad (16)$$

To compute the density function  $\rho(\xi')$  defined in Eq. 15 numerically, let  $\xi_{rc} = \xi_0 < \dots < \xi_n = \xi_q$  be a partition of the reaction coordinate and use the piecewise constant approximation

$$\begin{aligned} \rho(\xi) &\approx \frac{1}{\xi_{i+1} - \xi_i} \int_{\xi_i}^{\xi_{i+1}} d\xi' \rho(\xi') \\ &= \frac{1}{\xi_{i+1} - \xi_i} \int_{\xi_i \leq \xi(\mathbf{R}) \leq \xi_{i+1}} d\mathbf{R} e^{-U(\mathbf{R})/(k_B T)} (\nabla \xi(\mathbf{R}))^T D(\mathbf{R}) \nabla \xi(\mathbf{R}) \end{aligned} \quad (17)$$

for  $\xi_i \leq \xi \leq \xi_{i+1}$ . The above integral is computed using Monte Carlo trials. Note that a uniform scaling for  $\rho(\xi)$  does not affect the solution  $\hat{\beta}(\xi)$ ; only the relative magnitude for different  $\xi$  matters. Uniformly distributed random configurations  $\mathbf{R}$  are generated, and for every such configuration, the reaction coordinate  $\xi(\mathbf{R})$  and the integrand in Eq. 17 are evaluated. A sum of integrand values is accumulated for each range of reaction coordinates. A rough piecewise constant function  $\rho(\xi)$  is obtained through millions of such Monte Carlo trials and is stored as a set of  $(\ln \xi, \ln \rho(\xi))$  pairs.

A single Monte Carlo simulation usually does not give enough samples in the region with small  $\xi$ -values because of its small volume. Therefore, in practice, several Monte Carlo simulations with nested ranges are performed. As shown in the figures for the test problems, each Monte Carlo simulation gives a set of  $(\ln \xi, \ln \rho(\xi))$  pairs that is “good” in a certain range of  $\xi$ . The sets of  $(\ln \xi, \ln \rho(\xi))$  pairs from different Monte Carlo simulations are aligned to each other to form a merged density that is defined in the full range  $[\xi_{rc}, \xi_q]$ , which is further smoothed to obtain a smoothed density.

The smoothing for  $(\ln \xi, \ln \rho(\xi))$  is done by least-squares fitting  $\ln \rho(\xi)$  to piecewise linear functions of  $\ln \xi$  to obtain a set of  $(\ln \xi, \ln \rho(\xi), d \ln \rho(\xi)/d \ln \xi)$  triples, which are then used with Hermite interpolation to obtain a smoothed function. The reason to use linear functions to fit  $(\ln \xi, \ln \rho(\xi))$  is that there is an approximate power relation between  $\xi$  and  $\rho(\xi)$ , i.e.,  $\rho(\xi) \propto \xi^\alpha$ , for small and for large  $\xi$ .

Alignment of two sets of  $(\ln \xi, \ln \rho(\xi))$  pairs is done by computing an offset between the two sets of data. An offset is computed as the following: select a range of  $\xi$  such that both sets are “good” in this range. Smooth both sets in the range. The average difference between the smoothed data in this range is then used as the offset.

## RATE CONSTANTS FOR SOD AND NC6.8

### Implementation details and cost measurement

The simulations reported here are performed by a C program written by the first author, which uses methods from UHBD. Bonded forces are implemented as constraints and excluded

volume forces as hard sphere bumping. The hard spheres representing the enzyme have radii equal to their van der Waals radii. When an integration step results in penetration, one simply retries the step with another random vector. Intermolecular electrostatic forces use a test charge approximation, where the force on the subunit is the product of its charge and the gradient of an electrostatic potential for the enzyme only. The potential for the enzyme’s point charges in surrounding ionized solvent is modeled by the Poisson-Boltzmann equation and precomputed by UHBD on a three-dimensional grid.

The computational cost is measured by  $t_{\text{method}}$ , the CPU hours needed to make the relative error  $\Delta \bar{\beta}(b)/\bar{\beta}(b) = 5\%$ , where  $\Delta \bar{\beta}(b)$  is the 95% confidence interval of the estimate  $\bar{\beta}(b)$ . Let  $N'_{\text{trials}}$  be the required number of trajectories to make  $\Delta \bar{\beta}(b)/\bar{\beta}(b) = 5\%$ ,  $t_{\text{step}}$  be the CPU seconds per integration step of the SDE,  $\bar{n}_{\text{steps}}$  be the average number of integration steps per trajectory,  $\bar{\beta}(b)$  be our best estimate of the average reaction probability on the  $b$ -surface (obtained as a weighted average of the available estimates with weights proportional to the reciprocals of the variances), and  $\text{Var}B$  be the estimated variance of the exit value  $B = \exp u(\tau_\Omega) f(\mathbf{R}(\tau_\Omega))$  whose expectation  $EB = \bar{\beta}(b)$ . Then

$$0.05 \times \bar{\beta}(b) = \Delta \bar{\beta}(b) = 1.96 \times \sqrt{\text{Var}B/N'_{\text{trials}}},$$

so

$$N'_{\text{trials}} = (1.96/0.05)^2 \times \text{Var}B/\bar{\beta}(b)^2$$

(for standard BD,  $\bar{\beta}(b) - \bar{\beta}(b)^2$  is used instead of  $\text{Var}B$ ), and

$$\begin{aligned} t_{\text{method}} &= t_{\text{step}} \times \bar{n}_{\text{steps}} \times N'_{\text{trials}}/3600 = 0.4268 \times t_{\text{step}} \\ &\quad \times \bar{n}_{\text{steps}} \times \frac{\text{Var}B}{\bar{\beta}(b)^2}. \end{aligned} \quad (18)$$

To get  $t_{\text{method}}$  for a 10% relative error, one simply divides it by 4. All tests in this and the next section were run on Linux clusters of dual-processor Pentium III 1-GHz machines.

The relative error for  $\bar{\beta}(b)$  is a good measure of the relative error of the rate constant  $k$ , because the second term on the right-hand side of Eq. 4 is usually much smaller than the first.

### SOD tests

The enzyme CuZn SOD converts toxic  $\text{O}_2^-$  ions to oxygen and hydrogen peroxide. SOD is extremely reactive with a rate constant close to that obtained in the diffusion limit. SOD has been extensively studied by BD simulations (e.g., Sines et al., 1990; Getzoff et al., 1992). The experimental rate constant of SOD of bovine *Bos taurus* and shark *Prionace glauca* is reported to be  $3.92 \times 10^9 \text{ M}^{-1} \text{ s}^{-1}$  (Polticelli et al., 1994) compared to a calculated value of  $5 \times 10^9 \text{ M}^{-1} \text{ s}^{-1}$  (Rojnuckarin et al., 2000). Two types of SOD are examined here, *H. sapiens* SOD 1spd and *E. coli* SOD 1eso. The

computed rate constant of SOD 1spd is  $\sim 1.4 \times 10^9 \text{ M}^{-1} \text{ s}^{-1}$ . The computed rate constant of SOD 1eso is one order of magnitude less.

Coordinate data for SOD are taken from the Protein Data Bank (<http://www.rcsb.org/pdb>) and the partial charge and van der Waals parameter tables are taken from the standard CHARMM force field (Brooks et al., 1983) incorporated into UHBD. The substrate  $\text{O}_2^-$  is modeled by a subunit with a hard-sphere radius  $1.5 \text{ \AA}$  and a charge of  $-e$ . Reaction is said to happen when  $\text{O}_2^-$  is within  $7 \text{ \AA}$  of the copper atom of SOD. The diffusion tensor  $D$  is diagonal as in Eq. 2 where  $T = 300 \text{ K}$ , water viscosity  $\eta = 0.89 \text{ g m}^{-1} \text{ s}^{-1}$ , the  $\text{O}_2^-$  hydrodynamic radius  $a_1 = 2.05 \text{ \AA}$ , and that of SOD is  $a_{\text{mol}} = 25 \text{ \AA}$ . The simulations use  $b = 80 \text{ \AA}$  and  $q = 400 \text{ \AA}$ .

Fig. 2 shows the Monte Carlo simulation results for  $\rho(\xi)$  for SOD 1spd. In the figure, each of the three original densities is valid for a certain range of  $\xi$ -values. They are aligned to each other by shifting vertically to form a merged density that is valid in the full range. This merged density is then smoothed to form a smoothed density in the full range by the process given in the preceding section.

The  $\beta$  function is computed from the smoothed  $\rho(\xi)$  with Eq. 16. The bias potential  $U_b(\xi) = -2k_B T \ln \hat{\beta}(\xi)$  as a function of reaction coordinate  $\xi$  is shown in Fig. 3. The turning point of the curve is at  $\sim \xi = 20 \text{ \AA}$  where  $U_b(\xi) = 5k_B T$ . The magnitude of the bias potential indicates the strength of the free energy barrier needed to be overcome for the substrate to get to the reaction site of the enzyme. Compared to results for the tests that follow, the value  $5k_B T$  is relatively low. The corresponding reaction probability and rate constant are also large compared to those in other tests. Near  $\xi = \xi_q$ , the bias potential prevents the particle from reaching the  $\xi = \xi_q$  surface.

Table 1 shows the rate constants of SOD 1spd and computing costs for biased BD and standard BD. In addition to the quantities defined earlier, the tabulated value  $N_{\text{trials}}$  is the actual number of trajectories,  $EB_{\text{est}}$  is the estimate

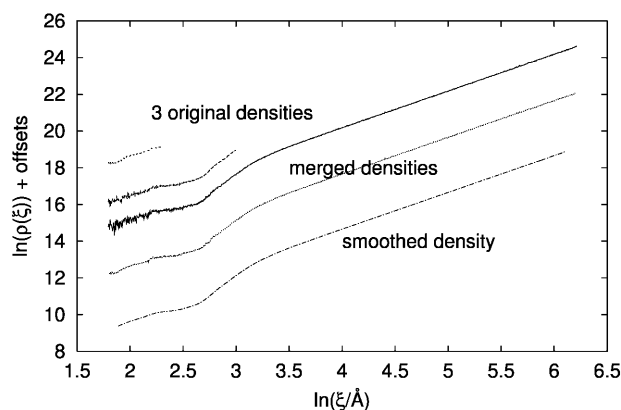


FIGURE 2 Monte Carlo simulations for SOD 1spd give three original densities for differing  $\xi$ -ranges. These are merged and smoothed to determine  $\rho(\xi)$ .

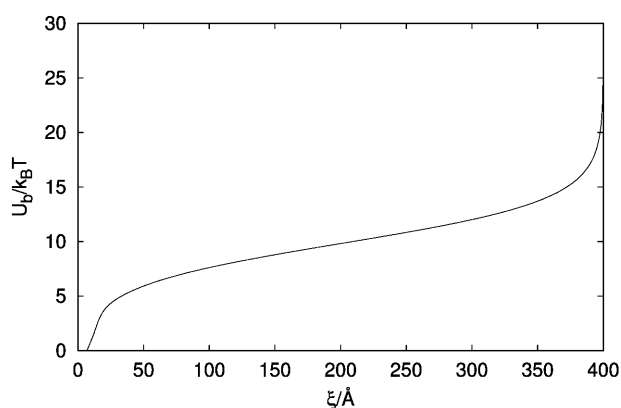


FIGURE 3 The bias potential for SPD 1spd with escape distance  $q = 400 \text{ \AA}$ .

produced for  $\hat{\beta}(b)$ , and  $CPU$  is the actual CPU time. The CPU cost per integration step of biased BD is  $2.7\times$  that of the standard BD. The number of integration steps per trajectory of biased BD is 43% of that of the standard BD due to the weight control algorithm. The variance of biased BD is only 5% of that of the standard BD. The overall speedup of biased BD relative to standard BD is 17. It is interesting to compare this speedup with the sevenfold speedup obtained for biased BD without weight control several years ago (Zou et al., 2000).

*E. coli* SOD 1eso has a lower attractive force to steer the  $\text{O}_2^-$  ions to its binding site, which results in a higher energy barrier for association and a lower rate constant. Its bias potential has a graph that is qualitatively similar to that in Fig. 3. The turning point of the curve is at  $\sim \xi = 20 \text{ \AA}$  where  $U_b(\xi) = 7k_B T$ . This is larger than that of SOD 1spd and the rate constant is smaller as a result. The effect of using partly numerical instead of heuristic estimates of the reaction probability is shown for SOD 1eso in Table 2. The heuristic choice is  $\hat{\beta}(\xi) = (1/\xi - 1/\xi_q)/(1/\xi_{rc} - 1/\xi_q)$  where  $\xi_{rc} = 7 \text{ \AA}$  and  $\xi_q = 405.26545 \text{ \AA}$  (Zou, 2002). The best result in this table is  $35\times$  that of unbiased BD given in Table 1.

## NC6.8 test

This section considers the antibody-antigen binding reaction between the antisweetener antibody NC6.8 (Protein Data Bank entry 2cgr) and the sweet-tasting ligand *n*-(p-cyanophenyl)-*n'*-(diphenylmethyl)-guanidinium acetic acid, formula  $\text{C}_{23}\text{H}_{20}\text{N}_4\text{O}_2$ ). The antibody is the enzyme and the ligand is the substrate. The NC6.8 binding reaction was previously studied for the WEBD (Rojnuckarin et al., 2000). Here we follow the same setup and test for biased BD.

The NC6.8 antibody and the sweet-tasting ligand structure files are from the Protein Data Bank. In the simulation, only the Fv fragment of NC6.8 is used. The setup is similar to that of SOD except the following: the sweet-tasting ligand is modeled as a two-subunit dumbbell with a distance

**TABLE 1 Results for unbiased and full-featured biased Brownian dynamics**

System	Unbiased BD			Biased BD	
	SOD 1spd	SOD 1eso	NC6.8	SOD 1spd	NC6.8
$t_{\text{step}}(\text{s})$	$2.55 \times 10^{-6}$	$2.58 \times 10^{-6}$	$7.86 \times 10^{-6}$	$6.83 \times 10^{-6}$	$1.65 \times 10^{-5}$
$\bar{n}_{\text{steps}}$	$1.94 \times 10^5$	$1.96 \times 10^5$	$1.29 \times 10^5$	$8.32 \times 10^4$	$7.58 \times 10^4$
$N_{\text{trials}}$	200,000	200,000	160,000	200,000	80,000
CPU(h)	27.5	28.2	45.1	31.6	27.8
$EB_{\text{est}}$	$1.49 \times 10^{-2}$	$1.73 \times 10^{-3}$	$6.50 \times 10^{-4}$	$1.45 \times 10^{-2}$	$6.82 \times 10^{-4}$
$\text{Var}B$	$1.47 \times 10^{-2}$	$1.73 \times 10^{-3}$	$6.50 \times 10^{-4}$	$7.21 \times 10^{-4}$	$1.40 \times 10^{-5}$
$k(\text{M}^{-1}\text{s}^{-1})$	$1.46 \times 10^9$	$1.71 \times 10^8$	$5.85 \times 10^7$	$1.42 \times 10^9$	$6.13 \times 10^7$
$\Delta k(\text{M}^{-1}\text{s}^{-1})$	$5.19 \times 10^7$	$1.79 \times 10^7$	$1.12 \times 10^7$	$1.15 \times 10^7$	$2.33 \times 10^6$
$t_{\text{method}}(\text{h})$	14.4	123.3	635.5	0.8	16.1

constraint of 4.374 Å between subunit centers. The negative subunit, centered at the carbonyl carbon (C19), has a radius of 1.5 Å and a charge of  $-e$ . The positive subunit, centered at the most cationic guanidinium nitrogen (N16), has a radius of 2.0 Å and a charge of  $+e$ . The reaction condition is that the positive subunit is within 5.0 Å of Glu H:162 OE2 (H:50), and the negative subunit is within 5.0 Å of Arg H:169 NH1 (H:57). The diffusion tensor  $D$  is a Rotne-Prager tensor as in Eqs. 2 and 3 where the hydrodynamic radius of the positive subunit  $a_1 = 2.0$  Å, that of the negative subunit  $a_2 = 1.5$  Å, and that of the enzyme  $a_{\text{mol}} = 25$  Å. The values  $b = 80$  Å and  $q = 300$  Å are used.

The bias potential  $U_b(\xi)$  for NC6.8 is again qualitatively similar to that in Fig. 3. The turning point of the curve is at  $\sim \xi = 20$  Å where  $U_b(\xi) = 8k_B T$ , which is larger than for both SOD tests and results in a smaller rate constant. Table 1 shows a 39× speedup of biased BD over standard BD for the NC6.8 test. A speedup of 8 is obtained from the WEBD/NAM method of Rojnuckarin et al. (2000).

## ARTIFICIAL TEST PROBLEM

### Problem description

The model problem described here is proposed in a 1992 article by Northrup and Erickson. It is later used by Huber and Kim in 1996 to show the speedup of the WEBD algo-

**TABLE 2 Results for SOD 1eso with heuristic and partly numerical estimates of reaction probability**

wt ctl?	Heuristic		Partly numerical	
	No	Yes	No	Yes
$t_{\text{step}}(\text{s})$	$5.50 \times 10^{-6}$	$6.91 \times 10^{-6}$	$5.59 \times 10^{-6}$	$6.83 \times 10^{-6}$
$\bar{n}_{\text{steps}}$	$6.14 \times 10^5$	$1.01 \times 10^5$	$1.57 \times 10^5$	$8.78 \times 10^4$
$N_{\text{trials}}$	200,000	200,000	200,000	200,000
CPU(h)	187.8	38.9	48.7	33.3
$EB$	$1.74 \times 10^{-3}$	$1.74 \times 10^{-3}$	$1.74 \times 10^{-3}$	$1.75 \times 10^{-3}$
$\text{Var}B$	$6.86 \times 10^{-5}$	$7.20 \times 10^{-5}$	$1.46 \times 10^{-4}$	$4.16 \times 10^{-5}$
$k(\text{M}^{-1}\text{s}^{-1})$	$1.71 \times 10^8$	$1.71 \times 10^8$	$1.72 \times 10^8$	$1.72 \times 10^8$
$\Delta k(\text{M}^{-1}\text{s}^{-1})$	$3.57 \times 10^6$	$3.66 \times 10^6$	$5.21 \times 10^6$	$2.78 \times 10^6$
$t_{\text{method}}(\text{h})$	32.9	7.1	18.0	3.5

rithm. Although this problem is outside the class of problems defined previously, the modification is straightforward.

In solvent are two types of proteins, each modeled as a sphere of radius 18 Å. For the purpose of defining a reaction condition, imagine that  $17 \text{ Å} \times 17 \text{ Å}$  squares are attached at their centers tangentially to each sphere as shown in Fig. 4. The set of reacted configurations  $\Omega_{\text{rc}}$  consists of those configurations with a position and orientation such that at least three of the four vertex pairs,  $A_1A_2$ ,  $B_1B_2$ ,  $C_1C_2$ , and  $D_1D_2$ , between the two squares, are within 2 Å.

For  $i = 1, 2$ , let  $\mathbf{r}_i$  be the center of protein  $i$ ,  $Q_i$  be an orthogonal matrix describing the orientation of protein  $i$ , and  $a_i = 18$  Å be the radius of protein  $i$ . Although the orthogonal matrix  $Q_i$  contains nine entries, it has only 3 degrees of freedom because of the constraints  $Q_i^T Q_i = I$ . Any point fixed in the body frame of protein  $i$  can be converted to lab frame coordinates with  $\mathbf{r}_{\text{lab}} = Q_i \mathbf{r}_{\text{body}} + \mathbf{r}_i$ .

The tangent space for the three-dimensional manifold  $Q_i^T Q_i = I$  at a particular point  $Q_i$  has as a basis  $Q_{x,i} = Q_i \text{skew}((1, 0, 0)^T)$ ,  $Q_{y,i} = Q_i \text{skew}((0, 1, 0)^T)$ , and  $Q_{z,i} = Q_i \text{skew}((0, 0, 1)^T)$ , where skew is the mapping

$$\text{skew}(\mathbf{v}) = \begin{pmatrix} 0 & v_3 & -v_2 \\ -v_3 & 0 & v_1 \\ v_2 & -v_1 & 0 \end{pmatrix}$$

from a vector to a skew matrix. The basis is orthogonal with respect to the double-dot inner product ( $A:C = \text{tr}(A^T C)$ ) for matrices. Hence, the motion of two proteins is described by their translational and rotational diffusion in the liquid,

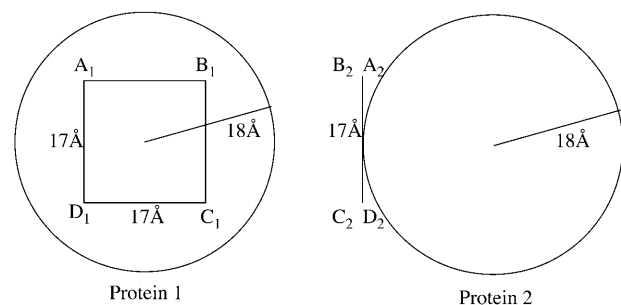


FIGURE 4 Model protein diagram.

$$\begin{aligned} d\mathbf{r}_i &= \sqrt{2D_t}d\mathbf{W}_i, \\ dQ_i &= Q_i \text{skew}(\sqrt{2D_r}d\mathbf{W}_i^T), \end{aligned} \quad (19)$$

for  $i = 1, 2$ , where  $D_t$  and  $D_r$  are translational and rotational diffusion coefficients given by the Stokes-Einstein relations  $D_t = k_B T / 6\pi\eta a_i = 1.36 \times 10^{-10} \text{ m}^2 \text{ s}^{-1}$ ;  $D_r = k_B T / 8\pi\eta a_i^3 = 3.16 \times 10^7 \text{ s}^{-1}$  for  $T = 298 \text{ K}$ ;  $\eta = 0.89 \text{ g m}^{-1} \text{ s}^{-1}$ ; and where  $\mathbf{W}_1$ ,  $\mathbf{W}_2$ ,  $\mathbf{W}_1^T$ , and  $\mathbf{W}_2^T$  are independent canonical three-dimensional Wiener processes. There is no interaction between the proteins except hard sphere bumping.

Consider the relative position  $\mathbf{r}_c = \mathbf{r}_1 - \mathbf{r}_2$ . We have

$$d\mathbf{r}_c = \sqrt{2 \times 2D_t}d\tilde{\mathbf{W}} \quad (20)$$

where  $\tilde{\mathbf{W}} = (\mathbf{W}_1 - \mathbf{W}_2)/\sqrt{2}$  is a canonical Wiener process. The  $b$ -surface and  $q$ -surface are the set of configurations with  $|\mathbf{r}_c| = b$  and  $|\mathbf{r}_c| = q$ , respectively. The distribution of starting points on the  $b$ -surface is uniform for the center  $\mathbf{r}_c$  and uniform with respect to the orientation of protein  $l$ .

We compute the rate constant  $k$  with Eq. 4, where  $U_{\text{ext}}(r) = 0$  and  $d(r) = 2D_t$ , although rigorous justification is not given here.

### Numerical integration and bias force

For the  $\mathbf{r}_c$  coordinates, there is no difference from previous sections for the numerical integration, bias force evaluation, and weight computation. However, these are all slightly different for the  $Q_i$  coordinates.

Directly applying a numerical integration method to Eq. 19 for  $Q_i$  usually does not preserve the orthogonality constraints of  $Q_i$ . To preserve the constraints, we update  $Q_i$  by multiplying it by orthogonal matrices. Let

$$R_x(\phi) = \exp(-\text{skew}((\phi, 0, 0)^T)) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & -\sin \phi \\ 0 & \sin \phi & \cos \phi \end{pmatrix},$$

in which  $R_x(\phi)$  is the matrix that rotates an angle  $\phi$  about the  $x$ -axis. Similarly define  $R_y$  and  $R_z$ . For a finite time step  $\Delta t$ , let  $\sqrt{\Delta t}\mathbf{Z}_i$  be the Wiener increment  $\mathbf{W}_i^T(t + \Delta t) - \mathbf{W}_i^T(t)$ , and let

$$\Delta\mathbf{X}_i = (\Delta X_{x,i}, \Delta X_{y,i}, \Delta X_{z,i})^T = \sqrt{2\Delta t}D_i\mathbf{Z}_i.$$

Then to integrate  $Q_i$  numerically, use

$$\begin{aligned} Q_i(t + \Delta t) &\approx Q_i(t)(I + \text{skew}(\Delta\mathbf{X}_i)) \\ &\approx Q_i(t)R_x(\Delta X_{x,i})^T R_y(\Delta X_{y,i})^T R_z(\Delta X_{z,i})^T. \end{aligned}$$

Let

$$D = \begin{pmatrix} 2D_t I & 0 & 0 \\ 0 & D_r I & 0 \\ 0 & 0 & D_r I \end{pmatrix}, \quad \Delta\mathbf{X} = \begin{pmatrix} \Delta\mathbf{r}_c \\ \Delta\mathbf{X}_1 \\ \Delta\mathbf{X}_2 \end{pmatrix}, \quad \mathbf{Z} = \begin{pmatrix} \tilde{\mathbf{Z}} \\ \mathbf{Z}_1 \\ \mathbf{Z}_2 \end{pmatrix},$$

where  $\Delta\mathbf{r}_c$  is the increment of  $\mathbf{r}_c$  during the time period  $[t, t + \Delta t]$  and  $\sqrt{\Delta t}\tilde{\mathbf{Z}}$  is the increment of  $\tilde{\mathbf{W}}(\tau)$ . The unbiased BD numerical scheme is thus

$$\Delta\mathbf{X} = 0\Delta t + \sqrt{2}D_{1/2}\sqrt{\Delta t}\mathbf{Z}$$

where  $D_{1/2}D_{1/2}^T = D$ .

Similar to Eqs. 7 and 8 for biased BD, here we add a bias vector to the stochastic increment and associate a weight  $\exp(u)$  with the system. Let  $\mathbf{F}_b = (\mathbf{F}_{b,c}^T \mathbf{F}_{b,1}^T \mathbf{F}_{b,2}^T)^T$  where  $\mathbf{F}_{b,c}$ ,  $\mathbf{F}_{b,1}$ , and  $\mathbf{F}_{b,2}$  are the bias forces for  $\mathbf{r}_c$ ,  $Q_1$ , and  $Q_2$ , respectively. The unbiased numerical scheme changes to the biased scheme

$$\begin{pmatrix} \Delta\mathbf{X} \\ \Delta u \end{pmatrix} = \begin{pmatrix} D \frac{\mathbf{F}_b}{k_B T} \\ -\frac{1}{4} \frac{\mathbf{F}_b^T}{k_B T} D \frac{\mathbf{F}_b}{k_B T} \end{pmatrix} \Delta t + \begin{pmatrix} \sqrt{2}D_{1/2} \\ -\frac{\sqrt{2}}{2} \frac{\mathbf{F}_b^T}{k_B T} D_{1/2} \end{pmatrix} \sqrt{\Delta t}\mathbf{Z}.$$

The bias force  $\mathbf{F}_b/(k_B T)$  is computed numerically from Eq. 12. For example, to compute the derivative of the approximate reaction probability  $\beta$  in the direction of  $Q_{x,i}$ , use  $(\beta' - \beta)/\Delta\phi$  where  $\beta$  is the approximate solution for the current configuration and  $\beta'$  is the approximate solution for the configuration obtained by replacing  $Q_i$  by  $Q_i R_x(\Delta\phi)^T$  for some small angle  $\Delta\phi$ .

### Reaction coordinates and Monte Carlo calculation

Uniformly distributed random configurations are generated by randomly placing protein  $l$  at a distance  $r$  with probability density  $\propto r^2$ , and rotating protein  $l$  with Euler angles  $\phi$ ,  $\theta$ ,  $\psi$ , where  $\phi$  is the angle of self-rotation along the  $z$ -axis,  $\theta$  is the angle of nutation along the  $x$ -axis, and  $\psi$  is the angle of precession along the  $z$ -axis. The positive  $z$ -axis goes through the center of the square on the protein. Angles  $\phi$  and  $\psi$  are uniformly distributed in  $[0, 2\pi]$  and  $\cos \theta$  is uniformly distributed in  $[-1, 1]$ .

Uniformly distributed random configurations will not give enough samples for small  $\xi$ , for example, for  $\xi < 10 \text{ \AA}$ . Importance sampling with restricted distance and restricted nutation are used. The distance is restricted by a small maximum distance  $r_{\text{max}}$ . The cosine of the nutation angle  $\theta$  is restricted to a uniform distribution in  $[(\cos \theta)_{\text{min}}, 1]$ , for example,  $(\cos \theta)_{\text{min}} = 0.8$  or  $(\cos \theta)_{\text{min}} = 0.95$ . Note that restricted Monte Carlo simulation gives correct relative  $\rho(\xi)$  only for those  $\xi$  that are small enough that they cannot be generated with nutation angle satisfying  $\cos \theta < (\cos \theta)_{\text{min}}$  or distance  $r > r_{\text{max}}$ . The restricted and unrestricted Monte Carlo simulation results are aligned at some common range of values of  $\xi$  which are valid for and have enough samples from each simulation.

In Fig. 5, each original density is valid for a certain region. These are aligned by vertical shifts and then smoothed to form a density for the full range.



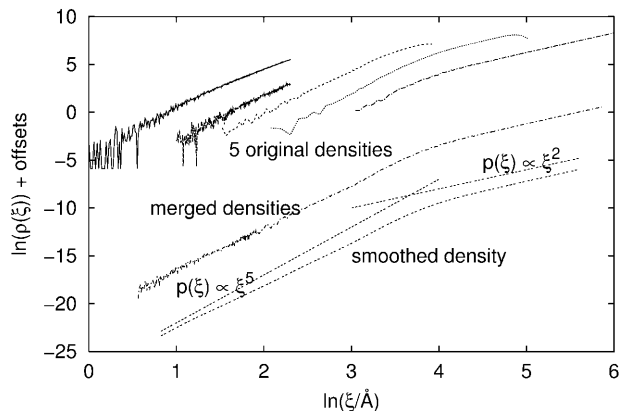


FIGURE 5 Monte Carlo simulations for the two-spheres problem give five original densities for differing ranges of distance and nutation angle. These are merged and smoothed to determine  $\rho(\xi)$ . This curve is compared with straight lines corresponding to  $\rho(\xi) \propto \xi^2$  and  $\rho(\xi) \propto \xi^5$ .

The bias potential  $U_b(\xi)$  is shown in Fig. 6. When  $\xi$  is small, the bias force needs to overcome the strong entropy barrier caused by both translational and orientational restrictions. The relative movement of the spheres is like a particle moving in six-dimensional space with the reaction site being a radii  $2 \text{ \AA}$  spherical surface in six-dimensional space. When  $\xi$  gets larger, the restriction from orientation vanishes, and the movement is similar to a three-dimensional free diffusion.

### Estimated cost of standard Brownian dynamics

We did not perform extensive numerical tests with the standard BD algorithm because of the time-consuming nature of these tests and because a good theoretical estimate is possible.

Recall Eq. 18 for the cost of an algorithm: it needs CPU seconds per integration step  $t_{\text{step}}$ , the average number of integration steps per trajectory  $\bar{n}_{\text{steps}}$ ,  $\text{Var}B$ , and  $\beta(b)$ . The

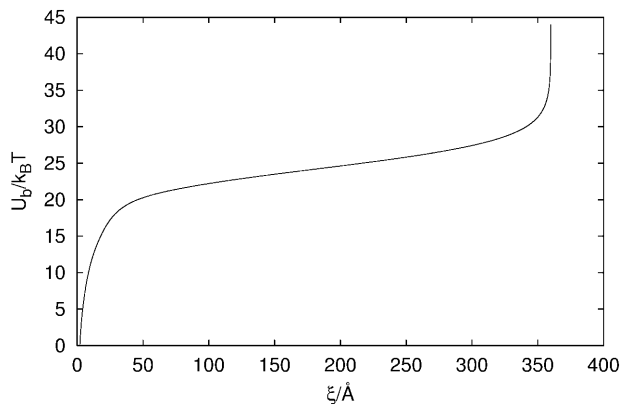


FIGURE 6 The bias potential of the two-spheres problem with escape distance  $q = 360 \text{ \AA}$ .

value of  $t_{\text{step}}$  is computed to be  $4.49 \times 10^{-6} \text{ s}$  for this test problem from computer timings. The value of  $\beta(b)$  is obtained from the biased BD result. Because the exit values of standard BD are either 1 or 0 depending on whether or not they react,  $EB^2 = EB$  and  $\text{Var}B = EB^2 - (EB)^2 = EB - (EB)^2 \approx EB$ , which is obtained from the biased BD result for  $\beta(b)$ . Similar to the method used in Huber and Kim (1996), the average number of integration steps of one trajectory  $\bar{n}_{\text{steps}}$  is computed by

$$\bar{n}_{\text{steps}} = T(b)/\Delta t,$$

where  $\Delta t = 2.38 \text{ ps}$  is the maximum time step in the numerical integration scheme for Eq. 19, and  $T(r)$  is the mean first passage time (MFPT) for protein 1 starting at distance  $r$  away from protein 2 with exit boundary at  $q$  and reflecting boundary at  $a = a_1 + a_2 = 36 \text{ \AA}$ , the minimum-allowed distance between two proteins. Since the probability of reaction is so low, we neglect the possibility of exit at the reaction site. Thus, the MFPT is assumed to depend only on the distance between two proteins and not on their orientations. It can be shown (see Appendix) that  $T(r)$  satisfies the equation

$$-\frac{1}{r^2} \frac{d}{dr} r^2 \times 2D_t \frac{d}{dr} T(r) = 1,$$

with boundary conditions  $(d/dr)T(r)|_{r=a} = 0$  and  $T(r)|_{r=q} = 0$ . The solution for the MFPT is

$$T(b) = \frac{q^2 - b^2}{12D_t} + \frac{a^3}{6D_t} \left( \frac{1}{q} - \frac{1}{b} \right).$$

### Results for the model problem

The effect of using partly numerical instead of heuristic estimates of the reaction probability is shown in Table 3. For a heuristic density function we use Eq. 16 where  $d \ln \rho(\xi)/d \ln \xi = a(\xi)$  and  $a(\xi)$  is a step function that takes the value 5 for  $2 < \xi < 30$ , the value 4 for  $30 < \xi < 40$ , the value 3 for  $40 < \xi < 50$ , and the value 2 for  $50 < \xi < 600$ . Comparison to unbiased BD is given in Table 4. Column 5 of each table is

TABLE 3 Results for model protein with heuristic and partly numerical estimates of reaction probability

wt ctl?	Heuristic		Partly numerical	
	No	Yes	No	Yes
$t_{\text{step}}(\text{s})$	$7.69 \times 10^{-6}$	$8.85 \times 10^{-6}$	$7.68 \times 10^{-6}$	$8.83 \times 10^{-6}$
$\bar{n}_{\text{steps}}$	$1.94 \times 10^5$	$1.48 \times 10^5$	$1.37 \times 10^5$	$1.33 \times 10^5$
$N_{\text{trials}}$	200,000	200,000	200,000	200,000
CPU(h)	82.8	73.0	58.4	65.3
$EB$	$4.95 \times 10^{-6}$	$8.66 \times 10^{-6}$	$5.48 \times 10^{-6}$	$8.52 \times 10^{-6}$
$\text{Var}B$	$6.41 \times 10^{-8}$	$4.06 \times 10^{-9}$	$4.92 \times 10^{-8}$	$3.40 \times 10^{-9}$
$k(\text{M}^{-1}\text{s}^{-1})$	$1.05 \times 10^5$	$1.83 \times 10^5$	$1.16 \times 10^5$	$1.80 \times 10^5$
$\Delta k(\text{M}^{-1}\text{s}^{-1})$	$2.35 \times 10^4$	$5.91 \times 10^3$	$2.06 \times 10^4$	$5.41 \times 10^3$
$t_{\text{method}}(\text{h})$	1667.3	30.4	735.3	23.5

**TABLE 4** Model protein results for unbiased and full-featured biased Brownian dynamics

	Unbiased BD		Biased BD		
$b(\text{\AA})$	45.0	80.0	45.0	80.0	140.0
$q(\text{\AA})$	360.0	360.0	360.0	360.0	1000.0
$t_{\text{step}}(\text{s})$	$4.49 \times 10^{-6}$	$4.49 \times 10^{-6}$	$8.73 \times 10^{-6}$	$8.68 \times 10^{-6}$	$8.59 \times 10^{-6}$
$\bar{n}_{\text{steps}}$	$3.24 \times 10^{5*}$	$3.15 \times 10^{5*}$	$6.65 \times 10^4$	$1.22 \times 10^5$	$7.39 \times 10^5$
$N_{\text{trials}}$			120,000	120,000	40,000
CPU(h)			19.4	35.3	70.6
$EB_{\text{est}}$			$1.72 \times 10^{-5}$	$8.56 \times 10^{-6}$	$5.42 \times 10^{-6}$
$\text{Var}B$			$1.67 \times 10^{-8}$	$3.28 \times 10^{-9}$	$1.33 \times 10^{-9}$
$k(\text{M}^{-1}\text{s}^{-1})$			$1.82 \times 10^5$	$1.81 \times 10^5$	$1.81 \times 10^5$
$\Delta k(\text{M}^{-1}\text{s}^{-1})$			$7.74 \times 10^3$	$6.85 \times 10^3$	$1.20 \times 10^4$
$t_{\text{method}}(\text{h})$	36089.5*	70487.1*	14.0	20.2	122.5

an independent computation. The starred entries for unbiased BD denote the theoretical estimates. The per-step cost for biased BD is about twice as expensive as that of standard BD. Biased BD gains in both the average number of steps per trajectory and the variance. Overall, biased BD gives a factor of 2578 speedup with  $b = 45 \text{ \AA}$  and  $q = 360 \text{ \AA}$  and a factor of 3341 speedup with  $b = 80 \text{ \AA}$  and  $q = 360 \text{ \AA}$ .

## APPENDIX: FEYNMAN-KAC FORMULA FOR ELLIPTIC PARTIAL DIFFERENTIAL EQUATIONS

Let

$$\beta(\mathbf{R}) = E_{0,\mathbf{R}} \left( f(\mathbf{R}(\tau_\Omega)) + \int_0^{\tau_\Omega} g(\mathbf{R}(t)) dt \right), \quad (21)$$

where  $\mathbf{R}(t)$  satisfies

$$d\mathbf{R}(t) = a(\mathbf{R}(t))dt + \sqrt{2}D_{1/2}(\mathbf{R}(t))dW(t),$$

with initial condition  $\mathbf{R}(0) = \mathbf{R}$  and where  $\tau_\Omega$  is the first exit time from domain  $\Omega$ . The expectation  $\beta(\mathbf{R})$  is the solution of the elliptic boundary value problem

$$-\mathcal{L}(\mathbf{R})\beta(\mathbf{R}) = g(\mathbf{R}), \quad \mathbf{R} \in \Omega, \beta(\mathbf{R}) = f(\mathbf{R}), \quad \mathbf{R} \in \partial\Omega, \quad (22)$$

with  $\mathcal{L}$  defined by

$$\mathcal{L}(\mathbf{R})\gamma(\mathbf{R}) = (a(\mathbf{R})^T \nabla_{\mathbf{R}} + D(\mathbf{R}) : \nabla_{\mathbf{R}} \nabla_{\mathbf{R}}^T) \gamma(\mathbf{R}),$$

where  $A:C$  means  $\text{tr}(A^T C)$  for two matrices  $A$  and  $C$  of the same dimensions,  $D = D_{1/2} D_{1/2}^T$ , and  $\nabla_{\mathbf{R}} \nabla_{\mathbf{R}}^T$  operates only on  $\gamma(\mathbf{R})$ .

To apply this to Eq. 5, note that  $\nabla_{\mathbf{R}}^T D(\mathbf{R}) \nabla_{\mathbf{R}} = (\nabla_{\mathbf{R}}^T D(\mathbf{R})) \nabla_{\mathbf{R}} + D(\mathbf{R}) : \nabla_{\mathbf{R}} \nabla_{\mathbf{R}}^T$ .

To apply this to the calculation of the MFPT in last section, use Eq. 20 as the SDE, and let  $f \equiv 0$ ,  $g \equiv 1$  in Eq. 21. Then  $\beta(\mathbf{R})$  is the MFPT, and  $\beta(\mathbf{R})$  satisfies Eq. 22.

The assistance of A. Rojnuckarin and D. Livesay in the setup for SOD and NC6.8 is gratefully acknowledged.

This material is based upon work supported by the National Science Foundation under grants 9974555 and 020442. It was also supported by a Computational Science and Engineering Fellowship from the University of Illinois.

## REFERENCES

- Antosiewicz, J., M. K. Gilson, I. H. Lee, and J. A. McCammon. 1995. Acetylcholinesterase: diffusional encounter rate constants for dumbbell models of ligand. *Biophys. J.* 68:62–68.
- Brooks, B. R., R. E. Bruccoleri, B. D. Olafson, D. J. States, S. Swaminathan, and M. Karplus. 1983. CHARMM: a program for macromolecular energy, minimization, and dynamics calculations. *J. Comput. Chem.* 4:187–217.
- Brune, D., and S. Kim. 1994. Hydrodynamic steering effects in protein association. *Proc. Natl. Acad. Sci. USA.* 91:2930–2934.
- Davis, M. E., J. D. Madura, B. A. Luty, and J. A. McCammon. 1991. Electrostatics and diffusion of molecules in solution: simulations with the University of Houston Brownian dynamics program. *Comp. Phys. Comm.* 62:187–197.
- Getzoff, E. D., C. L. Fisher, H. E. Page, M. S. Viezzoli, L. Banci, and R. A. Hallewell. 1992. Faster superoxide dismutase mutants designed by enhancing electrostatic steering. *Nature.* 358:347–351.
- Guddat, L. W., L. Shanand, J. M. Anchin, D. S. Linthicum, and A. B. Edmundson. 1994. Local and transmitted conformational changes on complexation of an anti-sweetener Fab. *J. Mol. Biol.* 236:247–274.
- Hänggi, P., P. Talkner, and M. Borkovec. 1990. Reaction-rate theory: fifty years after Kramers. *Rev. Mod. Phys.* 62:251–342.
- Huber, G. A., and S. Kim. 1996. Weighted-ensemble Brownian dynamics simulations for protein association reactions. *Biophys. J.* 70:97–110.
- Kozack, R. E., M. J. D’Mello, and S. Subramaniam. 1995. Computer modeling of electrostatic steering and orientational effects in antibody-antigen association. *Biophys. J.* 68:807–814.
- Milstein, G. N. 1988. The Numerical Integration of Stochastic Differential Equations. Urals University Press, Sverdlovsk, Russia. English ed., 1995, Kluwer Academic Publishers, Dordrecht, The Netherlands.
- Northrup, S. H., S. A. Allison, and J. A. McCammon. 1984. Brownian dynamics simulation of diffusion-influenced bimolecular reactions. *J. Chem. Phys.* 80:1517–1524.
- Northrup, S. S., K. A. Thomasson, C. M. Miller, P. D. Barker, L. D. Eltis, J. G. Guillemette, S. C. Inglis, and A. G. Mark. 1993. Effects of charged amino acid mutations on the bimolecular kinetics of reduction of yeast iso-1-ferricytochrome c by bovine ferrocycytochrome b5. *Biochemistry.* 32:6613–6623.
- Polticelli, F., M. Falconi, P. O’Neill, R. Petruzelli, A. Galtieri, A. Lania, L. Calabrese, G. Rotillio, and A. Desideri. 1994. Molecular modeling and electrostatic potential calculations on chemically modified Cu, Zn superoxide dismutases from *Bos taurus* and shark *Prionace glauca*: role of Lys134 in electrostatically steering the substrate to the active site. *Arch. Biochem. Biophys.* 312:22–30.
- Rojnuckarin, A., D. R. Livesay, and S. Subramaniam. 2000. Bimolecular reaction simulation using weighted ensemble Brownian dynamics and

- the University of Houston Brownian dynamics program. *Biophys. J.* 79: 686–693.
- Sines, J. J., S. A. Allison, and J. A. McCammon. 1990. Point charge distributions and electrostatic steering in enzyme/substrate encounter: Brownian dynamics of modified copper/zinc superoxide dismutases. *Biochemistry*. 29:9403–9412.
- Wade, R. C., B. A. Luty, E. Demchuk, J. D. Madura, M. E. Davis, J. M. Briggs, and J. A. McCammon. 1994. Simulation of enzyme-substrate encounter with gated active sites. *Struct. Biol.* 1:65–69.
- Zhou, H.-X. 1990. On the calculation of diffusive reaction rates using Brownian dynamics simulations. *J. Phys. Chem.* 92:3092–3095.
- Zou, G. 2002. Robust Biased Brownian Dynamics for Rate Constant Calculation. Ph. D. thesis, University of Illinois at Urbana-Champaign. Also Department of Computer Science Report No. 2294, University of Illinois at Urbana-Champaign, August 2002. Available online at <http://www.cs.uiuc.edu/research/reports.html>.
- Zou, G., R. D. Skeel, and S. Subramaniam. 2000. Biased Brownian dynamics for rate constant calculation. *Biophys. J.* 79:638–645.