Sensitivity Analysis in the Face of Rare Events

John Strahan[1] and Todd R. Gingrich[1,2,a)]
[1)] *Department of Chemistry, Northwestern University, 2145 Sheridan Road, Evanston, Illinois 60208, USA*
[2)] *Department of Engineering Sciences and Applied Mathematics, Northwestern University, 2145 Sheridan Road, Evanston, Illinois 60208, USA*

Molecular motors and other complex nonequilibrium systems are controlled by large sets of design parameters, and optimizing those parameters requires computing sensitivities—derivatives of dynamical observables with respect to the parameters. When the system's dynamics involves rare events, both the observable and its sensitivity are difficult to estimate from direct simulation. We present a practical computational pipeline that addresses both challenges by combining importance sampling with a Markov state model (MSM). The MSM separately captures the slow, rare-event dynamics and the fast, local dynamics, and the chain rule connects those two pieces to yield an efficient sensitivity estimator. An iterative reweighting procedure based on the RiteWeight algorithm substantially reduces approximation errors from the MSM coarse-graining. We validate the approach on diffusion in the Müller-Brown potential, where the sensitivity of a transition rate to landscape parameters can be computed exactly. We then use sensitivities to optimize the directional bias of a particle-based model of a catalysis-driven molecular motor.

I. INTRODUCTION

Many models of complex nonequilibrium systems are controlled by a large number of parameters, and optimizing performance requires sensitivities—derivatives of dynamical observables with respect to those parameters. One might use gradient descent to find a time-varying protocol that drives self-assembling components into a desired product in finite time[1,2]. Alternatively, one could tune the interaction strengths between motifs of a chemically fueled molecular motor to increase its directional bias[3]. In either case, each step of the optimization requires an efficient way to compute sensitivities of the relevant dynamical observable with respect to many parameters simultaneously.

Recent years have seen rapid progress in computing such sensitivities for molecular systems. Automatic differentiation through molecular dynamics trajectories can yield gradients of thermodynamic and kinetic observables with respect to all force-field parameters simultaneously[4–7], and the same technique has been applied to optimize nonequilibrium control protocols[8] and to fit coarse-grained models of nucleic acids to experimental data[9]. Likelihood ratio (score function) estimators[10] and trajectory-based estimators[11–14] offer complementary routes that do not require differentiating through the integrator but instead reweight, perturb, or couple trajectories.

These methods work well when trajectories are long enough to observe the events that determine the observable. The deeper obstacle is rare events. When the observable is controlled by infrequent barrier crossings, a finite simulation may observe none of them. No estimator can extract a gradient signal that the simulation does not contain. Markov state models were developed precisely to address this timescale separation[15–17]. By coarse-graining the state space and building a transition matrix from short trajectory fragments, an MSM captures the slow dynamics without requiring any single trajectory to span the rare crossings. Related Galerkin approximation schemes for dynamical observables share the same timescale-separation strategy[18,19].

Differentiating through coarse-grained kinetic models provides a more direct route to the rare-event regime. Indeed, Trubiano and Hagan [2] combined MSM analysis with adjoint-based gradients to optimize time-dependent self-assembly protocols, exploiting the fact that the sensitivity of a Markov chain's stationary distribution to its transition matrix is a well-studied algebraic problem[20,21]. That MSM-based approach, however, operated in a regime where the MSM could be built from direct simulation and the gradient of interest was the response of a finite-horizon yield to a time-dependent protocol. In the rare-event regime, the stationary distribution itself must be inferred from reweighted data, and the sensitivity requires differentiating not just through the transition matrix but through the importance weights that reconstruct the stationary measure from short trajectory fragments.

What has been missing is a pipeline suited to the rare-event regime, where the stationary distribution cannot be sampled directly and must instead be inferred. We describe such a pipeline here. We adapt the recently introduced RiteWeight algorithm[22], which iteratively refines importance weights for steady-state estimation, to simultaneously track the derivative of those weights with respect to model parameters. The result is a practical sensitivity estimator that maintains performance as the temperature decreases and rare transitions become exponentially less frequent. We validate the method on the Müller-Brown potential[23], a two-dimensional landscape with well-separated metastable states whose exact sensitivities are available for comparison, and show that

[a)] Author to whom correspondence should be addressed: todd.gingrich@northwestern.edu

accuracy is maintained as the temperature is lowered. We then apply the MSM-based sensitivity pipeline to the particle-based catenane motor model of Albaugh and Gingrich [3], demonstrating that gradient descent on the interaction parameters progressively improves the motor's directional bias.

The manuscript proceeds as follows. Section II establishes the sensitivity formula, decomposing the derivative of a reweighted dynamical expectation into a propagator derivative, computable from trajectory data, and an importance weight derivative obtained from the algebraic sensitivity of the MSM stationary distribution. Section III develops the RiteWeight adaptation for carrying sensitivity derivatives through the iterative reweighting. Section IV validates the method on the Müller-Brown potential across temperatures, bin counts, and lag times. Section V applies the method to optimize the molecular motor.

II. THE SENSITIVITY OF A DYNAMICAL ORDER PARAMETER

This section derives the working formula for the sensitivity of a steady-state observable to a model parameter. The key step is a decomposition of the derivative into two pieces with very different characters. One piece involves only single-step transition probabilities: how does each individual step of the dynamics change when the parameter is varied? That piece is straightforward to estimate from trajectory data. Because the MSM decomposition only evaluates it on short trajectory fragments, its variance stays controlled even as the rare-event timescale grows. The other piece involves the stationary distribution itself: how does the long-time probability of being in each part of state space change with the parameter? That piece is the hard one. The stationary distribution is the eigenvector of the dynamics, and there is generally no closed-form expression for how it responds to a perturbation. The rest of this section shows how to handle both pieces. For the first, we differentiate the integrator's transition kernel explicitly. For the second, we introduce importance weights whose derivatives can be estimated via a Markov state model.

Let Z_t be a Markov process with transition kernel P_λ,

$$\mathbb{P}_\lambda[Z_{dt} = y \mid Z_0 = x] = P_\lambda(x, y), \tag{1}$$

depending smoothly on a parameter λ. We denote the stationary expectation of a dynamical observable $F(Z_0, \ldots, Z_t)$ by

$$\langle F \rangle_{\pi_\lambda} \equiv \int F(z_0, \ldots, z_t)\, \pi_\lambda(z_0)\, dz_0 \prod_{s=1}^{t} P_\lambda(z_{s-1}, z_s)\, dz_s, \tag{2}$$

where π_λ is the stationary distribution of P_λ. The form is quite general. F could be an instantaneous positional observable, a configurational order parameter, or a path observable like a transition rate or molecular motor's directional bias.

Differentiating Eq. (2) with respect to λ yields the two-term decomposition anticipated above:

$$\begin{aligned}\frac{d\langle F\rangle_{\pi_\lambda}}{d\lambda} &= \left\langle F\, \frac{d \log \pi_\lambda(Z_0)}{d\lambda} \right\rangle_{\pi_\lambda} \\ &+ \left\langle F\, \frac{d}{d\lambda} \log\big[P_\lambda(Z_0, Z_1) \cdots P_\lambda(Z_{t-1}, Z_t)\big] \right\rangle_{\pi_\lambda}.\end{aligned} \tag{3}$$

The second term is the log-derivative of the path probability. As the sum of single-step log-derivatives, it is directly computable from trajectory data for any model with an explicit transition kernel. The same quantity appears as the reweighting factor when parameters are perturbed within transition path sampling[24]. It is known as the score function or REINFORCE estimator in the stochastic optimization literature[10,25] and as the Girsanov weight in the theory of stochastic processes[26]. The first term of Eq. (3) is the hard one. It requires π_λ (the top eigenvector of the Markov process) and its derivative with respect to λ. We are focused on situations in which the state space is sufficiently large (often continuous) that it is impractical to explicitly compute π_λ, much less its sensitivity to λ. For discrete-state Markov jump processes, exact identities based on the matrix tree theorem and related graph-theoretic tools can bound or express these sensitivities without enumerating the full state space[27–31]. For continuous-state diffusions, information-theoretic methods based on Rényi divergences provide rigorous bounds on rare-event sensitivities directly from path-space relative entropies[32]. Our approach is more computational in nature; we want to estimate the sensitivity with a sampling procedure.

To make the stationary distribution derivative tractable, it is useful to rewrite Eq. (2) as an expectation not over π_λ but over a fixed reference distribution μ. Let μ be any distribution whose support contains that of π_λ, and define $w_\lambda(x) = \pi_\lambda(x)/\mu(x)$ as importance weights that encode all the λ-dependence of the stationary measure. Then

$$\langle F \rangle_{\pi_\lambda} = \langle w_\lambda\, F \rangle_\mu, \tag{4}$$

and differentiating both sides gives

$$\begin{aligned}\frac{d\langle F\rangle_{\pi_\lambda}}{d\lambda} &= \left\langle F\, \frac{dw_\lambda(Z_0)}{d\lambda} \right\rangle_{\mu} \\ &+ \left\langle w_\lambda\, F\, \frac{d}{d\lambda} \log\big[P_\lambda(Z_0, Z_1) \cdots P_\lambda(Z_{t-1}, Z_t)\big] \right\rangle_{\mu}.\end{aligned} \tag{5}$$

The structure of the two terms is the same as before, but the troublesome $d\log\pi_\lambda/d\lambda$ has been replaced by $dw_\lambda/d\lambda$, the derivative of the importance weights. The advantage of this rewriting is twofold. First, the reference distribution μ can be chosen to have good coverage of the full state space—including rare-event regions—regardless of whether those regions are heavily weighted

by π_λ. By sampling trajectory starts from μ (for instance, uniformly over a region of interest) rather than from π_λ directly, one can populate rare regions without waiting for long spontaneous trajectories to visit them. Second, because $w_\lambda = \pi_\lambda/\mu$ is a ratio of probability densities, it can be approximated by a Markov state model at the coarse level. The MSM need only estimate the relative probability that the stationary measure assigns to each coarse cell, a much more tractable target than computing $d\log\pi_\lambda/d\lambda$ directly.

A. Computing the Importance Weight Derivatives via an MSM

The key observation is that $w_\lambda(x) = \pi_\lambda(x)/\mu(x)$ need not be estimated pointwise across the continuous space of microstates x. If we partition microstates into coarse cells $\{S_j\}$, we can instead approximate $w_\lambda(x)$ as a piecewise constant function taking the value $w_{\lambda,j}$ for all $x \in S_j$. Those weights for each cell are approximated as c_j/μ_j, where now c_j and μ_j are the probabilities of occupying cell j in the stationary and reference distributions, respectively. Procedurally, one chooses μ, which sets the fraction of trajectories initialized in cell j: μ_j. From those trajectories, one infers a Markov state transition matrix and the j^{th} component of the stationary vector for this transition matrix is the estimate for π_λ in cell j: c_j.

More formally, let N trajectory fragments be initialized in μ at time 0 and propagated to time t: $\{(Z_0^i,\dots,Z_t^i)\}_{i=1}^N$. Let J_s^i denote the coarse cell index of trajectory number i at time s, i.e. $Z_s^i \in S_{J_s^i}$. The coarse transition matrix is estimated as

$$P_{ab} = \frac{\sum_i \mathbb{1}_{S_a}(Z_0^i)\,\mathbb{1}_{S_b}(Z_t^i)}{\sum_i \mathbb{1}_{S_a}(Z_0^i)}, \tag{6}$$

where $\mathbb{1}_{S_a}(x)$ is the indicator function, equal to 1 when $x \in S_a$ and 0 otherwise. The number of coarse states is sufficiently modest that it is straightforward to numerically compute the top eigenvector of that transition matrix, which satisfies $\sum_j c_j P_{jk} = c_k$. The estimated importance weight contributed by trajectory i is

$$w_\lambda(Z_0^i) = \frac{c_{J_0^i}}{\sum_\ell \mathbb{1}_{S_{J_0^i}}(Z_0^\ell)}. \tag{7}$$

Since w_λ depends on λ only through the coarse stationary distribution c and the transition matrix P, the chain rule gives

$$\frac{dw_\lambda(Z_0^i)}{d\lambda} = \frac{1}{\sum_\ell \mathbb{1}_{S_{J_0^i}}(Z_0^\ell)} \sum_{pq} \frac{dc_{J_0^i}}{dP_{pq}} \frac{dP_{pq}}{d\lambda}. \tag{8}$$

The two factors reflect the two timescales of the problem. The first factor, dc_k/dP_{pq}, encodes how the long-time coarse-grained populations respond to a change in a single transition matrix element. Because the coarse chain has only M states, this is a finite linear algebra problem. We use the perturbation formula of Thiede, Van Koten, and Weare [21],

$$\frac{dc_k}{dP_{pq}} = c_p\left(\mathbb{E}_{J_0=q}\left[\sum_{s=0}^{\tau_p-1}\mathbb{1}_{J_s=k}\right] - c_k\,\mathbb{E}_{J_0=q}[\tau_p]\right), \tag{9}$$

where $\mathbb{E}_{J_0=q}$ denotes the expectation conditional on the coarse chain starting in state q and $\tau_p = \min\{s>0 : J_s = p\}$ is the first return time to state p. The factor in parentheses measures how much time the chain started at q spends in state k before returning to p, relative to the expected return time weighted by c_k—a mean-first-passage-time calculation that reduces to a linear system. Specifically, defining $F_{qpk} = \mathbb{E}_{J_0=q}[\sum_{s=0}^{\tau_p-1}\mathbb{1}_{J_s=k}]$ as the expected number of visits to k before the chain first returns to p, for fixed p and k this satisfies the Feynman-Kac system

$$\sum_{r\neq p}(I_{qr} - P_{qr})\,F_{rpk} = \delta_{qk}, \qquad F_{ppk} = 0, \tag{10}$$

where the linear system is solved over $r, q \neq p$ with p acting as an absorbing index, and the boundary $F_{ppk} = 0$ encodes the convention that no visits are accumulated once the chain has reached p. The system is solved for all (p,k) pairs by standard linear solvers, with the mean return time following as $\mathbb{E}_{J_0=q}[\tau_p] = \sum_k F_{qpk}$. These expected visit counts and return times are properties of the estimated coarse transition matrix P; they are obtained by solving the linear system above, not by directly observing $q \to p$ first-passage events in the trajectory data. Consequently, a slow $q \to p$ process poses no obstacle to evaluating dc_k/dP_{pq}. Even if no fragment ever realizes that transition, the matrix element P_{pq} and the resulting first-passage statistics are still well defined and computed exactly by linear algebra on the finite coarse chain. What must be sampled adequately are the short-time transitions that populate P, not the rare long-time crossings themselves. The second factor, $dP_{ab}/d\lambda$, encodes how the short-time, local transition probabilities between coarse cells change with the parameter. We compute it by applying Eq. (5) with the indicator observable $F(Z_0^i,\dots,Z_t^i) = \mathbb{1}_{S_a}(Z_0^i)\,\mathbb{1}_{S_b}(Z_t^i)$, recovering $dP_{ab}/d\lambda$ from the same trajectory data used for everything else.

B. Derivative of the Propagator

The second term in Eq. (5) requires the log-derivative of the transition kernel, $d\log P_\lambda/d\lambda$, summed over the steps of each trajectory. The number of terms in this sum is set by the lag time, not by the rare-event timescale. Each trajectory fragment spans only the lag time t, which is chosen far shorter than the mean first-passage time $1/k$ (Sec. IV), so the sum runs over the modest number of integrator steps in a single lag window rather than over the

enormous number of steps needed to observe a spontaneous rare transition. This is precisely what the MSM decomposition buys: the score-function sum never accumulates over the full $1/k$ horizon. For any integrator with an explicit transition density, this derivative can be computed analytically from the noise realization at each step. The concrete form depends on the integrator. For overdamped Brownian dynamics discretized with the Euler-Maruyama scheme, the propagator is Gaussian in the positions and the log-derivative is straightforward (Section IV). For Langevin dynamics with momenta, the choice of splitting scheme matters. We use the ABOBA integrator, whose propagator has a closed Gaussian form that makes its log-derivative analytically tractable, and derive the explicit expression in Section V C.

III. REDUCING APPROXIMATION ERROR WITH RITEWEIGHT

The importance weight formula of Section II A approximates the true change of measure by a piecewise-constant function defined by the coarse partition. If the partition is coarse, the approximation can be poor; if it is fine, the MSM may not be well-estimated from the available data. The recently introduced RiteWeight algorithm[22] iteratively reduces this approximation error without requiring a finer partition. Here we describe how to adapt RiteWeight to track not just the weights but also their derivatives with respect to λ.

The idea is simple. At each iteration, we use the current weights to build a new MSM, extract a new coarse stationary distribution, and use it to correct the weights. If the initial coarse partition was too coarse to resolve the stationary distribution accurately, the corrected weights will better represent the true measure, and the next MSM built from those corrected weights will be more accurate still. The derivative of the weights can be carried through each iteration by the product rule, so the sensitivity estimate improves alongside the weights themselves.

At iteration α, we have weights $\{w_\lambda^\alpha(Z_0^i)\}$ and a coarse partition $\{S_j^\alpha\}$. The reweighted coarse transition matrix is

$$P_{ab}^\alpha = \frac{\sum_i w_\lambda^\alpha(Z_0^i)\,\mathbb{1}_{S_a^\alpha}(Z_0^i)\,\mathbb{1}_{S_b^\alpha}(Z_t^i)}{\sum_i w_\lambda^\alpha(Z_0^i)\,\mathbb{1}_{S_a^\alpha}(Z_0^i)}, \tag{11}$$

with stationary distribution c^α satisfying $\sum_a c_a^\alpha P_{ab}^\alpha = c_b^\alpha$. The weights are then updated as

$$w^{\alpha+1}(Z_0^i) = (1-\eta^\alpha)w^\alpha(Z_0^i) + \eta^\alpha \frac{w^\alpha(Z_0^i)\,c_{J_0^i}^\alpha}{\sum_\ell w^\alpha(Z_0^\ell)\,\mathbb{1}_{S_{J_0^i}^\alpha}(Z_0^\ell)}, \tag{12}$$

where $\eta^\alpha \in (0,1]$ is a learning rate. Taking the derivative of this update with respect to λ gives the companion recursion for $dw_\lambda^\alpha/d\lambda$:

$$\begin{aligned}\frac{dw_\lambda^{\alpha+1}(Z_0^i)}{d\lambda} &= (1-\eta^\alpha)\frac{dw_\lambda^\alpha(Z_0^i)}{d\lambda} \\ &+ \eta^\alpha \frac{dw_\lambda^\alpha(Z_0^i)}{d\lambda}\frac{c_{J_0^i}^\alpha}{\sum_\ell w^\alpha(Z_0^\ell)\,\mathbb{1}_{S_{J_0^i}^\alpha}(Z_0^\ell)} \\ &+ \frac{\eta^\alpha\, w^\alpha(Z_0^i)}{D_i^\alpha}\sum_{ab}\frac{dc_{J_0^i}^\alpha}{dP_{ab}^\alpha}\frac{dP_{ab}^\alpha}{d\lambda} \\ &- \frac{\eta^\alpha\, w^\alpha(Z_0^i)\,c_{J_0^i}^\alpha}{(D_i^\alpha)^2}\sum_\ell \frac{dw^\alpha(Z_0^\ell)}{d\lambda}\,\mathbb{1}_{S_{J_0^i}^\alpha}(Z_0^\ell). \end{aligned} \tag{13}$$

Here $D_i^\alpha = \sum_\ell w^\alpha(Z_0^\ell)\,\mathbb{1}_{S_{J_0^i}^\alpha}(Z_0^\ell)$ is the reweighted count in the coarse state containing Z_0^i. The coarse stationary distribution derivative $dc_{J_0^i}^\alpha/dP_{ab}^\alpha$ is computed via Eq. (9) with the reweighted transition matrix, and $dP_{ab}^\alpha/d\lambda$ is estimated as in Section II A using the current weights.

The four terms have a clear structure. The first two terms propagate the existing weight derivative forward, modulated by the new coarse stationary distribution—they are the counterpart, for the derivative, of the weight update itself. The third term is the genuinely new piece. It accounts for how the coarse stationary distribution changes with λ at this iteration, injecting fresh sensitivity information from the current MSM. The fourth term corrects for the fact that the normalization denominator D_i^α itself depends on the weights and therefore on λ. Each RiteWeight iteration therefore carries along a sensitivity estimate at no fundamental change in algorithmic complexity.

In practice, the partition $\{S_j^\alpha\}$ is redrawn at each iteration (for instance, by choosing new Voronoi centers from the dataset), so successive iterations probe the state space at different resolutions. The learning rate η^α controls how aggressively the weights are updated. We find that a few iterations with $\eta^\alpha = 1$ suffice for the problems studied here, consistent with the findings of Kania *et al.*[22].

IV. MÜLLER-BROWN POTENTIAL

We first validate the method on the Müller-Brown (MB) potential[23]:

$$\begin{aligned}V_{\mathrm{MB}}(y,z) = \frac{1}{20}\sum_{i=1}^{4} C_i \exp\big[a_i(y-y_i)^2 + \\ b_i(y-y_i)(z-z_i) + c_i(z-z_i)^2\big],\end{aligned} \tag{14}$$

a canonical two-dimensional landscape with well-separated metastable states connected by a saddle point. The MB potential is small enough to admit an exact treatment. The transition rate k_{AB} and its sensitivity

to a landscape parameter can be computed on a discrete spatial grid by direct matrix methods, with no sampling noise. This exact check is unavailable for the high-dimensional motor of Sec. V, whose configuration space spans the coordinates of dozens of interacting particles. MB therefore provides a rare opportunity to verify the method against ground truth before deploying it where no such check exists.

Figure 1 shows the potential, with the two metastable states A and B highlighted. The parameters are set to

$$\begin{aligned}(C_1, C_2, C_3, C_4) &= (-200, -100, -170, 15),\\ (a_1, a_2, a_3, a_4) &= (-1, -1, -6.5, 0.7),\\ (b_1, b_2, b_3, b_4) &= (0, 0, 11, 0.6),\\ (c_1, c_2, c_3, c_4) &= (-10, -10, -6.5, 0.7),\\ (y_1, y_2, y_3, y_4) &= (1, -0.27, -0.5, -1),\\ (z_1, z_2, z_3, z_4) &= (0, 0.5, 1.5, 1).\end{aligned} \tag{15}$$

The dynamics of the position $X_t = (y, z)_t$ are overdamped Brownian motion discretized with the Euler-Maruyama scheme,

$$X^j_{t+1} = X^j_t - dt\,\nabla V_{\mathrm{MB}}(X_t) + \sqrt{2\,dt\,\beta^{-1}}\,\xi^j_t, \tag{16}$$

where $\beta = 1/k_{\mathrm{B}}T$ is the inverse temperature and ξ^j_t is a standard normal noise. Transitions between A and B constitute the rare events that challenge direct sensitivity estimation. At low temperature, the mean crossing time grows exponentially with the barrier height, so any computationally feasible ensemble of trajectories will observe few or no $A \to B$ events. An estimator built from such an ensemble cannot reliably extract k_{AB}, let alone its sensitivity to a landscape parameter.

The parameters of Eq. (14) are the standard Müller-Brown parameters[23] except that y_2 has been shifted from 0 to -0.27, which deepens the intermediate basin near $(-0.5, 1.5)$. This modification is deliberate. Simply lowering the temperature on the unmodified landscape makes barrier crossings rarer but does not make the problem structurally harder, since the kinetics remain effectively two-state. With the deepened intermediate, lowering the temperature has a qualitatively different effect. Trajectories transitioning from A to B must pass through the intermediate basin, which itself becomes a kinetic trap at low temperature. The dwell time in the intermediate grows exponentially as the temperature decreases, so the coarse-grained model must correctly resolve three metastable regions—not merely two—and accurately capture the populations and fluxes among them. The sensitivity to C_2, the depth of this intermediate basin, is therefore a particularly demanding test. An accurate estimate requires the method to faithfully represent how changes to the intermediate affect the overall $A \to B$ rate.

To compute how k_{AB} changes with a landscape parameter such as C_2, we must cast the rate as an expectation of the form (2). This requires a stationary distribution π and a path functional F that depends on only a short

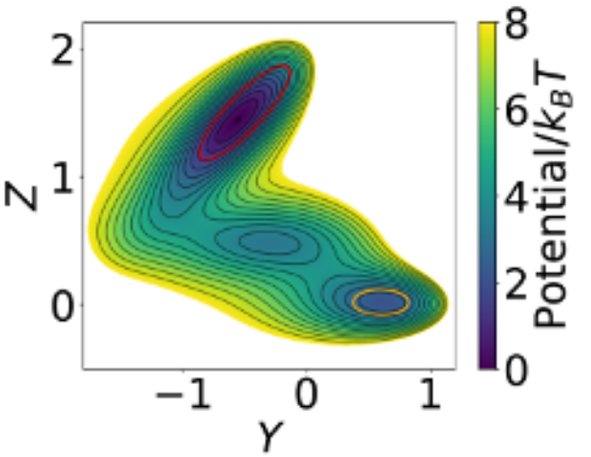

FIG. 1. Müller-Brown potential energy landscape, used here as a validation testbed. The potential has three local minima, two deep metastable states A (upper left, red ellipse) and B (lower right, orange ellipse), and a shallower intermediate basin between them whose depth is controlled by the parameter C_2. At low temperature, transitions between A and B are rare. We test whether the method correctly estimates the sensitivity dk^{-1}_{AB}/dC_2—how the $A \to B$ rate responds to changes in the intermediate's energy—against exact numerical reference values. Contours are spaced every $0.5\,k_{\mathrm{B}}T$.

trajectory segment. A naive choice, taking π to be the Boltzmann distribution and F to count $A \to B$ transitions, fails because the rate is defined through a long-time limit, forcing trajectories long enough to capture the rare crossing. The likelihood ratio estimator for the second term of Eq. (3) then becomes very noisy, since the variance of the score-function sum scales with the trajectory length[33]. Centering strategies can bound this variance for generic steady-state averages, but a cleaner solution is to augment the state space with a label recording whether the system was last in A or in B. In this enlarged space, the transition rate becomes an instantaneous current from the "last in A" state to the "last in B" state, and F need only examine a single timestep. We make this construction precise below and then write k_{AB} as a ratio of two expectations of the form Eq. (2).

The augmented Markov process is $Z_t = (X_t, \mathbb{1}_A(X_{T^-_{A\cup B}}))$, where $T^-_{A\cup B} = \min\{t > 0 : X_{-t} \in A \cup B\}$ records the time of the most recent visit to A or B. The augmented variable is simply a binary label. It equals 1 if the system was last in A and 0 if it was last in B. The $A \to B$ transition rate is the instantaneous flux out of the "last in A" population,

$$k_{AB} = \lim_{t\to 0} \frac{1}{t} \frac{\langle \mathbb{1}_A(X_{T^-_{A\cup B}})\,\mathbb{1}_B(X_t)\rangle_\pi}{\langle \mathbb{1}_A(X_{T^-_{A\cup B}})\rangle_\pi}, \tag{17}$$

where π now denotes the stationary distribution of the augmented process. This is the standard reactive flux ex-

pression for the transition rate[34,35]. The numerator measures the probability that a trajectory last in A reaches B within time t. The prefactor $1/t$ converts this to a flux. The denominator is the fraction of time the system spends having last visited A.

Equation (17) is a ratio of two expectations over π, each of the form treated in Eq. (2). We apply Eq. (5) to compute the sensitivity of both the numerator and denominator to a landscape parameter, then combine using the quotient rule to get the sensitivity of k_{AB}. Both the numerator and denominator depend on λ through the same stationary distribution π_λ, so the importance weights and their MSM derivatives need only be computed once. The same is true of the propagator log-derivative, which depends on the integrator and the landscape, not on the choice of observable F. For the Euler-Maruyama scheme used here, the propagator is Gaussian and the log-derivative takes the form

$$\frac{d\log P_\lambda(X_{t+1}\mid X_t)}{d\lambda} = -\sqrt{\frac{dt\,\beta}{2}}\sum_j \xi_t^j\,\frac{d(\nabla V_\lambda)^j(X_t)}{d\lambda}. \tag{18}$$

The same construction applies to parameters beyond the potential. Differentiating the Euler-Maruyama log-density with respect to the inverse temperature gives

$$\frac{\partial\log P_\beta(X_{t+1}\mid X_t)}{\partial\beta} = \frac{d}{2\beta} - \frac{|X_{t+1}-X_t+\nabla V(X_t)\,dt|^2}{4\,dt}, \tag{19}$$

where d is the dimensionality and the force is evaluated at the pre-step configuration. The expression has zero mean under the dynamics, as a score function must, and substituting it for Eq. (18) yields temperature sensitivities with no other change to the pipeline.

The exact rate is computed on a discrete spatial grid using the scheme of Lorpaiboon, Weare, and Dinner[19]. The sensitivity dk_{AB}^{-1}/dC_2 is obtained by central finite differences of the exact inverse rate,

$$\frac{dk_{AB}^{-1}}{dC_2} \approx \frac{k_{AB}^{-1}(C_2+\epsilon)-k_{AB}^{-1}(C_2-\epsilon)}{2\epsilon}, \tag{20}$$

with $\epsilon = 10^{-4}$. To test the method, we sample initial conditions uniformly from the region $V_{\mathrm{MB}}(x) < 7.0$, assign last-visited-state labels, run a short burn-in of 0.1 time units, and propagate each starting condition for a lag time t. We then build a Voronoi partition of the state space by selecting M cluster centers from the dataset via k-means clustering.

The lag time t requires care. The reactive flux expression Eq. (17) involves a $t \to 0$ limit, but that limit cannot be taken literally with finite data. If t is too small, biases from the non-Markovianity of the coarse-grained process make the MSM stationary distribution poorly determined. If t is too large, on the order of $1/k_{AB}$ or longer, trajectory fragments are long enough to capture the rare transition itself, defeating the purpose of the MSM decomposition. A good lag time is large enough for trajectories to explore their local cell but still much shorter than $1/k_{AB}$. At $\beta = 2.0$, $1/k_{AB}$ exceeds our chosen lag time $t = 0.5$ by roughly three orders of magnitude.

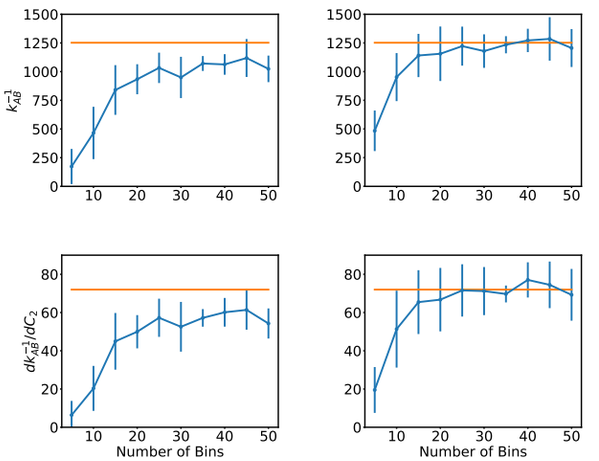

FIG. 2. Sensitivity of the Müller-Brown transition rate to the landscape parameter C_2, as the number of Voronoi cells M is varied. Top row: estimated inverse rate $1/k_{AB}$ from a single-iteration MSM (left) and from RiteWeight (right). Bottom row: estimated sensitivity dk_{AB}^{-1}/dC_2 from an MSM (left) and from RiteWeight (right). Horizontal orange lines indicate the numerically exact reference. Lag time $t = 0.5$, inverse temperature $\beta = 2.0$.

We examine how the estimates depend on the number of Voronoi cells M, the lag time t, and the inverse temperature β, comparing in each case the single-iteration MSM estimate against the RiteWeight estimate. Figure 2 shows the inverse rate and sensitivity to C_2 at $\beta = 2$, using $N = 2\times10^5$ fragments each of length $t = 0.5$, as the number of Voronoi cells M is varied. The RiteWeight estimate converges to the exact reference once the cell count exceeds roughly 15, while the single-iteration MSM retains substantial bias across all tested cell counts. Figure 3 holds $M = 20$ and $N = 2\times10^5$ fixed while sweeping the lag time t. The estimates stabilize once t exceeds roughly 0.5 time units, well below $1/k_{AB}$. In both sweeps, the MSM provides a reasonable but biased first approximation, and the RiteWeight iteration removes most of that bias. Figure 4 tests the method as the inverse temperature β is increased from 0.5 to 2.25, so that the barrier grows and transitions become exponentially rarer. As discussed above, this is precisely the regime where brute-force simulation fails; the question is how the MSM approach scales. We expect to pay for lower temperatures with more data, so the question is not whether N must grow but how fast. Two factors drive the per-trajectory variance of the sensitivity estimator up with β. The score-function term in Eq. (18) carries an explicit factor of $\sqrt{\beta}$, contributing one factor of β to the variance. A second factor

of β comes from the increasing stiffness of the MSM transition matrix as the underlying timescales lengthen because the spectral gap between the stationary eigenvalue and the next-slowest mode shrinks, amplifying any error in the importance weights through the standard eigenvector-sensitivity relation[20,21]. The same scaling has been observed in a reinforcement-learning context by Cheng and Weare[36]; analogous variance challenges arise more broadly in reinforcement-learning approaches to rare-event sampling, where they are addressed by learning control forces that bias the dynamics toward the events of interest[37]. The two combine to give a variance that grows as β^2, and we therefore scale $N = 5\times10^4\,\beta^2$ to hold the statistical error roughly constant across temperatures. On that footing, any divergence in Fig. 4 reflects a coarse-graining error rather than a sample-size artifact. We fix $M = 20$ and $t = 0.5$. The RiteWeight sensitivity estimate tracks the exact reference closely throughout this range, while the MSM error grows with β.

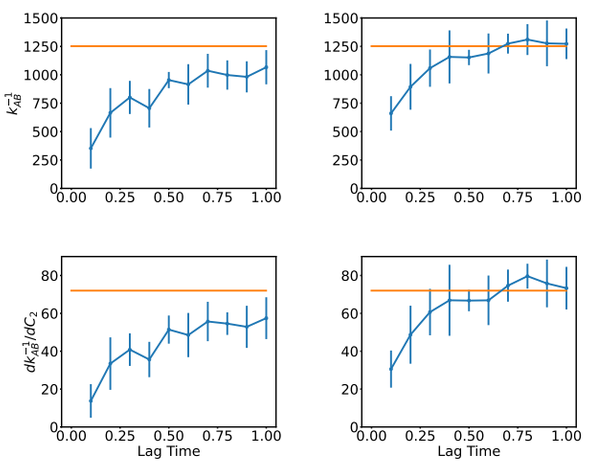

FIG. 3. Same layout as Fig. 2, with $M = 20$ fixed and the lag time t varied. Both the rate and sensitivity estimates stabilize once t is large enough for trajectories to explore their local cell, and remain accurate well below $1/k_{AB}$. Inverse temperature $\beta = 2.0$, $N = 2 \times 10^5$ fragments.

V. OPTIMIZING THE BIAS OF A MOLECULAR MOTOR

The Müller-Brown calculations tested the method against exact reference values in a two-dimensional landscape. We now apply it to a problem where no such reference is available, optimizing the directional bias of a particle-based model of a chemically fueled molecular motor. The motor model, introduced in Albaugh and Gingrich[3] and subsequently studied in Refs.[38–41], involves Langevin dynamics of dozens of interacting particles, a grand canonical Monte Carlo chemostat that drives the system out of equilibrium, and 30 adjustable Lennard-

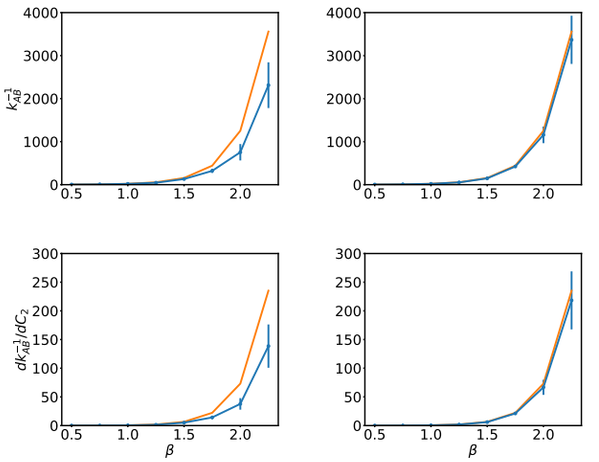

FIG. 4. Performance as the rare-event regime deepens. Inverse rate and sensitivity as the inverse temperature β is increased from 0.5 to 2.25, with $M = 20$ and $t = 0.5$ fixed. The number of trajectory fragments is scaled as $N = 5 \times 10^4\,\beta^2$ to hold the statistical error roughly constant across temperatures (see text). In each panel the orange curve is the numerically exact reference and the blue curve with error bars is the estimate. RiteWeight (right column) maintains accuracy throughout; the single-iteration MSM (left column) shows growing error as β increases.

Jones parameters that control the interactions between the fuel and the catalytic machinery. Computing sensitivities of the motor's bias to all 30 parameters simultaneously would be impractical by finite differences, but the MSM-based pipeline of Section II yields the full gradient from a single ensemble of short trajectories. Gradient descent on that gradient climbs the bias from about 50% to above 90% within roughly a dozen iterations, with parameter changes of at most tens of percent—a small, structured correction to a seemingly reasonable initial force field.

A. Motor Model

The motor is a [2]-catenane; it has two interlocked rings that cannot separate but can rotate relative to each other (Fig. 5). A large 30-bead ring serves as a track for a smaller 12-bead shuttling ring. Most beads on the large ring are inert and purely volume-excluding. Two binding-site beads, located at diametrically opposite positions, attract the shuttling ring, and adjacent to each binding site in the clockwise direction is a three-bead catalytic site (CAT1, CAT2, CAT3) that interacts with fuel molecules. The fuel is a tetrahedral cluster of four particles (TET) enclosing a central blocking-group particle (C). The intact cluster is called a full tetrahedral cluster (FTC); when the catalytic site stretches the cage and releases the blocking group, the remaining empty tetrahe-

dral cluster (ETC) and free C particle dissociate. Three grand canonical Monte Carlo chemostats maintain the populations of FTC, ETC, and C at prescribed chemical potentials, holding the system out of equilibrium with a thermodynamic driving force that favors FTC decomposition[3].

Once deposited on the catalytic ring, the blocking group prevents the shuttling ring from passing. Directionality arises because the shuttling ring, when bound to a binding site, sterically blocks fuel molecules from accessing the adjacent clockwise catalytic site while leaving the counterclockwise site accessible. Blocking groups therefore accumulate preferentially in one direction, gating the shuttling ring's diffusion and producing a net current[3,38]. The motor functions only within a narrow window of parameters. The blocking group must dissociate neither too quickly (gating fails) nor too slowly (ring locks), and the shuttling ring's binding must be strong enough to suppress catalysis but not so strong as to immobilize it. These competing requirements make gradient-based optimization a natural strategy.

The rings are held together by FENE bonds between adjacent beads and harmonic angle potentials that maintain circular geometry. All pairwise interactions are described by a modified Lennard-Jones potential,

$$V_{ij}^{LJ}(r_i, r_j) = 4\epsilon_{ij}^{r}\left(\frac{\sigma_{ij}}{|r_i - r_j|}\right)^{12} - 4\epsilon_{ij}^{a}\left(\frac{\sigma_{ij}}{|r_i - r_j|}\right)^{6}, \tag{21}$$

where σ_{ij} is a pair-specific length scale and the repulsive and attractive amplitudes ϵ_{ij}^{r} and ϵ_{ij}^{a} are allowed to differ, reducing to the standard form when $\epsilon^{a} = \epsilon^{r}$; purely volume-excluding pairs have $\epsilon^{a} = 0$. All bond, angle, mass, and Lennard-Jones radius parameters are identical to those in Ref.[3]; only the Lennard-Jones amplitudes governing fuel–catalytic-site interactions are varied in the optimization. The starting values and simulation parameters are collected in Tables I and II.

B. Directional Bias and Its Gradient

The quantity we wish to maximize is the directional bias of the motor. Let $N_{\mathrm{CW}}^{\mathcal{T}}$ and $N_{\mathrm{CCW}}^{\mathcal{T}}$ be the number of complete clockwise and counterclockwise cycles of the shuttling ring in a long interval $[0, \mathcal{T}]$. The bias is

$$B = \lim_{\mathcal{T}\to\infty} \frac{N_{\mathrm{CW}}^{\mathcal{T}}}{N_{\mathrm{CW}}^{\mathcal{T}} + N_{\mathrm{CCW}}^{\mathcal{T}}}. \tag{22}$$

To write B as a stationary expectation of the form (2), we introduce the discrete displacement J_t of the shuttling ring. Letting j_t be the index of the track particle closest to the ring's center of mass at time t, that discrete displacement is given by

$$J_{t+dt} = J_t + (j_{t+dt} - j_t) - 30\ \mathrm{nint}\left[\frac{j_{t+dt} - j_t}{30}\right], \tag{23}$$

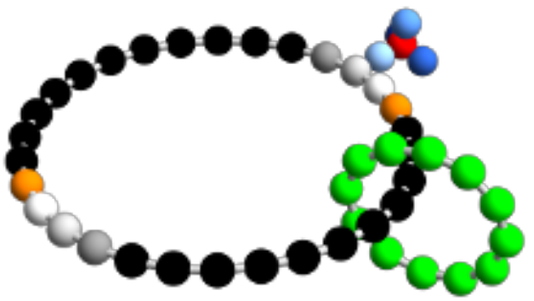

FIG. 5. Illustration of the catenane motor system. The shuttling ring (green) diffuses around the large track ring, whose inert beads are shown in black. Two binding sites (orange beads) attract the shuttling ring. Adjacent to each binding site in the clockwise direction is a three-bead catalytic site (CAT1, CAT2, CAT3), shown in shades of gray, lightest nearest the binding site. A tetrahedral fuel cluster (shades of blue) carries a blocking group (red) to the catalytic site, where interactions between the cluster and the catalytic beads catalyze release of the blocking group onto the central catalytic bead, CAT2. Once deposited, the blocking group prevents the shuttling ring from passing.

where nint[·] denotes the nearest integer, so each increment of J is a signed minimum-image step of the shuttling ring. Unlike the site index j_t, which runs from 0 to 29, the displacement J_t is signed; it starts at 0 when a cycle begins and ranges over the 59 values $-29, \ldots, 29$. A complete clockwise (counterclockwise) cycle corresponds to J crossing from 29 (from -29) back to 0. If $\pi(X, J)$ is the joint stationary distribution of particle positions X and displacement J, then

$$\begin{aligned} \dot{N}_{\mathrm{CW}} &= \lim_{dt\to 0} \frac{1}{dt} \left\langle \mathbb{1}_{J_0=29}\, \mathbb{1}_{J_{dt}=0} \right\rangle_{\pi}, \\ \dot{N}_{\mathrm{CCW}} &= \lim_{dt\to 0} \frac{1}{dt} \left\langle \mathbb{1}_{J_0=-29}\, \mathbb{1}_{J_{dt}=0} \right\rangle_{\pi}, \end{aligned} \tag{24}$$

where a completed clockwise cycle is registered when J steps from 29 back to 0 and a counterclockwise cycle when J steps from -29 back to 0. The bias $B = \dot{N}_{\mathrm{CW}}/(\dot{N}_{\mathrm{CW}} + \dot{N}_{\mathrm{CCW}})$ is a ratio of two expectations over π, each of the form treated in Eq. (2). The structure is the same as the Müller-Brown rate. B is a ratio of two $\langle F \rangle_{\pi}$ objects, so its gradient follows from the quotient rule applied to Eq. (5), and the MSM importance weights and propagator derivatives are shared between numerator and denominator.

TABLE I. Simulation and gradient-descent parameters for the molecular motor optimization. All quantities are in the non-dimensionalized units of Ref.[3]. The GCMC values μ_i' are the shifted chemical potentials $\mu_i - A_i^0$ defined in that reference, which absorb the cluster free energies and are the quantities that enter the Monte Carlo acceptance probabilities.

Parameter	Symbol	Value
Gradient descent		
Number of adjustable parameters	—	30
Learning rate	η	0.007
Maximum step cap	c	0.1
MSM / trajectory data		
Number of MSM coarse states	M	531
Lag time	τ	0.1
Trajectories per coarse state	—	10
Trajectory length	—	80
Langevin dynamics		
MD timestep	dt	5×10^{-3}
Inverse temperature	β	2.0
Damping coefficient	γ	1.0
GCMC chemical potentials		
Filled tetrahedral cluster (FTC)	μ'_{FTC}	1
Empty tetrahedral cluster (ETC)	μ'_{ETC}	−3
Central particle (C)	μ'_{C}	−10

TABLE II. Starting Lennard-Jones parameters for the fuel–motor interactions that are optimized by gradient descent. BG: blocking group (red in Fig. 5); CAT1, CAT2, CAT3: catalytic beads, shown white to gray in order of increasing distance from the binding site; TET: tetrahedral cluster beads (shades of blue); SHUTTLE: green shuttling ring beads. Parameters that are adjusted during the optimization are highlighted in blue.

Interaction	ϵ_r	ϵ_a
BG–CAT1	1	0
BG–CAT3	1	0
BG–CAT2	3.0	4.0
BG–SHUTTLE	100000	0
BG–all others	1	0
TET(1–4)–CAT(1–3)	0.58	0.58
TET(1–4)–all others	1	0

The parameters we optimize are 30 attractive and repulsive Lennard-Jones amplitudes governing fuel–catalytic-site and blocking-group–motor interactions (Table II). Following Ref.[3], the motor is confined to an inner simulation box by a wall potential, and GCMC insertions and deletions act only on a surrounding outer volume. A freshly inserted or deleted molecule therefore has no direct interaction with the motor at the moment of the chemostat move, so the GCMC acceptance probability does not depend on the Lennard-Jones parameters being optimized. Only the Langevin propagator derivative contributes to the bias gradient. All other force-field parameters are held fixed at the values in Ref.[3].

C. Integrator and Propagator Derivative

In the original motor simulations[3], the dynamics were integrated with the Athènes-Adjanor scheme[42]. Here we switch to the ABOBA splitting[43], because ABOBA evaluates the force at a deterministic midpoint configuration rather than at a random one, which gives the propagator a closed Gaussian form and makes its log-derivative analytically tractable. The integrator propagates both positions and momenta, but the friction coefficient ($\gamma = 1.0$) is large enough that inertial effects are negligible on the timescales relevant to the motor; the momentum degrees of freedom are retained because the ABOBA propagator has a convenient analytic form, not because the dynamics are inertially dominated. Letting X_t, P_t be the position and momentum, ξ_t a standard normal random variable, m the particle mass, γ the friction coefficient, and ∇U_λ the force, the ABOBA half-steps are

$$X_{t+1/2} = X_t + \tfrac{dt}{2m} P_t, \tag{25}$$

$$P_{t+1/2} = P_t - \tfrac{dt}{2} \nabla U_\lambda(X_{t+1/2}), \tag{26}$$

$$\hat{P}_{t+1/2} = e^{-\gamma dt} P_{t+1/2} + \sqrt{\beta^{-1} m (1 - e^{-2\gamma dt})}\, \xi_t, \tag{27}$$

$$P_{t+1} = \hat{P}_{t+1/2} - \tfrac{dt}{2} \nabla U_\lambda(X_{t+1/2}), \tag{28}$$

$$X_{t+1} = X_{t+1/2} + \tfrac{dt}{2m} P_{t+1}. \tag{29}$$

Because the force is evaluated only at the non-random midpoint $X_{t+1/2}$, the propagator is Gaussian in P_{t+1}:

$$P_\lambda(X_{t+1}, P_{t+1} \mid X_t, P_t) = (2\pi f^2)^{-d/2} \exp\left[\frac{-(P_{t+1} - e^{-\gamma dt} P_t + G_t)^2}{2 f^2}\right] \times \delta\big(X_{t+1} - X_t - \tfrac{dt}{2m}(P_{t+1} + P_t)\big), \tag{30}$$

where $f = \sqrt{\beta^{-1} m (1 - e^{-2\gamma dt})}$, d is the dimension, and $G_t = \frac{dt}{2}(1 + e^{-\gamma dt}) \nabla U_\lambda(X_{t+1/2})$. Taking the log derivative and recognizing the exponential argument as $\xi_t^T \xi_t / 2$ yields

$$\frac{d \log P_\lambda(X_{t+1}, P_{t+1} \mid X_t, P_t)}{d\lambda} = -\sum_i \xi_t^i \, \frac{d(\nabla U_\lambda)^i(X_{t+1/2})}{d\lambda} \, g^i, \tag{31}$$

where $g^i = dt(1 + e^{-\gamma dt})/[2\sqrt{\beta^{-1} m^i (1 - e^{-2\gamma dt})}]$, and ξ_t^i, m^i are the noise sample and mass for particle i at step t. This gives an explicit, simulation-accessible expression for the propagator derivative in the second term of Eq. (5). The more common BAOAB splitting[43] would not admit such a direct expression because it evaluates the force at a random-input configuration, so its propagator does not have a closed Gaussian form in general. We have verified that the ABOBA dynamics used here produce motor current and bias statistically consistent with those obtained from BAOAB, so the choice of splitting does not affect the physical conclusions.

D. Collective Variables and MSM Construction

The motor's configuration space is far too high-dimensional for a spatial Voronoi partition like the one used for Müller-Brown. Instead, we partition a small set of physically motivated collective variables into discrete MSM states. The discretization is a triplet (J_t, S_t^1, S_t^2). The first index J_t is the signed shuttling-ring displacement of Eq. (23), which takes 59 integer values and tracks the slow progress of the ring around the track. The second and third indices $S_t^i \in \{0, 1, 2\}$ report the state of each catalytic site. $S_t^i = 2$ if a FTC sits within a distance 3.0 of the central catalytic bead and $S_t^i = 1$ if no FTC is nearby but a blocking group lies within 2.6 of that bead, and $S_t^i = 0$ otherwise. Together these give $59 \times 3 \times 3 = 531$ MSM states, resolving both the shuttling ring's position and the gating state at each catalytic site.

The coarse transition matrix and importance-weight derivatives are computed exactly as in Section II A, with Voronoi cell indicators replaced by indicators on these discrete states. Because 531 states is still modest, the linear algebra in Eqs. (8)–(9) is inexpensive. We use a single-iteration MSM for the motor without the RiteWeight refinement of Section III. The discrete labels (J, S^1, S^2) do not come with a natural distance metric for generating candidate Voronoi centers, so the iterative reweighting would require a different construction of refined partitions than the one used for Müller-Brown. The agreement between the MSM-based gradient and direct simulation in the next section (Fig. 6) will confirm that a single MSM pass is adequate here.

E. Optimization Protocol and Results

Gradient descent requires an ensemble of initial configurations that covers the relevant MSM states. At the first step we construct 10 copies of a configuration in each state, about 5310 configurations in total. Each configuration is built by placing both rings as circles, rotating the shuttling ring to the position specified by J, inserting FTC and blocking-group particles to match the target S^1, S^2, and relaxing the result with 15,000 steps of energy minimization at step size 10^{-5}. Some MSM states place the shuttling ring directly on top of a fuel molecule or blocking group; we do not initialize configurations in these high-energy states, which leaves 451 accessible states at the first iteration. Each configuration is then propagated for 80 time units with snapshots every 0.1 time units. For subsequent gradient steps, starting configurations are drawn from the endpoints of the previous iteration's trajectories (10 per occupied MSM state). Parameters are updated as

$$\theta_{i+1} = \max\{0,\ \theta_i + \mathrm{clip}\big(\eta\,(\nabla_\theta B)_i,\ -c,\ c\big)\}\,, \qquad (32)$$

where $\mathrm{clip}(x, -c, c) = \max\{-c, \min\{x, c\}\}$. The cap c prevents integrator instabilities and the outer max enforces non-negativity of the Lennard-Jones amplitudes.

To check that the MSM-based gradient reflects the underlying motor rather than a coarse-graining artifact, we also estimate the bias directly from 50 long trajectories of length 2×10^5 time units at each parameter setting. Each such trajectory spans roughly ten complete cycles, enough to determine the bias reliably.

Figure 6(a) shows the optimization in action. The bias climbs from about 50%—essentially no directional preference—to over 90% within the first several iterations and then plateaus. The MSM-based estimate used to compute the gradient and the independent long-trajectory estimate track each other throughout the climb and plateau, so the gradient direction is not an artifact of the coarse graining. Gradient descent works on this problem.

Gradient descent does not, however, know when to stop. Iterating past the plateau degrades the motor, dropping the independently measured bias in Fig. 6(a) from near 1 at iteration 15 to below 0.5 by iteration 20. The decline is a signal-to-noise problem, not a failure of the gradient formula. As the optimization climbs toward the plateau, the true gradient shrinks; its norm falls nearly an order of magnitude between the climb and the peak, while the statistical error of each iteration's estimate does not shrink with it. A walker-level bootstrap at the peak (iteration 13) quantifies that noise. Resampling the data alone reorients the estimated gradient to a cosine similarity of 0.69 ± 0.16 with the full-data estimate. Once noise dominates, the parameter updates devolve into a random walk, and because working and non-working motors sit close together in parameter space (a point Fig. 7 will make concrete), the walk soon steps off the ridge. Worse, the walk feeds on itself. The parameters themselves show the spiral in action. In the iterations past the plateau, one of the blocking-group–catalyst attractions climbs from 0.77 to 1.17 through four consecutive updates at the maximum allowed step size, with its paired repulsion pinned at the lower bound, while the independently measured bias falls. The motor's slowest relaxation time depends exponentially on exactly this gating amplitude, so the overshoot lengthens the timescales the MSM must resolve, degrades the conditioning of the estimate, and inflates the very gradient variance that produced the overshoot. A machine-learning optimization rarely faces this spiral, because its gradient variance is rarely exponentially sensitive to one parameter, and when it is, the architecture can be regularized; here the "architecture" is fixed by the physics of the interactions, so early stopping is not optional.

The estimator itself gives the problem away. Past the plateau the MSM estimate, which had tracked the long-trajectory measurement faithfully, swings to both sides of it (0.20 versus 0.77 at iteration 18, 0.70 versus 0.48 at iteration 20), a symptom of stratified ensembles that no longer sample the drifting parameters well. As in any stochastic optimization, the remedy is to monitor the gradient magnitude against a resampling estimate of

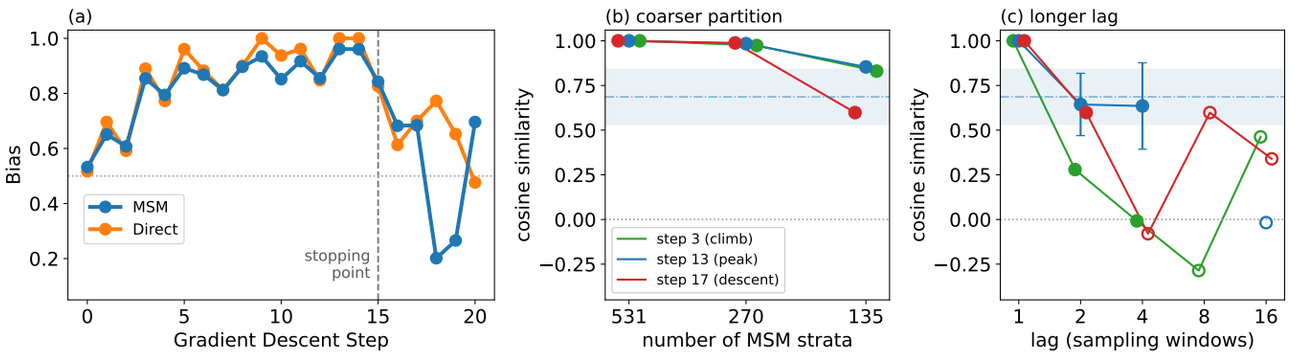

FIG. 6. (a) Motor directional bias B as a function of gradient-descent iteration. Blue circles: bias estimated from the MSM used to compute the gradient. Orange circles: bias estimated independently from 50 long trajectories at each parameter setting. The two estimates track each other through the climb and plateau, confirming that the gradient direction reflects the motor's actual bias rather than an artifact of the coarse graining; they separate once the optimization deteriorates, as discussed in the text. The dashed line marks iteration 15, the early-stopping point whose parameters are analyzed in Fig. 7. (b) Dependence of the gradient on the resolution of the MSM partition. Each point is the cosine similarity, over the optimized parameter components, between the gradient of the production computation (531 strata, lag of one sampling window) and the gradient recomputed after merging adjacent strata in pairs (270 strata) or in fours (135), evaluated at an iteration during the climb (step 3), at the peak (step 13), and during the decline (step 17). (c) The same comparison with the lag time lengthened from one sampling window to 2, 4, 8, or 16. Filled symbols mark settings whose own bias estimate remains within 0.15 of the production value. Open symbols mark settings where the coarsened MSM no longer reproduces the bias itself, so the direction of its gradient carries no information; the absent point (step 13, lag 8) is a disconnected coarse chain. Error bars on the step-13 points give the walker-bootstrap spread, and the shaded band is the noise floor, the spread of cosine similarities between the full-data gradient and gradients recomputed from resampled data at the production settings (0.69 ± 0.16).

its uncertainty and to stop once the signal sinks below the noise. By that criterion the optimization is finished by the plateau, and we take iteration 15 as the optimized design; those are the parameters analyzed in Fig. 7. Unlike a typical machine-learning objective, the physical objective also supplies a verification oracle for the stopped design, and the long trajectories of Fig. 6(a) certify directly that the chosen force field turns the motor.

Because the gradient comes from an MSM, one should ask whether it depends on the two discretionary choices in the MSM's construction, the resolution of the state partition and the lag time. Figures 6(b) and (c) answer by recomputing the gradient with each choice perturbed and comparing the perturbed gradient to the production one through the cosine similarity of their directions, at an iteration during the climb (step 3), at the peak (step 13), and during the decline (step 17). A cosine of 1 means the perturbed MSM would drive the optimization the same way; the shaded band shows how far the cosine falls from 1 under resampling of the data alone, so only excursions below the band signal a genuine dependence on the construction. Merging adjacent strata in pairs, which halves the partition resolution, leaves the direction essentially unchanged; Fig. 6(b) shows cosine similarities above 0.97 at all three iterations. The partition is evidently fine enough that the gradient has converged with respect to refinement, and only a fourfold coarsening begins to reorient it.

Figure 6(c) shows that the lag time matters more, and for a physical reason. The gating events that control cycling decorrelate within a few sampling windows, so lengthening the lag integrates over, and thereby erases, the very transitions the gradient must resolve. Doubling the lag genuinely reorients the gradient during the climb, where the gradient is strong (cosine 0.29); at the peak, where it is weak, the change (cosine 0.63) sits within the sampling noise floor. At the longest lags the coarsened MSM stops reproducing even the bias, reporting below 0.1 where the single-window value is 0.59–0.85. The open symbols in Fig. 6(c) mark those settings, and the seemingly recovered cosines there are the meaningless orientation of a broken estimator, not a return of fidelity. The single-window lag used throughout is validated empirically by Fig. 6(a), since steps taken along its gradient demonstrably increase the independently measured bias.

The more surprising finding in Fig. 7 is that the transition from essentially unbiased to strongly biased motors requires modest nudges of the Lennard-Jones amplitudes. Working and non-working motors, in other words, live close to each other in parameter space. The same observation appears implicitly in the original motor model, where the "Motor I" and "Motor II" variants of Ref.[3] differ by small parameter adjustments and yet show noticably different performance. Our starting parameters were chosen to be a reasonable initial guess in that the necessary attraction between the blocking group and the catalytic site is set somewhat large, and the tetrahedral fuel is weakly attracted to the catalytic site. Despite be-

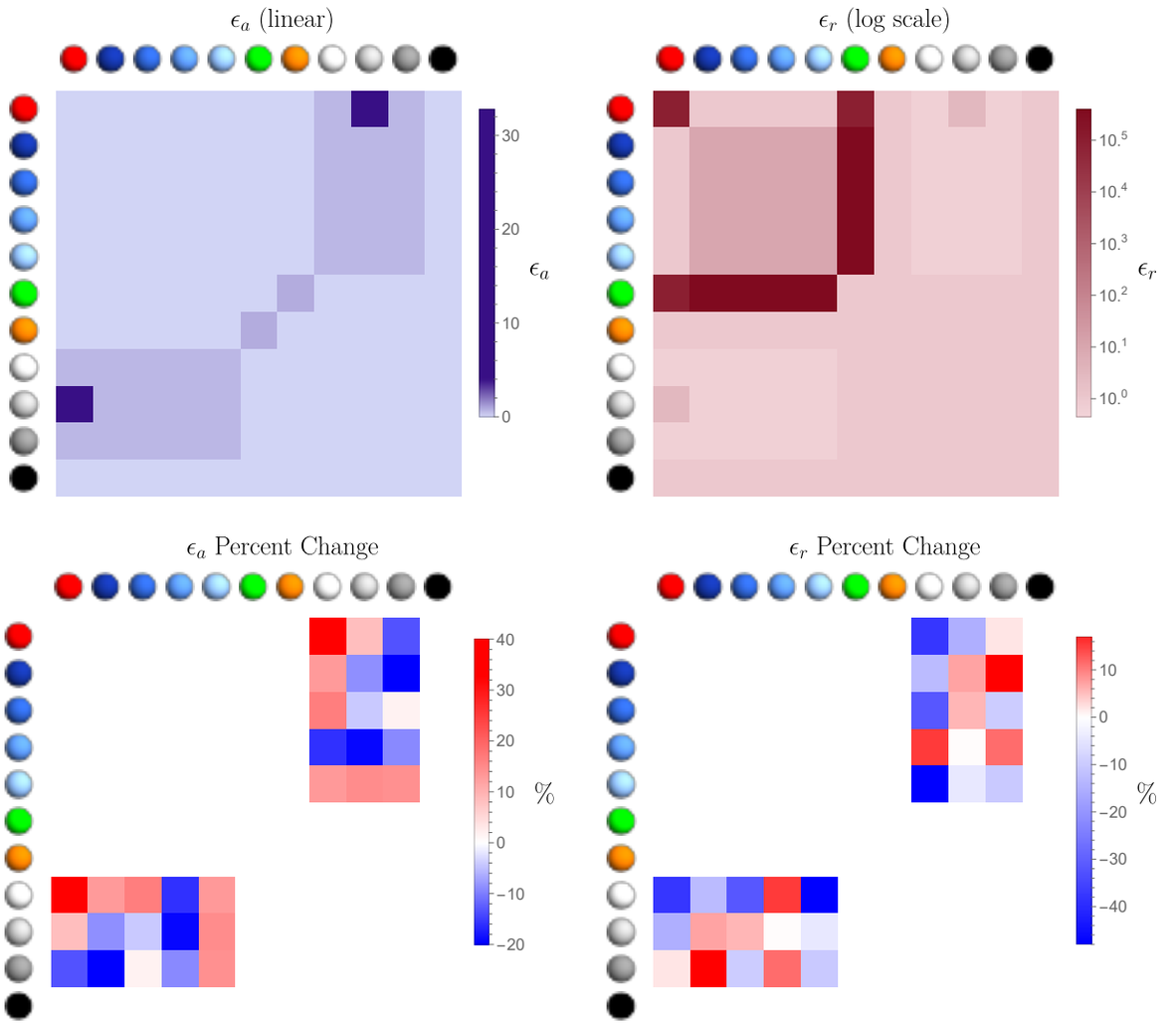

FIG. 7. Top row: starting Lennard-Jones amplitudes; the repulsive parameters ϵ_r are shown on a log scale. Bottom row: percent change in ϵ_a (left) and ϵ_r (right) between the starting force field and the 15th iteration. The attractive amplitude ϵ_a between the fuel and the catalytic bead closest to the binding site (white-ball column) grows, while the amplitude between the fuel and the catalytic bead farthest from the binding site (dark-gray column) shrinks. The repulsive amplitudes ϵ_r show the opposite trend. The overall effect is to shift the catalytic event toward the binding site.

ing a reasonable parameter set, the initial motor design generates no current because directional bias requires a specific pattern of attractions among fuel, catalytic site, and shuttling ring, and small perturbations of that pattern can turn a dud into a working motor.

The structure of the changes in Fig. 7 reveals that pattern. The attractive amplitudes ϵ_a grow between fuel particles and the catalytic bead closest to the binding site and shrink between fuel particles and the catalytic bead farthest from the binding site, with the opposite trend for the repulsive amplitudes ϵ_r. Gradient descent is shifting the catalytic event toward the binding site. That shift amplifies the kinetic asymmetry underlying the motor's Brownian-ratchet mechanism[40,44]. A shuttling ring resting on a binding site sterically suppresses catalysis on the adjacent clockwise site more effectively when that catalytic site is nearby, so biasing catalysis toward the near-side bead magnifies the gating difference between the two cycling directions. The gradient has rediscovered, from a purely numerical criterion, the design principle argued for qualitatively in Refs.[38,40].

Finally, a word on what we have and have not optimized. Our objective is directional bias, not current or thermodynamic efficiency. The three are distinct design targets, and they can trade off against each other. A highly biased motor can turn slowly, and parameter changes that increase current may suppress bias[40]. The pipeline accommodates any of these objectives with a different choice of F in Eq. (2). We selected bias because it has a clean form as a ratio of two counting ex-

pectations and so provides a convenient demonstration that gradient-based optimization can navigate a high-dimensional Lennard-Jones design space.

VI. DISCUSSION

We have presented a method for computing the sensitivity of dynamical order parameters to model parameters in systems with rare events. The method works by separating the problem along the natural timescale separation. Short trajectory data teaches a Markov state model how local rates depend on parameters, and linear algebra propagates those rate changes into changes in the stationary distribution. The RiteWeight iteration systematically reduces the error introduced by the coarse-graining.

The central computational advantage is that we avoid long trajectories. For problems dominated by rare events, the MSM importance weights allow us to represent the stationary distribution and its sensitivity from an ensemble of short trajectories, none of which need to span multiple barrier crossings. The validation on the Müller-Brown potential demonstrates that the method maintains accuracy even as the temperature is lowered and the timescale separation grows.

Beyond demonstrating the pipeline, the motor optimization sheds light on the parameter landscape of the Albaugh and Gingrich[3] model. Seemingly dysfunctional choices of the Lennard-Jones amplitudes sit close in parameter space to working ones, and gradient descent on the bias systematically migrates catalysis toward the binding site—a mechanism consistent with the Brownian-ratchet account of Penocchio *et al.*[40]. The pipeline extends naturally to other systems governed by Langevin dynamics or a similarly explicit, parameterized propagator, provided two ingredients are available: a coarse partition that resolves the slow dynamics and a propagator whose log-derivative can be evaluated. Neither is automatic. Constructing a good partition in high dimensions is itself nontrivial, and integrators without a closed-form propagator require the numerical or adjoint treatment discussed among the limitations below. Nor must the parameters be force-field amplitudes. Any quantity that enters the propagator differentiably can be treated the same way; we have verified for the Müller-Brown system that the inverse temperature β, which enters both the noise amplitude and the normalization of the Euler-Maruyama propagator, yields sensitivities in agreement with finite-difference references. Sensitivity to a chemostatted concentration would additionally require differentiating the grand canonical Monte Carlo acceptance probabilities, a natural extension we have not pursued.

The strategy of differentiating through a coarse-grained kinetic model has appeared in other contexts. Trubiano and Hagan[2] combined MSM analysis with adjoint-based gradients to optimize a time-dependent protocol for maximizing the transient yield of a self-assembly target. The key distinction is that their setting involves a time-varying protocol optimized for a finite-horizon objective, whereas here we compute steady-state sensitivities to static parameters. Both approaches build an MSM and then differentiate through its transition matrix, suggesting that MSM-based gradient pipelines may be broadly useful for design problems in complex molecular systems, whether the objective is a steady-state property or a transient one.

Several limitations of the current approach deserve mention. The method requires a coarse partition of state space, and the quality of the sensitivity estimate depends on how well that partition resolves the relevant dynamics. For the Müller-Brown potential, a Voronoi tessellation with a modest number of cells suffices; for higher-dimensional systems, choosing a good partition is itself a nontrivial problem, and adaptive or data-driven partitioning strategies[15] may be needed. The ABOBA integrator was chosen because it yields an analytically tractable propagator derivative, but this choice is not a fundamental limitation of the framework. Extending the method to integrators whose propagator lacks a closed form would require numerical differentiation or adjoint methods for the propagator derivative. We have also worked throughout with observables that take the form of stationary expectations or ratios of such expectations; extending the pipeline to observables that do not (for instance, entropy production rates or thermodynamic efficiency) would require reworking the path functional F in Eq. (5) but is conceptually straightforward. Finally, the gradient descent in the motor application is not guaranteed to find a global optimum; the parameter landscape may contain local optima or saddle points that trap the optimization.

DATA AND CODE AVAILABILITY

The source code for the motor optimization, the scripts and data that generate the motor figures and the gradient diagnostics, and a self-contained Jupyter notebook that reproduces the Müller-Brown example are openly available in a Zenodo repository at `https://doi.org/10.5281/zenodo.19950483`. Numerical data underlying the motor figures can also be regenerated by running the deposited optimization pipeline with the parameters reported in Tables I and II.

ACKNOWLEDGMENTS

This work was initiated with support from the Gordon and Betty Moore Foundation under Grant Number GBMF10790. This material is additionally based upon work supported by the U.S. Department of Energy, Office of Science, Office of Basic Energy Sciences program under Award Number DE-SC0026333.

[1] C. P. Goodrich, E. M. King, S. S. Schoenholz, E. D. Cubuk, and M. P. Brenner, "Designing self-assembling kinetics with differentiable statistical physics models," Proceedings of the National Academy of Sciences **118**, e2024083118 (2021).
[2] A. Trubiano and M. F. Hagan, "Optimization of non-equilibrium self-assembly protocols using Markov state models," The Journal of Chemical Physics **157**, 244901 (2022).
[3] A. Albaugh and T. R. Gingrich, "Simulating a chemically fueled molecular motor with nonequilibrium molecular dynamics," Nature Communications **13**, 2204 (2022).
[4] J. G. Greener and D. T. Jones, "Differentiable molecular simulation can learn all the parameters in a coarse-grained force field for proteins," PLOS ONE **16**, e0256990 (2021).
[5] W. Wang, S. Axelrod, and R. Gómez-Bombarelli, "Differentiable molecular simulations for control and learning," (2020), arXiv:2003.00868.
[6] S. S. Schoenholz and E. D. Cubuk, "JAX MD: A framework for differentiable physics," Advances in Neural Information Processing Systems **33**, 11428–11441 (2020).
[7] J. G. Greener, "Reversible molecular simulation for training classical and machine-learning force fields," Proceedings of the National Academy of Sciences **122**, e2426058122 (2025).
[8] M. C. Engel, J. A. Smith, and M. P. Brenner, "Optimal control of nonequilibrium systems through automatic differentiation," Physical Review X **13**, 041032 (2023).
[9] R. K. Krueger, M. C. Engel, R. Hausen, and M. P. Brenner, "Fitting coarse-grained models to macroscopic experimental data via automatic differentiation," Proceedings of the National Academy of Sciences **123**, e2508255123 (2026).
[10] P. W. Glynn, "Likelihood ratio gradient estimation for stochastic systems," Communications of the ACM **33**, 75–84 (1990).
[11] M. Rathinam, P. W. Sheppard, and M. Khammash, "Efficient computation of parameter sensitivities of discrete stochastic chemical reaction networks," The Journal of Chemical Physics **132**, 034103 (2010).
[12] P. W. Sheppard, M. Rathinam, and M. Khammash, "A pathwise derivative approach to the computation of parameter sensitivities in discrete stochastic chemical systems," The Journal of Chemical Physics **136**, 034115 (2012).
[13] Y. Pantazis, M. A. Katsoulakis, and D. G. Vlachos, "Parametric sensitivity analysis for biochemical reaction networks based on pathwise information theory," BMC Bioinformatics **14**, 311 (2013).
[14] D. F. Anderson, "An efficient finite difference method for parameter sensitivities of continuous time Markov chains," SIAM Journal on Numerical Analysis **50**, 2237–2258 (2012).
[15] B. E. Husic and V. S. Pande, "Markov state models: From an art to a science," Journal of the American Chemical Society **140**, 2386–2396 (2018).
[16] J.-H. Prinz, H. Wu, M. Sarich, B. Keller, M. Senne, M. Held, J. D. Chodera, C. Schütte, and F. Noé, "Markov models of molecular kinetics: Generation and validation," The Journal of Chemical Physics **134**, 174105 (2011).
[17] C. Schütte, A. Fischer, W. Huisinga, and P. Deuflhard, "A direct approach to conformational dynamics based on hybrid Monte Carlo," Journal of Computational Physics **151**, 146–168 (1999).
[18] E. H. Thiede, D. Giannakis, A. R. Dinner, and J. Weare, "Galerkin approximation of dynamical quantities using trajectory data," The Journal of Chemical Physics **150**, 244111 (2019).
[19] C. Lorpaiboon, J. Weare, and A. R. Dinner, "Augmented transition path theory for sequences of events," The Journal of Chemical Physics **157**, 094115 (2022).
[20] C. D. Meyer, "Sensitivity of the stationary distribution of a Markov chain," SIAM Journal on Matrix Analysis and Applications **15**, 715–728 (1994).
[21] E. Thiede, B. Van Koten, and J. Weare, "Sharp Entrywise Perturbation Bounds for Markov Chains," SIAM Journal on Matrix Analysis and Applications **36**, 917–941 (2015).
[22] S. Kania, R. J. Webber, G. Simpson, D. Aristoff, and D. M. Zuckerman, "Randomized iterative trajectory reweighting for steady-state distributions without discretization error," Proceedings of the National Academy of Sciences **123**, e2529246123 (2026).
[23] K. Müller and L. D. Brown, "Location of saddle points and minimum energy paths by a constrained simplex optimization procedure," Theoretica Chimica Acta **53**, 75–93 (1979).
[24] C. Dellago, P. G. Bolhuis, and P. L. Geissler, "Transition Path Sampling," in *Advances in Chemical Physics*, Vol. 123 (John Wiley & Sons, Ltd, 2002) pp. 1–78.
[25] R. J. Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning," Machine Learning **8**, 229–256 (1992).
[26] B. Øksendal, *Stochastic Differential Equations: An Introduction with Applications*, 6th ed. (Springer, 2003).
[27] J. A. Owen, T. R. Gingrich, and J. M. Horowitz, "Universal Thermodynamic Bounds on Nonequilibrium Response with Biochemical Applications," Physical Review X **10**, 011066 (2020).
[28] G. Fernandes Martins and J. M. Horowitz, "Topologically constrained fluctuations and thermodynamics regulate nonequilibrium response," Physical Review E **108**, 044113 (2023).
[29] T. Aslyamov and M. Esposito, "Nonequilibrium Response for Markov Jump Processes: Exact Results and Tight Bounds," Physical Review Letters **132**, 037101 (2024).
[30] T. Aslyamov and M. Esposito, "General Theory of Static Response for Markov Jump Processes," Physical Review Letters **133**, 107103 (2024).
[31] J. Zheng and Z. Lu, "Universal response inequalities beyond steady states via trajectory information geometry," Physical Review E **112**, L012103 (2025).
[32] P. Dupuis, M. A. Katsoulakis, Y. Pantazis, and L. Rey-Bellet, "Sensitivity analysis for rare events based on Rényi divergence," The Annals of Applied Probability **30**, 1507–1533 (2020).
[33] G. Arampatzis, M. A. Katsoulakis, and L. Rey-Bellet, "Efficient estimators for likelihood ratio sensitivity indices of complex stochastic dynamics," The Journal of Chemical Physics **144**, 104107 (2016).
[34] D. Chandler, "Statistical mechanics of isomerization dynamics in liquids and the transition state approximation," The Journal of Chemical Physics **68**, 2959–2970 (1978).
[35] W. E and E. Vanden-Eijnden, "Transition-path theory and path-finding algorithms for the study of rare events," Annual Review of Physical Chemistry **61**, 391–420 (2010).
[36] X. Cheng and J. Weare, "The surprising efficiency of temporal difference learning for rare event prediction," Advances in Neural Information Processing Systems **37**, 81257–81286 (2024).
[37] A. Das, D. C. Rose, J. P. Garrahan, and D. T. Limmer, "Reinforcement learning of rare diffusive dynamics," The Journal of Chemical Physics **155**, 134105 (2021).
[38] A. Albaugh, G. Gu, and T. R. Gingrich, "Sterically driven current reversal in a molecular motor model," Proceedings of the National Academy of Sciences **120**, e2210500120 (2023).
[39] A. Albaugh, R.-S. Fu, G. Gu, and T. R. Gingrich, "Limits on the precision of catenane molecular motors: Insights from thermodynamics and molecular dynamics simulations," Journal of Chemical Theory and Computation **20**, 1–6 (2024).
[40] E. Penocchio, G. Gu, A. Albaugh, and T. R. Gingrich, "Power strokes in molecular motors: Predictive, irrelevant, or somewhere in between?" Journal of the American Chemical Society **147**, 1063–1073 (2025).
[41] G. Gu, D. Alvarez, J. Strahan, A. Albaugh, E. Penocchio, and T. R. Gingrich, "It takes two to make a thing go right: Boosting current in coupled motors," (2026), arXiv:2601.09907.
[42] M. Athènes and G. Adjanor, "Measurement of nonequilibrium entropy from space-time thermodynamic integration," The Journal of Chemical Physics **129**, 024116 (2008).
[43] B. Leimkuhler and C. Matthews, *Molecular Dynamics: With Deterministic and Stochastic Numerical Methods* (Springer, 2015).
[44] R. D. Astumian, "Kinetic asymmetry allows macromolecular catalysts to drive an information ratchet," Nature Communications **10**, 3837 (2019).