Optimizing probability of barrier crossing with differentiable simulators

Martin Šípka [1 2 3] **Johannes C. B. Dietschreit** [1] **Michal Pavelka** [2] **Lukáš Grajciar** [3] **Rafael Gómez-Bombarelli** [1]

Abstract

Simulating events that involve some energy barrier often requires us to promote the barrier crossing in order to increase the probability of the event. One example of such a system can be a chemical reaction which we propose to explore using differentiable simulations. Transition path discovery and estimation of the reaction barrier are merged into a single end-to-end problem that is solved by path-integral optimization. We show how the probability of transition can be formulated in a differentiable way and increase it by introducing a trainable position-dependent bias function. We also introduce improvements over standard methods making DiffSim training stable and efficient.

1. Introduction

A chemical reaction is a transition from one depression (reactant, R) to another (product, P) on the potential energy surface (PES), this includes but is not limited to the rearrangement of chemical bonds, the exchange between conformers, and phase transitions. The reaction mechanism is defined by the most likely paths connecting the two states. Determining the reaction path is challenging due to the high dimensionality of molecular configuration space, which can have thousands of degrees of freedom (DoF). Standard unbiased sampling algorithms like molecular dynamics (MD) or Monte-Carlo (MC) often get trapped in stable regions, making it inefficient to explore candidate paths that have to cross low probability regions in phase space (Chipot & Pohorille, 2007; Chipot, 2014). Hence, it is necessary to bias the exploration.

[1]Department of Materials Science and Engineering, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, USA [2]Mathematical Institute, Faculty of Mathematics and Physics, Charles University, Sokolovská 83, 186 75 Prague, Czech Republic [3]Department of Physical and Macromolecular 853 Chemistry, Faculty of Sciences, Charles University, 128 43 854 Prague 2, Czech Republic. Correspondence to: Rafael Gómez-Bombarelli <rafagb@mit.edu>.

Published at the Differentiable Almost Everything Workshop of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. July 2023.

Finding reaction mechanisms is commonly divided into two sub-tasks. First, a dimensionality reduction from all DoFs down to the so-called collective variables (CVs) and second, enhanced sampling along those CVs (Torrie et al., 1977; Darve & Pohorille, 2001; Laio & Parrinello, 2002; Abrams & Bussi, 2013; Spiwok et al., 2015; Valsson et al., 2016). Recently, machine learning has partially automatized the task of CV identification. (Sultan & Pande, 2018; Mendels et al., 2018; Wehmeyer & Noé, 2018; Wang et al., 2019; Bonati et al., 2020; Wang & Tiwary, 2021; Sun et al., 2022; Šípka et al., 2022). The ideal CVs represent those DoFs, which fully describe the rare transition event, and thus are associated with the slowest motions. However, identification of the ideal CVs requires knowledge of the transition path that one is trying to discover, i.e., one still ends up with the proverbial "chicken-and-egg problem" (Rohrdanz et al., 2013). A problem not solved by previously proposed machine learning based tools. Furthermore, once the CVs are chosen it is very difficult to correct them on-the-fly. Once equipped with a good, low-dimensional representation of the chemical reaction, enhanced sampling algorithms usually introduce a biasing potential such that the occurrence of reactive events is increased significantly. Subsequent analysis can yield the reaction mechanism, reaction barrier, and rate constant.

Differentiable Simulations (DiffSims) have been developed for optimization, control, and learning of motion (Degrave et al., 2016; de Avila Belbute-Peres et al., 2018; Hu et al., 2019; 2020), but also for quantities of interest in molecular dynamics (Wang et al., 2020; Ingraham et al., 2019; Greener & Jones, 2021). Differentiating along trajectories follows naturally from optimizing path-dependent quantities. DiffSim results are often promising, but it is well known (Metz et al., 2021) that naïvely backpropagated gradients may vanish or explode. This problem is associated with the spectrum of the system's Jacobian (Metz et al., 2021; Galimberti et al., 2021). Therefore, in order to use DiffSim successfully, gradients have to become controllable. Here, the loss gradient behaviour is thoroughly studied, and mechanisms to control fluctuations and magnitude are proposed. Employing the improved DiffSim, we define a differentiable loss function that, when minimized, results in the robust training of a biasing potential, which enhances the sampling of reactive transitions without prior determination of CVs.

2. Problem Definition

Reaction processes on an atomistic level are explored using molecular dynamics. Let $U_0(\boldsymbol{x})$ be the system's PES, where the column vector $\boldsymbol{x} \in \mathbb{R}^N$ denotes the mass-weighted coordinates of the system and $\boldsymbol{p}$ the conjugate momenta. The PES is modified by a learnable bias term $B(\boldsymbol{x}, \theta)$ to increase the probability of the barrier crossing

$$U(\boldsymbol{x}, \theta) = U_0(\boldsymbol{x}) + B(\boldsymbol{x}, \theta). \tag{1}$$

Commonly, the system is coupled to a thermostat to generate trajectories from the canonical (NVT) ensemble (Callen & Scott, 1998). Here, we choose the Langevin thermostat because of its simplicity and its favorable properties with respect to differentiating along the computational graph, as will be shown later (see Section 3.2). The thermostat is coupled to the system through the friction constant γ (2). Hence, the biased dynamics evolve according to

$$\begin{aligned} \dot{\boldsymbol{x}}(t) &= \boldsymbol{p}(t) \\ \dot{\boldsymbol{p}}(t) &= -\frac{\partial U(\boldsymbol{x}(t))}{\partial \boldsymbol{x}} - \gamma \boldsymbol{p}(t) + \sqrt{2\gamma k_B T}\boldsymbol{R}(t), \end{aligned} \tag{2}$$

where k_b is the Boltzmann constant, T the absolute temperature of the bath, and $\boldsymbol{R}(t)$ a Gaussian process.

2.1. Differentiable barrier crossing

It is often suitable to use general curvilinear coordinate to describe reactions. These special coordinates are denoted with $\boldsymbol{\xi}(\boldsymbol{x}) \in \mathbb{R}^M$ and $M \leq N$. The wells W_α of reactant (-1) and product (1), divided by a reaction barrier, are characterized by the set of points Γ_α ($\alpha = -1, 1$), corresponding to equilibrium configurations in each well, i.e., an unbiased simulation on the PES $U_0(\boldsymbol{x})$ is likely to stay in these wells. The indicator function for a well is

$$\mathbb{1}_\alpha(\boldsymbol{x}) = \begin{cases} 1 & \text{for} \quad \boldsymbol{x} \in W_\alpha \\ 0 & \text{for} \quad \boldsymbol{x} \notin W_\alpha. \end{cases} \tag{3}$$

Under suitable regularity conditions, we can cast the escape probability p_α of a trajectory $\boldsymbol{X}$ with $\boldsymbol{x}(t_0) \in W_{-\alpha}$ as

$$p_\alpha = P\left(\sup_{t<t_e} (\mathbb{1}_\alpha(\boldsymbol{x}(t)) > 0 \;\middle|\; \boldsymbol{x}(t_0) \in W_{-\alpha} \right). \tag{4}$$

A soft loss function that is continuous everywhere and differentiable is defined as

$$L = L_{\boldsymbol{\xi}_\alpha} = \begin{cases} 0 & \text{if } \exists\, \boldsymbol{x}(t) \in W_\alpha \\ \min\limits_{t_0<t<t_e} (\boldsymbol{\xi}(\boldsymbol{x}(t)) - \boldsymbol{\xi}_\alpha)^2 & \text{otherwise} \end{cases}, \tag{5}$$

where for each trajectory a random single $\boldsymbol{\xi}_\alpha \in \Gamma_\alpha$ is selected for the loss function. The term $(\boldsymbol{\xi}(\boldsymbol{x}(t)) - \boldsymbol{\xi}_\alpha)^2$ is the distance metric measuring how close a trajectory got to the target. Other more complex metrics can be chosen. Minimizing this loss function leads to a maximization of the probability (4) and can be seen as the minimization of a path-dependent integral. Note that the loss function is defined only for one point of the trajectory and is influenced by the dynamics of every preceding point. In this manuscript, we only consider transitions between two wells, W_{-1} and W_1. Additional basins would be handled analogously.

2.2. Differentiable simulations

The loss in (5) is minimized by optimizing the parameters θ of the bias potential (1). Instead of saving the computational graph for the entire forward path, as it would be extremely memory-demanding, the optimization process can be conveniently reformulated using the adjoint equation. With the adjoint vectors the system dynamics can be run backwards, leading to memory-savings and the ability to adjust the extent of backpropagation based on the sought-after dynamical scale (see Section 3). We employ the framework and notation adapted recently for neural networks (Chen et al., 2018) from the original work by (Lev Semenovich Pontryagin et al., 1962).

Propagating $\boldsymbol{z}(t) = (\boldsymbol{x}(t), \boldsymbol{p}(t))$ using $f(\boldsymbol{z}(t), \theta) = \dot{\boldsymbol{z}}(t)$ (the right side of (2)) is called the *forward process*. Notice that $f(\boldsymbol{z}(t))$ is only time-dependent through $\boldsymbol{z}(t)$. The adjoint vectors are

$$\boldsymbol{a}(t) = \frac{\partial L}{\partial \boldsymbol{z}(t)}\,. \tag{6}$$

To solve (6) we introduce the new time $\tau \in (0, \tau_e)$ such that $\boldsymbol{z}(\tau = 0) = \boldsymbol{z}(t = t_e)$ and $\boldsymbol{z}(\tau = \tau_e) = \boldsymbol{z}(t = 0)$. This backward flowing time reflects that the loss is only influenced by preceding points in time. In backward flowing time τ the adjoint vectors obey the equations

$$\begin{aligned} \boldsymbol{a}(\tau = 0) &= \frac{\partial L}{\partial \boldsymbol{z}(\tau = 0)} \\ \dot{\boldsymbol{a}}(\tau) &= \boldsymbol{a}(\tau)^T \frac{\partial f(\boldsymbol{z}(\tau), \theta)}{\partial \boldsymbol{z}}. \end{aligned} \tag{7}$$

The total gradient of the loss function with respect to bias parameters is then obtained by

$$\frac{\partial L}{\partial \theta} = \int_0^{\tau_e} \boldsymbol{a}(\tau)^T \frac{\partial f(\boldsymbol{z}(\tau), \theta)}{\partial \theta}\, \mathrm{d}\tau\,. \tag{8}$$

While solving (7), $\boldsymbol{z}(\tau)$ can be either saved or reconstructed by running dynamics (2) backward, depending on the memory and computational trade-off. The algorithm for running the adjoint method for the dynamics that includes random noise is developed and analyzed in (Li et al., 2020).

3. Challenges and Solutions

3.1. Challenges

Ideally, one would simulate the biased dynamics (2), compute the loss (5), backpropagate by solving (7), and after a number of training epochs obtain the biasing potential that enhances transitions. However, currently the use of DiffSim is hampered by the following issues:

1. **Gradient control** Past efforts have been devoted to understand the behavior of gradients (their explosion or vanishing) that arise while optimizing DiffSims (Suh et al., 2022; Huang et al., 2021; Metz et al., 2021). A DiffSim can be arbitrarily deep, however, it can be challenging to controllably backpropagate complex Hamiltonians. In fact, even simple Hamiltonians can yield exploding gradients (Galimberti et al., 2021).
2. **Multiscale Problem** The dynamics on $U_0(\boldsymbol{x})$ can include very high and very low frequency motions. However, only the slow dynamics should be controlled by the trainable bias, as those are associated with the rare event. Avoiding fitting fast fluctuations reduces the noise in the gradients used for training.
3. **Chaotic behaviour** Small changes in initial conditions of some Hamiltonian systems can result in exponentially different trajectories (Percival I, 1987).
4. **One large parameter update per trajectory** DiffSims produce one update per trajectory. Obtaining a sufficient number of gradient updates is expensive when long trajectories are required.

We address all these challenges and show how to efficiently and controllably learn slow dynamics necessary for the investigation of reactions.

3.2. Partial backpropagation

Stopping the flow of the gradient through $\boldsymbol{x}$ with the *.detach()* operator introduced, e.g., in Refs. (Foerster et al., 2018; Schulman et al., 2015; Zhang et al., 2019) reduces the complexity and level of detail in the equations, the computational graph is pruned such that backpropagation occurs only in the momenta. The mathematical idea behind the use of *.detach()* operator is better explained in (Šípka et al., 2023). We demonstrate how the approach suppresses high order oscillation in Figure 1.

The use of the *.detach()* operator simplifies the adjoint time derivative to

$$\begin{aligned}\dot{\boldsymbol{a}}(\tau) &= \boldsymbol{a}^T \frac{\partial}{\partial \boldsymbol{p}(\tau)} \left(-\frac{\partial U(\boldsymbol{x}(\tau))}{\partial \boldsymbol{x}} - \gamma \boldsymbol{p}(\tau) + \sqrt{2\gamma k_B T}\boldsymbol{R}(\tau) \right) \\ &= -\boldsymbol{a}^T(\tau) \frac{\partial^2 U(\boldsymbol{x}(\tau))}{\partial^2 \boldsymbol{x}} \Delta\tau - \gamma \boldsymbol{a}(\tau) \,. \end{aligned} \tag{9}$$

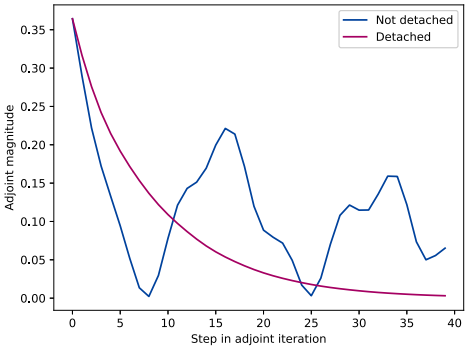

Figure 1. Comparison of the adjoint evolution for original and partially detached graphs for simulations on the 2D Müller-Brown potential with parameters described in (Šípka et al., 2023).

Additionally, the *.detach()* operator allows for theorems describing the adjoint magnitude and its controllability by the friction constant γ. Consider a trajectory x_t generated by (2) with time t in a possibly infinite time interval $I \subset (-\infty, \infty)$. The loss (5) is defined for a point $\boldsymbol{x}_{t_L}$. To optimize $B(\boldsymbol{x}, \theta)$, we need to backpropagate the gradient of this loss through every point preceding $\boldsymbol{x}_{t_L}$, using backwards flowing time τ.

Definition 3.1 (Differentiable Trajectory)**.** A Differentiable Trajectory $\mathcal{T}$ is defined by the following quadruple $(\boldsymbol{z}(t), L(\boldsymbol{z}(t_L)), f(\boldsymbol{z}(t)), \tilde{f}(\boldsymbol{z}(t)))$: Let $\boldsymbol{z}(t) \in \Omega_{\boldsymbol{x}} \times \Omega_{\boldsymbol{p}}$ where $\Omega_{\boldsymbol{x}} \subset \mathbb{R}^N$ and $\Omega_{\boldsymbol{p}} \subset \mathbb{R}^N$ for $t \in (t_i, t_e)$, where $-\infty \leq t_i < t_e \leq \infty$ be the sequence of states generate by the dynamics $f(\boldsymbol{z}(t))$ from a certain initial state $\boldsymbol{z}(t_0), t_0 \in [t_i, t_e]$. We define a loss function $L(\boldsymbol{z}(t_L))$ at time t_L. The gradient dynamics of the loss function is guided by the dynamics $\tilde{f}(\boldsymbol{z}(t))$ that includes possible $.detach()$ operators. The backward dynamics is represented in the reverse flowing time τ starting from $\boldsymbol{z}(t_L) = \boldsymbol{z}(\tau = 0)$ to $\boldsymbol{z}(\tau_e) = \boldsymbol{z}(t_i)$.

This allows us to formulate a theorem, that is proven in (Šípka et al., 2023)

Theorem 3.2 (Converging adjoints)**.** *$\mathcal{T}_d$ be a Differentiable Trajectory for the molecular dynamics presented in the paper. Let $U(\boldsymbol{x}) \in C^2(\Omega_{\boldsymbol{x}})$ and denote the spectrum of its Hessian $\frac{\partial^2 U(\boldsymbol{x}(t))}{\partial^2 \boldsymbol{x}}$ by $\lambda_i(\boldsymbol{x}(t))$. Define λ_{min} as*

$$\lambda_{min} = \inf_{\tau > 0} \min_i \lambda_i(\boldsymbol{x}(\tau)). \tag{10}$$

Then for every γ that fulfills: $(\Delta\tau\lambda_{min} + \gamma) = \epsilon > 0$, it holds:

$$\forall \tau > 0 : \|\boldsymbol{a}(\tau)\|^2 \leq \|\boldsymbol{a}(0)\|^2 e^{-2\epsilon\tau} \tag{11}$$

This theorem allows us to backpropagate the dynamics without exploding gradients as long as γ is sufficiently large. The exponential scaling of the adjoints also indicates that after identifying the point for which the loss is calculated, only a few points need to be considered before $\boldsymbol{a}(\tau)$ van-

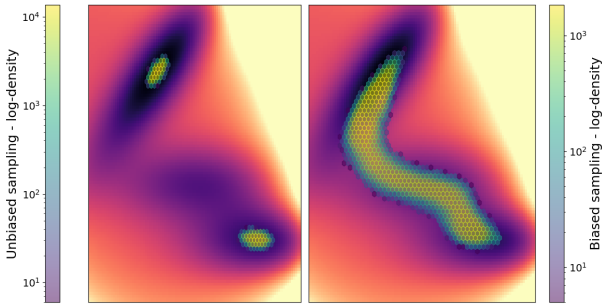

Figure 2. Log-density of simulated points before (*left*) and after the training (*right*) of bias function by differentiable simulations. The right plot shows how well all important regions are sampled after training. The background of the Figure is the $U_{\mathrm{MB}}(x, y)$, the underlying Muller-Brown potential.

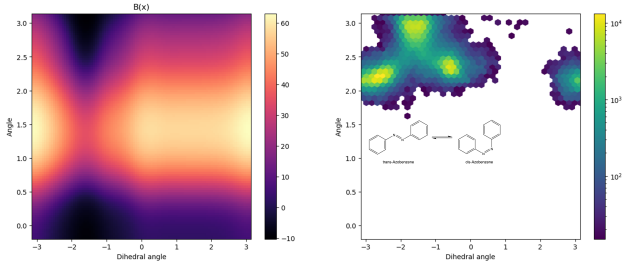

Figure 3. left Bias function (*right*) log-density obtained from long biased simulations.

ishes. Further adjoint propagation does not contribute to the gradient update.

The theorem also highlights when backpropagation may lead to exploding gradients. If the expression $(\Delta\tau\lambda_{min} + \gamma) = \epsilon < 0$, then the upper bound may not hold, and gradients can increase exponentially. A strongly negative λ_i of the Hessian indicates a concave part in the potential landscape, generally problematic for control. However, with $\Delta\tau$ and γ, we have two robust dials to ensure non-exploding adjoints.

3.3. Mini-batching the graph

To alleviate challenge 4 we introduce a technique called *graph mini-batching*. The idea is to calculate trajectory depended gradients first (the adjoints $\boldsymbol{a}$) in one pass and then split them to mini-batches. The adjoints are then used during training as vectors in Jacobi-vector products (8). The approach stabilizes learning and allows for much lower learning rates, better suited for training neural networks. An example of a use case is more thoroughly discussed in the work (Šípka et al., 2023).

4. Applications

4.1. 2D Muller-Brown potential

Details of the Muller-Brown PES (Müller & Brown, 1979) and the biasing function are given in (Šípka et al., 2023). This PES is difficult as no linear combination of the Cartesian coordinates yields a good CV. After training the bias potential via DiffSim, the biased dynamics generate increasingly many successful transitions between reactants and products along the expected transition path. Comparing right and left side of Figure 2, the unbiased dynamics was trapped in the local minima whereas the biased dynamics produces an almost even log-density along the expected transition path.

4.2. Cis-trans isomerization of azobenzene

Here, we study the cis-trans isomerization of azobenzene in the electronic ground state (the reaction is shown in the right side of Figure 3). A trained neural network potentials was obtained from Ref. (Axelrod et al., 2022) and is used for U_0. For this system we chose two descriptors, the CNNC dihedral angle and one NNC bond angle. These two angles describe rotation and inversion mechanisms. See (Axelrod et al., 2022) for more details about different possible mechanisms.

As the isomerization barrier is relatively high and reactant and product geometries are comparatively stiff, the reaction investigation is split into two steps. First, we simulate at 300 K with a relatively high learning rate to get an estimate of the reaction barrier. Then, the bias is refined at 1000 K with a much lower learning rate (see left part of Figure 3). The log-density (Figure 3, right side) obtained from simulations with the fully trained bias clearly shows a rotation assisted inversion mechanism. This is analogous to the findings of (Axelrod et al., 2022).

5. Conclusion

We present a way for DiffSims to increase the probability of a barrier crossing by learning a biasing potential. The effectiveness of the framework is demonstrated on a common toy system used for testing enhanced sampling algorithms and on a challenging cis-trans isomerization. It is shown that a neural network potential to can provide forces for a differentiable simulator and create a setup suitable for accelerators as dozens of replicas are simulated at the same time, significantly speeding up the algorithm. In the theoretical section of the manuscript, the gradient behavior in a differentiable simulator is discussed and their convergence in a so called *detached* and *undetached* regime is investigated.

References

Abrams, C. and Bussi, G. Enhanced Sampling in Molecular Dynamics Using Metadynamics, Replica-Exchange, and Temperature-Acceleration. *Entropy 2014, Vol. 16, Pages 163-199*, 16(1):163–199, 12 2013. ISSN 1099-4300. doi: 10.3390/E16010163. URL https://www.mdpi.com/1099-4300/16/1/163/htmhttps://www.mdpi.com/1099-4300/16/1/163.

Axelrod, S., Shakhnovich, E., and Gómez-Bombarelli, R. Excited state non-adiabatic dynamics of large photoswitchable molecules using a chemically transferable machine learning potential. *Nature Communications*, 13(1):3440, 6 2022. ISSN 2041-1723. doi: 10.1038/s41467-022-30999-w.

Bonati, L., Rizzi, V., and Parrinello, M. Data-Driven Collective Variables for Enhanced Sampling. *The journal of physical chemistry letters*, 11(8):2998–3004, 4 2020. ISSN 1948-7185. doi: 10.1021/ACS.JPCLETT.0C00535. URL https://pubmed.ncbi.nlm.nih.gov/32239945/.

Callen, H. B. and Scott, H. L. *Thermodynamics and an Introduction to Thermostatistics, 2nd ed*, volume 66. 1998. doi: 10.1119/1.19071.

Chen, R. T., Rubanova, Y., Bettencourt, J., and Duvenaud, D. Neural ordinary differential equations. In *Advances in Neural Information Processing Systems*, volume 2018-December, 2018.

Chipot, C. Frontiers in free-energy calculations of biological systems. *Wiley Interdisciplinary Reviews: Computational Molecular Science*, 4(1):71–89, 1 2014. ISSN 1759-0884. doi: 10.1002/WCMS.1157. URL https://onlinelibrary.wiley.com/doi/full/10.1002/wcms.1157https://onlinelibrary.wiley.com/doi/abs/10.1002/wcms.1157https://wires.onlinelibrary.wiley.com/doi/10.1002/wcms.1157.

Chipot, C. and Pohorille, A. *Free Energy Calculations*, volume 86 of *Springer Series in CHEMICAL PHYSICS*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2007. ISBN 978-3-540-38447-2. doi: 10.1007/978-3-540-38448-9. URL http://link.springer.com/10.1007/978-3-540-38448-9.

Darve, E. and Pohorille, A. Calculating free energies using average force. 2001. doi: 10.1063/1.1410978. URL http://jcp.aip.org/jcp/copyright.jsp.

de Avila Belbute-Peres, F., Smith, K., Allen, K., Tenenbaum, J., and Kolter, J. Z. End-to-End Differentiable Physics for Learning and Control. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper/2018/file/842424a1d0595b76ec4fa03c46e8d755-Paper.pdf.

Degrave, J., Hermans, M., Dambre, J., and wyffels, F. A Differentiable Physics Engine for Deep Learning in Robotics. *arXiv*, (1611.01652), 11 2016.

Foerster, J., Farquhar, G., Al-Shedivat, M., Rocktäschel, T., Xing, E. P., and Whiteson, S. DiCE: The Infinitely Differentiable Monte-Carlo Estimator. 2 2018.

Galimberti, C. L., Furieri, L., Xu, L., and Ferrari-Trecate, G. Hamiltonian Deep Neural Networks Guaranteeing Non-vanishing Gradients by Design. *arXiv*, (2105.13205), 5 2021.

Greener, J. G. and Jones, D. T. Differentiable molecular simulation can learn all the parameters in a coarse-grained force field for proteins. *PLOS ONE*, 16(9):e0256990, 9 2021. ISSN 1932-6203. doi: 10.1371/journal.pone.0256990.

Hu, Y., Li, T.-M., Anderson, L., Ragan-Kelley, J., and Durand, F. Taichi. *ACM Transactions on Graphics*, 38(6):1–16, 12 2019. ISSN 0730-0301. doi: 10.1145/3355089.3356506.

Hu, Y., Anderson, L., Li, T.-M., Sun, Q., Carr, N., Ragan-Kelley, J., and Durand, F. DiffTaichi: Differentiable Programming for Physical Simulation. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=B1eB5xSFvr.

Huang, Z., Hu, Y., Du, T., Zhou, S., Su, H., Tenenbaum, J. B., and Gan, C. PlasticineLab: A Soft-Body Manipulation Benchmark with Differentiable Physics. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=xCcdBRQEDW.

Ingraham, J., Riesselman, A., Sander, C., and Marks, D. Learning Protein Structure with a Differentiable Simulator. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Byg3y3C9Km.

Laio, A. and Parrinello, M. Escaping free-energy minima. *Proceedings of the National Academy of Sciences of the United States of America*, 99(20):12562–12566, 10 2002. ISSN

00278424. doi: 10.1073/PNAS.202427399/ASSET/2381B9FD-FD4C-4ED9-9DCB-75FD02E8EA7E/ASSETS/GRAPHIC/PQ2024273003.JPEG. URL https://www.pnas.org/doi/abs/10.1073/pnas.202427399.

Lev Semenovich Pontryagin, MishGamkrelidze RV, Boltyanskii VG, and Gamkrelidze RV. *The Mathematical Theory of Optimal Processes*. 1962.

Li, X., Wong, T.-K. L., Chen, R. T. Q., and Duvenaud, D. K. Scalable Gradients and Variational Inference for Stochastic Differential Equations. In *Proceedings of The 2nd Symposium on Advances in Approximate Bayesian Inference*, volume 118 of *Proceedings of Machine Learning Research*, pp. 1–28. PMLR, 9 2020. URL https://proceedings.mlr.press/v118/li20a.html.

Mendels, D., Piccini, G., and Parrinello, M. Collective Variables from Local Fluctuations. *Journal of Physical Chemistry Letters*, 9(11):2776–2781, 6 2018. ISSN 19487185. doi: 10.1021/ACS.JPCLETT.8B00733/ASSET/IMAGES/LARGE/JZ-2018-00733T{_}0005.JPEG. URL https://pubs.acs.org/doi/full/10.1021/acs.jpclett.8b00733.

Metz, L., Freeman, C. D., Schoenholz, S. S., and Kachman, T. Gradients are Not All You Need. *arXiv*, (2111.05803), 11 2021.

Müller, K. and Brown, L. D. Location of saddle points and minimum energy paths by a constrained simplex optimization procedure. *Theoretica chimica acta*, 53(1):75–93, 1979. ISSN 1432-2234. doi: 10.1007/BF00547608. URL https://doi.org/10.1007/BF00547608.

Percival I. Chaos in hamiltonian systems. *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences*, 413(1844):131–143, 9 1987. ISSN 0080-4630. doi: 10.1098/rspa.1987.0105.

Rohrdanz, M. A., Zheng, W., and Clementi, C. Discovering Mountain Passes via Torchlight: Methods for the Definition of Reaction Coordinates and Pathways in Complex Macromolecular Reactions. *Annual Review of Physical Chemistry*, 64(1):295–316, 4 2013. ISSN 0066-426X. doi: 10.1146/annurev-physchem-040412-110006.

Schulman, J., Heess, N., Weber, T., and Abbeel, P. Gradient Estimation Using Stochastic Computation Graphs. *arXiv*, (1506.05254), 6 2015.

Šípka, M., Erlebach, A., and Grajciar, L. Understanding chemical reactions via variational autoencoder and atomic representations. *arXiv*, (2202.00817), 3 2022.

Šípka, M., Dietschreit, J. C. B., Grajciar, L., and Gómez-Bombarelli, R. Differentiable Simulations for Enhanced Sampling of Rare Events. 1 2023.

Spiwok, V., Sucur, Z., and Hosek, P. Enhanced sampling techniques in biomolecular simulations. *Biotechnology advances*, 33(6 Pt 2):1130–1140, 2015. ISSN 1873-1899. doi: 10.1016/J.BIOTECHADV.2014.11.011. URL https://pubmed.ncbi.nlm.nih.gov/25482668/.

Suh, H. J. T., Simchowitz, M., Zhang, K., and Tedrake, R. Do Differentiable Simulators Give Better Policy Gradients? 2 2022.

Sultan, M. M. and Pande, V. S. Automated design of collective variables using supervised machine learning. *The Journal of Chemical Physics*, 149(9):94106, 2018. doi: 10.1063/1.5029972. URL https://doi.org/10.1063/1.5029972.

Sun, L., Vandermause, J., Batzner, S., Xie, Y., Clark, D., Chen, W., and Kozinsky, B. Multitask Machine Learning of Collective Variables for Enhanced Sampling of Rare Events. *Journal of Chemical Theory and Computation*, 18(4):2341–2353, 2022. doi: 10.1021/acs.jctc.1c00143. URL https://doi.org/10.1021/acs.jctc.1c00143.

Torrie, G. M., Valleau, J. P., Torrie, G. M., and Valleau, J. P. Nonphysical Sampling Distributions in Monte Carlo Free-Energy Estimation: Umbrella Sampling. *JCoPh*, 23(2):187–199, 1977. ISSN 0021-9991. doi: 10.1016/0021-9991(77)90121-8. URL https://ui.adsabs.harvard.edu/abs/1977JCoPh..23..187T/abstract.

Valsson, O., Tiwary, P., and Parrinello, M. Enhancing Important Fluctuations: Rare Events and Metadynamics from a Conceptual Viewpoint. *http://dx.doi.org/10.1146/annurev-physchem-040215-112229*, 67:159–184, 5 2016. ISSN 0066426X. doi: 10.1146/ANNUREV-PHYSCHEM-040215-112229. URL https://www.annualreviews.org/doi/abs/10.1146/annurev-physchem-040215-112229.

Wang, D. and Tiwary, P. State predictive information bottleneck. *The Journal of Chemical Physics*, 154(13):134111, 4 2021. ISSN 0021-9606. doi: 10.1063/5.0038198. URL https://aip.scitation.org/doi/abs/10.1063/5.0038198.

Wang, W., Axelrod, S., and Gómez-Bombarelli, R. Differentiable Molecular Simulations for Control and Learning. *arXiv*, (2003.00868), 2 2020.

Wang, Y., Ribeiro, J. M. L., and Tiwary, P. Past–future information bottleneck for sampling molecular reaction coordinate simultaneously with thermodynamics and kinetics. *Nature Communications 2019 10:1*, 10

(1):1–8, 8 2019. ISSN 2041-1723. doi: 10.1038/s41467-019-11405-4. URL https://www.nature.com/articles/s41467-019-11405-4.

Wehmeyer, C. and Noé, F. Time-lagged autoencoders: Deep learning of slow collective variables for molecular kinetics. *The Journal of Chemical Physics*, 148 (24):241703, 2018. doi: 10.1063/1.5011399. URL https://doi.org/10.1063/1.5011399.

Zhang, S.-X., Wan, Z.-Q., and Yao, H. Automatic Differentiable Monte Carlo: Theory and Application. *arXiv*, (1911.09117), 11 2019.