

A semigroup method for high dimensional committor functions based on neural network

Haoya Li

LIHAOYA@STANFORD.EDU

Department of Mathematics, Stanford University, Stanford, CA 94305, USA

Yuehaw Khoo

YKHOO@STATISTICS.UCHICAGO.EDU

Department of Statistics and the College, The University of Chicago, Chicago, IL 60637, USA

Yinuo Ren

RENYINUO@PKU.EDU.CN

School of Mathematical Sciences, Peking University, Beijing, China

Lexing Ying

LEXING@STANFORD.EDU

Department of Mathematics and ICME, Stanford University, Stanford, CA 94305, USA

Editors: Joan Bruna, Jan S Hesthaven, Lenka Zdeborova

Abstract

This paper proposes a new method based on neural networks for computing the high-dimensional committor functions that satisfy Fokker-Planck equations. Instead of working with partial differential equations, the new method works with an integral formulation based on the semigroup of the differential operator. The variational form of the new formulation is then solved by parameterizing the committor function as a neural network. There are two major benefits of this new approach. First, stochastic gradient descent type algorithms can be applied in the training of the committor function without the need of computing any mixed second order derivatives. Moreover, unlike the previous methods that enforce the boundary conditions through penalty terms, the new method takes into account the boundary conditions automatically. Numerical results are provided to demonstrate the performance of the proposed method.

Keywords: Committor function, Fokker-Planck equation, neural network, transition path theory.

1. Introduction

Understanding rare transition events between two states is important for studying the behavior of stochastic systems in physics, chemistry, and biology. One important method to describe the transition events is the transition path theory (Vanden-Eijnden et al., 2010; E and Vanden-Eijnden, 2006; Lu and Nolen, 2015), and the central object in the transition path theory is the committor function. Assume that the transition between two states is governed by the overdamped Langevin process

$$d\mathbf{x}_t = -\nabla V(\mathbf{x}_t) dt + \sqrt{2\beta^{-1}} d\mathbf{w}_t, \quad (1)$$

where $\mathbf{x}_t \in \Omega \subset \mathbb{R}^d$ is the state of the system, $V : \mathbb{R}^d \rightarrow \mathbb{R}$ is a smooth potential function, $\beta = 1/T$ is the inverse of the temperature T , and \mathbf{w}_t is the standard d -dimensional Brownian motion. For two given simply connected domains A and B in Ω with smooth boundaries, the committor function $q(\mathbf{x})$ is defined as

$$q(\mathbf{x}) = \mathbb{P}(\tau_B < \tau_A \mid \mathbf{x}_0 = \mathbf{x}), \quad (2)$$

where τ_A and τ_B are the hitting times for the sets A and B , respectively. Many statistical properties of the transition are encoded in the committor function.

The committor function satisfies the Fokker-Planck equation (also known as the steady-state backward Kolmogorov equation)

$$(-1/\beta\Delta + \nabla V \cdot \nabla)q = 0 \text{ in } \Omega \setminus (A \cup B), \quad q(\mathbf{x})|_{\partial A} = 0, \quad q(\mathbf{x})|_{\partial B} = 1. \quad (3)$$

If the potential V is confining, i.e. $V(\mathbf{x}) \rightarrow \infty$ as $|\mathbf{x}| \rightarrow \infty$ and the partition function $Z_\beta = \int_{\mathbb{R}^d} \exp(-\beta V(\mathbf{x})) d\mathbf{x} < \infty$ for any $\beta > 0$, then

$$\rho(\mathbf{x}) = \frac{1}{Z_\beta} \exp(-\beta V(\mathbf{x})) \quad (4)$$

is the equilibrium distribution of the Langevin dynamics (1). Under this condition, the Langevin process (1) is ergodic (Pavliotis, 2014), which enables the use of Monte Carlo methods to sample from the equilibrium distribution. For the rest of the paper we will always assume that V is confining and ρ exists.

One major difficulty in solving (3) is the curse of dimensionality. Various methods have been proposed under the assumption that the transition from A to B is concentrated in a quasi-one dimensional tube or low dimensional manifold. For example, the finite temperature string method (E et al., 2005; Vanden-Eijnden and Venturoli, 2009) approximates isosurfaces of the committor function with hyperplanes normal to the most probable transition paths, and updates the transition paths together with the isocommittor-surfaces. Diffusion map (Coifman et al., 2008) solves for q on a set of points by applying point cloud discretization to the operator $L = (-1/\beta\Delta + \nabla V \cdot \nabla)$. In order to obtain better convergence order, (Lai and Lu, 2018) improves on diffusion map by discretizing L using a finite element method on local tangent planes of the point cloud.

More recently, the method proposed in (Khoo et al., 2019) works with the variational form of the Fokker-Planck equation

$$\operatorname{argmin}_q \int_{\Omega \setminus (A \cup B)} |\nabla q(\mathbf{x})|^2 \rho(\mathbf{x}) d\mathbf{x}, \quad q(\mathbf{x})|_{\partial A} = 0, \quad q(\mathbf{x})|_{\partial B} = 1, \quad (5)$$

and parameterizes the high-dimensional committor function $q(\cdot)$ by a neural network (NN) $q_\theta(\cdot)$. The main advantage of working with an optimization formulation is that under rather mild conditions stochastic gradient descent (SGD) type algorithms can avoid the saddle points and converge efficiently to at least local minimums, as shown for example in Jin et al. (2019). The boundary conditions can be enforced by additional penalty terms as in

$$\operatorname{argmin}_{\theta \in \mathbb{R}^p} \int_{\Omega \setminus (A \cup B)} |\nabla q_\theta(\mathbf{x})|^2 \rho(\mathbf{x}) d\mathbf{x} + \tilde{c} \int_{\partial A} q_\theta(\mathbf{x})^2 dm_{\partial A}(\mathbf{x}) + \tilde{c} \int_{\partial B} (q_\theta(\mathbf{x}) - 1)^2 dm_{\partial B}(\mathbf{x}), \quad (6)$$

where m_A and m_B are measures supported on ∂A and ∂B . The integral in (6) is approximated by a Monte-Carlo method where the samples are generated according to the stochastic process (1). Satisfactory numerical results are obtained with this method even in high dimensional cases where traditional methods like finite-element-type methods are intractable. However, because the loss depends on $|\nabla q_\theta|^2$, obtaining the gradient of the loss with respect to θ requires an inconvenient second order derivative computation.

The contribution of this paper is two-fold. First, we propose a new method that removes the dependence on the second order derivatives by working with an integral formulation based on the semigroup of (1). Second, the boundary conditions are treated automatically by the semigroup formulation rather than relying solely on the penalty terms, which makes it easier to tune the penalty coefficients. The paper is organized as follows. Section 2 describes the new formulation and the neural network approximation. Section 3 analyzes this new method under the so-called lazy-training regime. Finally, the performance of our method is examined numerically in Section 4.

2. Proposed method

2.1. A new variational formulation

Consider the Langevin process starting from a point $\mathbf{x} \in \Omega$

$$\begin{aligned} d\mathbf{x}_t &= -\nabla V(\mathbf{x}_t) dt + \sqrt{2\beta^{-1}} d\mathbf{w}_t, \\ \mathbf{x}_0 &= \mathbf{x}. \end{aligned} \quad (7)$$

Let τ_A and τ_B be the stopping time of the process hitting ∂A and ∂B , respectively. Similarly, define $\tau \equiv \tau_{A \cup B} = \min(\tau_A, \tau_B)$ be the hitting time of $A \cup B$. For a fixed small time step $\delta > 0$, we define the operator P as follows:

$$(Pf)(\mathbf{x}) := \mathbb{E}^{\mathbf{x}}(f(\mathbf{x}_{\tau \wedge \delta})), \quad (8)$$

where $\mathbb{E}^{\mathbf{x}}$ is the expectation taken with respect to the law of the process (7).

Proposition 1 *When ∇V is bounded and Lipschitz continuous on \mathbb{R}^d , the solution to the committor function (3) satisfies the following semigroup formulation*

$$q(\mathbf{x}) = (Pq)(\mathbf{x}) \quad \text{in } \Omega \setminus (A \cup B), \quad q|_{\partial A} = 0, \quad q|_{\partial B} = 1. \quad (9)$$

The proof of Proposition 1 is provided in Appendix A. The main advantage of working with (9) over (3) is that it does not contain any differential operator.

For notational convenience, we introduce a function $r : \partial A \cup \partial B \rightarrow \mathbb{R}$ with $r(\mathbf{x})|_{\partial A} = 0$ and $r(\mathbf{x})|_{\partial B} = 1$. With this definition, the boundary condition of (9) is simply $q = r$ on $\partial A \cup \partial B$. In order to introduce the variational formulation, Pq can be split into two parts as follows:

$$\begin{aligned} (Pq)(\mathbf{x}) &= \mathbb{E}^{\mathbf{x}}(q(\mathbf{x}_{\tau \wedge \delta})) = \mathbb{E}^{\mathbf{x}}(q(\mathbf{x}_{\tau \wedge \delta}) \mathbf{1}_{\{\delta < \tau\}}) + \mathbb{E}^{\mathbf{x}}(q(\mathbf{x}_{\tau \wedge \delta}) \mathbf{1}_{\{\delta \geq \tau\}}) \\ &= \mathbb{E}^{\mathbf{x}}(q(\mathbf{x}_{\delta}) \mathbf{1}_{\{\delta < \tau\}}) + \mathbb{E}^{\mathbf{x}}(q(\mathbf{x}_{\tau}) \mathbf{1}_{\{\delta \geq \tau\}}) \\ &= \mathbb{E}^{\mathbf{x}}(q(\mathbf{x}_{\delta}) \mathbf{1}_{\{\delta < \tau\}}) + \mathbb{E}^{\mathbf{x}}(r(\mathbf{x}_{\tau}) \mathbf{1}_{\{\delta \geq \tau\}}), \end{aligned} \quad (10)$$

where the last equality results from the fact that $\mathbf{x}_{\tau} \in \partial A \cup \partial B$ and $q = r$ on $\partial A \cup \partial B$.

We denote the first part of (10) as

$$(P^i q)(\mathbf{x}) \equiv \mathbb{E}^{\mathbf{x}}(q(\mathbf{x}_{\tau \wedge \delta}) \mathbf{1}_{\{\delta < \tau\}}) = \mathbb{E}^{\mathbf{x}}(q(\mathbf{x}_{\delta}) \mathbf{1}_{\{\delta < \tau\}}), \quad (11)$$

where the superscript i stands for the interior contribution and the second part of (10) as

$$(P^b r)(\mathbf{x}) \equiv \mathbb{E}^{\mathbf{x}}(r(\mathbf{x}_{\tau \wedge \delta}) \mathbf{1}_{\{\delta \geq \tau\}}) = \mathbb{E}^{\mathbf{x}}(r(\mathbf{x}_{\tau}) \mathbf{1}_{\{\delta \geq \tau\}}), \quad (12)$$

where the superscript b stands for the boundary contribution. With these definitions, we can rewrite (9) compactly as

$$(I - P^i)q(\mathbf{x}) - (P^b r)(\mathbf{x}) = 0, \quad (13)$$

where I is the identity operator. The following result states that P^i is symmetric on the Hilbert space $L^2_\rho(\Omega \setminus (A \cup B))$.

Proposition 2 *P^i is a symmetric operator on $L^2_\rho(\Omega \setminus (A \cup B))$, in other words, $\langle u, P^i v \rangle_\rho = \langle P^i u, v \rangle_\rho$, where $\langle \cdot, \cdot \rangle_\rho$ denotes the inner product of the Hilbert space $L^2_\rho(\Omega \setminus (A \cup B))$.*

For the proof of this proposition, see Appendix B.

Based on Proposition 2, we are now ready to propose the following variational formulation for (13)

$$\min_q \frac{1}{2} \int_{\Omega \setminus (A \cup B)} q(\mathbf{x}) ((I - P^i)q(\mathbf{x})) \rho(\mathbf{x}) d\mathbf{x} - \int_{\Omega \setminus (A \cup B)} q(\mathbf{x}) P^b r(\mathbf{x}) \rho(\mathbf{x}) d\mathbf{x} \quad (14)$$

The two formulations (14) and (13) share the same solution as shown below. Let q^* be the solution to the variational problem (14) and $q(\mathbf{x}, \epsilon) = q^*(\mathbf{x}) + \epsilon \eta(\mathbf{x})$, where η is continuous with compact support. By taking derivative with respect to ϵ at $\epsilon = 0$ we obtain

$$\begin{aligned} 0 &= \frac{\partial}{\partial \epsilon} \left(\frac{1}{2} \int_{\Omega \setminus (A \cup B)} q(\mathbf{x}, \epsilon) ((I - P^i)q(\mathbf{x}, \epsilon)) \rho(\mathbf{x}) d\mathbf{x} - \int_{\Omega \setminus (A \cup B)} q(\mathbf{x}, \epsilon) P^b r(\mathbf{x}) \rho(\mathbf{x}) d\mathbf{x} \right) \Big|_{\epsilon=0} \\ &= \int_{\Omega \setminus (A \cup B)} \eta(\mathbf{x}) ((I - P^i)q^*(\mathbf{x})) \rho(\mathbf{x}) d\mathbf{x} - \int_{\Omega \setminus (A \cup B)} \eta(\mathbf{x}) P^b r(\mathbf{x}) \rho(\mathbf{x}) d\mathbf{x} \\ &= \int_{\Omega \setminus (A \cup B)} \eta(\mathbf{x}) ((I - P^i)q^*(\mathbf{x}) - P^b r(\mathbf{x})) \rho(\mathbf{x}) d\mathbf{x}, \end{aligned} \quad (15)$$

where the second equality uses Proposition 2. Since this is true for any continuous η with compact support and $\rho(\mathbf{x}) > 0$, we conclude that $(I - P^i)q^*(\mathbf{x}) - P^b r(\mathbf{x}) = 0$ on $\Omega \setminus (A \cup B)$.

Plugging in the definitions of P^i and P^b into (14) leads to

$$\min_q \frac{1}{2} \int_{\Omega \setminus (A \cup B)} q(\mathbf{x}) (q(\mathbf{x}) - \mathbb{E}^{\mathbf{x}}(q(\mathbf{x}_\delta) \mathbf{1}_{\{\delta < \tau\}})) \rho(\mathbf{x}) d\mathbf{x} - \int_{\Omega \setminus (A \cup B)} q(\mathbf{x}) \mathbb{E}^{\mathbf{x}}(r(\mathbf{x}_\tau) \mathbf{1}_{\{\delta \geq \tau\}}) \rho(\mathbf{x}) d\mathbf{x}. \quad (16)$$

It is clear from this formulation that there is no need for taking gradient of $q(\mathbf{x})$ with respect to \mathbf{x} .

Although there is no need to enforce the boundary conditions $q|_{\partial A} = 0$ and $q|_{\partial B} = 1$ explicitly in (14), one can still include the penalty terms that can sometimes give a better performance

$$\begin{aligned} \min_q \frac{1}{2} \int_{\Omega \setminus (A \cup B)} q(\mathbf{x}) ((I - P^i)q(\mathbf{x})) \rho(\mathbf{x}) d\mathbf{x} - \int_{\Omega \setminus (A \cup B)} q(\mathbf{x}) P^b r(\mathbf{x}) \rho(\mathbf{x}) d\mathbf{x} \\ + \frac{c}{2} \int_{\partial A} q(\mathbf{x})^2 dm_A(\mathbf{x}) + \frac{c}{2} \int_{\partial B} (q(\mathbf{x}) - 1)^2 dm_B(\mathbf{x}), \end{aligned} \quad (17)$$

where m_A and m_B are measures supported on ∂A and ∂B , respectively, and $c > 0$ is a penalty constant.

2.2. Nonlinear parameterization

In order to deal with the high dimensionality of the committor function $q(\mathbf{x})$, we propose to approximate $q(\mathbf{x})$ with an NN $q_\theta(\mathbf{x})$, where θ stands for the NN parameters. In terms of θ , the optimization problem takes the form

$$\begin{aligned} \min_{\theta} \frac{1}{2} \int_{\Omega \setminus (A \cup B)} q_\theta(\mathbf{x}) ((I - P^i)q_\theta(\mathbf{x})) \rho(\mathbf{x}) d\mathbf{x} - \int_{\Omega \setminus (A \cup B)} q_\theta(\mathbf{x}) P^b r(\mathbf{x}) \rho(\mathbf{x}) d\mathbf{x} \\ + \frac{c}{2} \int q_\theta(\mathbf{x})^2 dm_A(\mathbf{x}) + \frac{c}{2} \int (q_\theta(\mathbf{x}) - 1)^2 dm_B(\mathbf{x}). \end{aligned} \quad (18)$$

When applying an stochastic gradient descent (SGD) to optimize (18), one needs to compute the derivative for each of the terms.

Derivative of the first two terms. By the symmetric property stated in Proposition 2, the derivative of the first two terms of (18) is

$$\begin{aligned} \int_{\Omega \setminus (A \cup B)} \nabla_\theta q_\theta(\mathbf{x}) ((I - P^i)q_\theta(\mathbf{x})) \rho(\mathbf{x}) d\mathbf{x} - \int_{\Omega \setminus (A \cup B)} \nabla_\theta q_\theta(\mathbf{x}) P^b r(\mathbf{x}) \rho(\mathbf{x}) d\mathbf{x} \\ = \int_{\Omega \setminus (A \cup B)} \nabla_\theta q_\theta(\mathbf{x}) (q_\theta(\mathbf{x}) - \mathbb{E}^\mathbf{x} (q_\theta(\mathbf{x}_\delta) \mathbf{1}_{\{\delta < \tau\}}) - \mathbb{E}^\mathbf{x} (r(\mathbf{x}_\tau) \mathbf{1}_{\{\delta \geq \tau\}})) \rho(\mathbf{x}) d\mathbf{x}. \end{aligned} \quad (19)$$

Notice that (19) is an integral with measure $\rho(\mathbf{x}) d\mathbf{x}$. Therefore, if $\mathbf{x} \sim \rho$ and \mathbf{x}_δ follows (7), (19) can be written as

$$\mathbb{E}_{\mathbf{x} \sim \rho} \nabla_\theta q_\theta(\mathbf{x}) (q_\theta(\mathbf{x}) - \mathbb{E}^\mathbf{x} (q_\theta(\mathbf{x}_\delta) \mathbf{1}_{\{\delta < \tau\}}) - \mathbb{E}^\mathbf{x} (r(\mathbf{x}_\tau) \mathbf{1}_{\{\delta \geq \tau\}})) . \quad (20)$$

An unbiased estimator for (19) in an SGD method is therefore

$$\nabla_\theta q_\theta(\mathbf{x}) (q_\theta(\mathbf{x}) - q_\theta(\mathbf{x}_\delta) \mathbf{1}_{\{\delta < \tau\}} - r(\mathbf{x}_\tau) \mathbf{1}_{\{\delta \geq \tau\}}) = \nabla_\theta q_\theta(\mathbf{x}) (q_\theta(\mathbf{x}) - q_\theta(\mathbf{x}_\delta) \mathbf{1}_{\{\delta < \tau\}} - \mathbf{1}_{\{\delta \geq \tau = \tau_B\}}) . \quad (21)$$

The significance of (21) is that one only needs to take the gradient once with respect to the parameter θ , whereas in (Khoo et al., 2019) one needs to take the gradient of both \mathbf{x} and θ .

Let us comment on the implementation details of (21). First, the sample \mathbf{x} is supposed to be sampled from ρ . As mentioned previously, if the potential function V is confining, then the Langevin process (1) is ergodic. This implies that the distribution of the samples \mathbf{x}_t generated from the stochastic differential equation (SDE) (7) converges to ρ in the limit of $t \rightarrow \infty$. Therefore, after an initial burn-in period, the samples from the SDE trajectory follow the distribution ρ . There has been a large literature on how to solve SDE (7) numerically. The simplest method is the Euler-Maruyama scheme (see for example (Kloeden and Platen, 2013)). Let $\Delta t > 0$ be the time step size and $\mathbf{w}_{\Delta t} \sim \mathcal{N}(0, \Delta t I_d)$, and I_d is the d -dimensional identity matrix. The Euler-Maruyama approximation $\tilde{\mathbf{x}}_{n\Delta t}$ at time steps $n\Delta t$ are computed via

$$\tilde{\mathbf{x}}_{(n+1)\Delta t} = \tilde{\mathbf{x}}_{n\Delta t} - \nabla V(\tilde{\mathbf{x}}_{n\Delta t}) \Delta t + \sqrt{2\beta^{-1}} \mathbf{w}_{\Delta t}.$$

Assuming ergodicity, the distribution $\mathbf{x} \sim \rho$ can be approximated by $\tilde{\mathbf{x}}_{N\Delta t}$ for a sufficiently small Δt and sufficiently large N with an arbitrary $\tilde{\mathbf{x}}_0$.

Second, given $\mathbf{x} \sim \rho$, one needs to sample \mathbf{x}_δ . We approximate \mathbf{x}_δ by Euler-Maruyama scheme as well

$$\mathbf{x}_\delta = \mathbf{x} - \nabla V(\mathbf{x})\delta + \sqrt{2\beta^{-1}}\mathbf{w}_\delta, \quad (22)$$

where $\mathbf{w}_\delta \sim \mathcal{N}(0, \delta I_d)$.

Finally, it is necessary to determine the indicators $\mathbf{1}_{\{\delta < \tau\}}$, $\mathbf{1}_{\{\delta \geq \tau = \tau_A\}}$ and $\mathbf{1}_{\{\delta \geq \tau = \tau_B\}}$ in order to compute (21). For this, the following approximations are used

$$\mathbf{1}_{\{\delta < \tau\}} = 1 \text{ if } \mathbf{x}_\delta \in \Omega \setminus A \cup B, \quad \mathbf{1}_{\{\delta \geq \tau = \tau_A\}} = 1 \text{ if } \mathbf{x}_\delta \in A, \quad \mathbf{1}_{\{\delta \geq \tau = \tau_B\}} = 1 \text{ if } \mathbf{x}_\delta \in B.$$

Intuitively these approximations should work well when δ is sufficiently small, which is also supported by the numerical experiments. Even for a fixed δ , we can apply Euler-Maruyama scheme with multiple steps to improve the accuracy. Specifically, given $\mathbf{x} \sim \rho$, one can approximate \mathbf{x}_δ by the last term of the sequence $\mathbf{x}_0, \mathbf{x}_{\delta/M}, \dots, \mathbf{x}_\delta$, where

$$\mathbf{x}_{(k+1)\delta/M} = \mathbf{x}_{k\delta/M} - \nabla V(\mathbf{x}_{k\delta/M})\delta/M + \sqrt{2\beta^{-1}}\mathbf{w}_{\delta/M}, \quad (23)$$

and $\mathbf{x}_0 = \mathbf{x}$. When using the multiple-step approximation, the indicators are determined by

$$\begin{aligned} \mathbf{1}_{\{\delta < \tau\}} &= 1 \text{ if } \{\mathbf{x}_{k\delta/M}\}_{k=1}^M \subset \Omega \setminus A \cup B, \\ \mathbf{1}_{\{\delta \geq \tau = \tau_A\}} &= 1 \text{ if } \{\mathbf{x}_{k\delta/M}\}_{k=1}^M \cap A \neq \emptyset, \\ \mathbf{1}_{\{\delta \geq \tau = \tau_B\}} &= 1 \text{ if } \{\mathbf{x}_{k\delta/M}\}_{k=1}^M \cap B \neq \emptyset. \end{aligned}$$

It is worth mentioning that various importance sampling methods (see for example (Li et al., 2019; Rotskoff and Vanden-Eijnden, 2020)) can potentially be used to approximate the integral in (20) and the corresponding gradients calculated above. One can generate the initial state \mathbf{x} in (22) using importance sampling techniques, and then generate \mathbf{x}_δ and also subsequent samples according to (22).

Derivative of the penalty terms. For the third and fourth terms of (18), unbiased estimators of their gradients are

$$c\nabla_{\theta} q_{\theta}(\mathbf{x}_A)q_{\theta}(\mathbf{x}_A), \quad c\nabla_{\theta} q_{\theta}(\mathbf{x}_B)(q_{\theta}(\mathbf{x}_B) - 1), \quad (24)$$

respectively, where $\mathbf{x}_A \sim m_A$ and $\mathbf{x}_B \sim m_B$.

Connection with reinforcement learning. Solving committor function can be viewed as a special case of the policy evaluation problem in reinforcement learning (RL): $\Omega \setminus (A \cup B)$ is the state space; the transition kernel is given by the operator P^i defined in (11); the discount factor is equal to one; there is no immediate reward for each individual step but the final reward is $(P^b g)(\mathbf{x})$; $q(\mathbf{x})$ is the value function of the problem; (13) is the Bellman equation; and (21) is the temporal difference (TD) update.

However, the key difference with the general policy evaluation problem is that, due to the detailed balancing of overdamped Langevin dynamics, there is a variational formulation (14) for (13), which is not available for general RL problems except the special case considered in (Ollivier, 2018). Because of the variational formulation, (21) is guaranteed to converge at least to a local minimum, even in the neural network parameterization. On the other hand, such a stability result fails to exist for neural network approximations for general RL problems, as many techniques and tricks are required in order for the neural network parameters to converge.

2.3. Neural network architecture

The architecture of the neural network $q_\theta(\mathbf{x})$ follows the one used in (Khoo et al., 2019) and is specifically designed for this problem. Below we briefly summarize the main design considerations.

In the low temperature regime, i.e. when $T \rightarrow 0$, there is typically a sharp interface between A and B , as pointed out by (Khoo et al., 2019). In order to account for this sharp transition, the tangent function \tanh is used as the activation function at the last nonlinear layer, as shown in Fig. 1.

In the high temperature regime, i.e. when $T \rightarrow \infty$, there is another type of singular behavior. As $T \rightarrow \infty$, the equation (3) converges heuristically to a Laplace equation with a Dirichlet boundary condition. When the domains A and B are relatively small, the solution near the boundary ∂A and ∂B are dictated asymptotically by the fundamental solution

$$\Phi(\mathbf{x}) := \begin{cases} -\frac{1}{2\pi} \log |\mathbf{x}| & (n = 2) \\ \frac{\Gamma(n/2)}{(2\pi)^{n/2} |\mathbf{x}|^{n-2}} & (n \geq 3) \end{cases}, \quad (25)$$

where n is the inherent dimension of the Laplace equation of $T \rightarrow \infty$, as demonstrated in (Khoo et al., 2019). For example, in the rugged-Mueller problem 4.2, the inherent dimension $n = 2$, while in the Ginzburg-Landau problem 4.3 we have $n = d = 49$.

In order to address these two types of singular behaviors, we introduce the following parameterization

$$q_\theta(\mathbf{x}) := n_{\theta_A}(\mathbf{x})S_A(\mathbf{x} - \mathbf{y}^A) + n_{\theta_B}(\mathbf{x})S_B(\mathbf{x} - \mathbf{y}^B) + n_{\theta_0}(\mathbf{x}), \quad (26)$$

where \mathbf{y}^A and \mathbf{y}^B are the centers of A and B , $S_A(\mathbf{x} - \mathbf{y})$ and $S_B(\mathbf{x} - \mathbf{y})$ are set to be fundamental solutions (25), with n depending on the inherent dimension of the problem. Finally, n_{θ_A} , n_{θ_B} are fully connected neural networks with ReLU activation, and n_{θ_0} is a fully connected neural network with \tanh activation at the last nonlinear layer and ReLU activation at other nonlinear layers. This architecture is summarized in Fig. 1.

3. Lazy training analysis of the optimization

When the learning rate approaches zero, the dynamics of the SGD can be approximated by the corresponding gradient flow (see (Kushner and Yin, 2003) for example):

$$\theta'(t) = -\nabla_\theta R(q_\theta), \quad \theta(0) = \theta_0, \quad (27)$$

where in our case the loss function $R(\cdot)$ takes the form

$$R(q) = \frac{1}{2} \langle (I - P^i)q - P^b r, q \rangle_\rho + \frac{c}{2} \langle q, q \rangle_{m_A} + \frac{c}{2} \langle q - 1, q - 1 \rangle_{m_B}, \quad (28)$$

with $\langle u, v \rangle_{m_A} = \int u v d m_A$ and $\langle u, v \rangle_{m_B} = \int u v d m_B$. In this section, we consider instead the rescaled gradient flow

$$\theta'(t) = -\frac{1}{\alpha^2} \nabla_\theta R(\alpha q_\theta), \quad \theta(0) = \theta_0, \quad (29)$$

with $R(\alpha q) = \frac{1}{2} \langle (I - P^i)(\alpha q) - P^b r, \alpha q \rangle_\rho + \frac{c}{2} \langle \alpha q, \alpha q \rangle_{m_A} + \frac{c}{2} \langle \alpha q - 1, \alpha q - 1 \rangle_{m_B}$, and analyze the training dynamics of q_θ in the lazy training regime. The reason for considering this rescaled formula is that the scaling effect caused by α arises in several situations, for example, when the

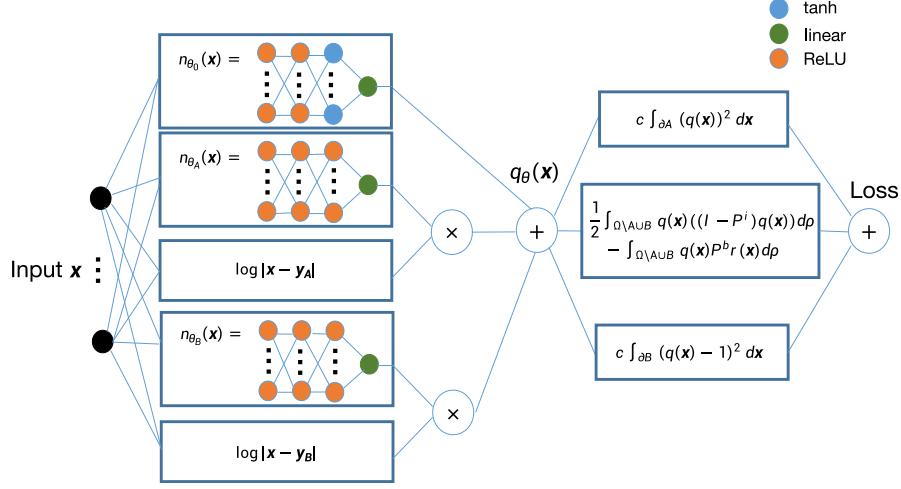


Figure 1: An example of the neural network architecture for a committor function and the corresponding loss. In this example we have two $\log |\mathbf{x} - \mathbf{y}|$ type singularities.

weights of the NN are large in magnitude at initialization and the learning rate is small (see for example, (Chizat et al., 2019; Agazzi and Lu, 2020)).

In (Chizat et al., 2019), it has been shown that when the scaling factor α is sufficiently large, the gradient flow (29) converges at a geometric rate to a local minimum of $F_{\alpha}(\theta) \equiv R(\alpha q_{\theta})/\alpha^2$, under some conditions that are detailed below.

Theorem 3 (Chizat et al., 2019) *Assume that: (1) $\theta \mapsto q_{\theta} \in \mathcal{F}$, where \mathcal{F} is a separable Hilbert space; (2) q is differentiable with a locally Lipschitz differential $\nabla_{\theta} q_{\theta}$; (3) $q_{\theta_0} = 0$ and $\text{rank } \nabla_{\theta} q_{\theta}$ is a constant in a neighborhood of θ_0 ; (4) R is strongly convex and differentiable with a Lipschitz gradient.*

Then there exists $\alpha_0 > 0$, such that for any $\alpha > \alpha_0$, the gradient flow (29) converges at a geometric rate to a local minimum of F_{α} .

In our setting, \mathcal{F} is the separable Hilbert space $L^2_{\nu}((\Omega \setminus (A \cup B)) \cup \partial A \cup \partial B)$, where $\nu = \rho + c m_A + c m_B$ and the rescaled gradient flow is given by

$$\theta'(t) = -\frac{1}{\alpha} \left[\langle (I - P^i) \alpha q_{\theta(t)} - P^b r, \nabla_{\theta} q_{\theta(t)} \rangle_{\rho} - c \langle \alpha q_{\theta(t)}, \nabla_{\theta} q_{\theta(t)} \rangle_{m_A} - c \langle \alpha q_{\theta(t)} - 1, \nabla_{\theta} q_{\theta(t)} \rangle_{m_B} \right]. \quad (30)$$

The following result states the strong convexity of R .

Proposition 4 *Assume that the operator P^i defined in (11) has a corresponding probability density function $p^i(\mathbf{x}, \mathbf{y})$ such that*

$$P^i f(\mathbf{x}) = \int_{\Omega \setminus (A \cup B)} p^i(\mathbf{x}, \mathbf{y}) f(\mathbf{y}) d\mathbf{y}, \quad (31)$$

and

$$\iint_{\Omega \setminus (A \cup B) \times \Omega \setminus (A \cup B)} p^i(\mathbf{x}, \mathbf{y}) p^i(\mathbf{y}, \mathbf{x}) d\mathbf{x} d\mathbf{y} < \infty. \quad (32)$$

Then the quadratic form $f \mapsto \langle f, (I - P^i)f \rangle_\rho$ is strongly convex on $L^2_\rho(\Omega \setminus A \cup B)$.

Based on Proposition 4, the following theorem is a direct consequence of Theorem 3.

Theorem 5 Assume that (31) and (32) hold, and $q_{\theta_0} = 0$, and that $\text{rank } \nabla_\theta q_\theta$ is a constant in a neighborhood of θ_0 , then there exists $\alpha_0 > 0$, such that for any $\alpha > \alpha_0$, the gradient flow (30) converges at a geometric rate to a local minimum of $F_\alpha(\theta) = R(\alpha q_\theta)/\alpha^2$.

4. Numerical experiments

This section presents the numerical results of the proposed method on several examples. The neural network architecture follows Fig. 1 and the stochastic gradient is computed following the description in Section 2.2. The training is carried out with the Adam optimizer (Kingma and Ba, 2014). Whenever the true solution q^* is available, the performance is evaluated by the relative error metric:

$$E = \frac{\|q_\theta - q^*\|_{L^2_\rho(\Omega \setminus A \cup B)}}{\|q^*\|_{L^2_\rho(\Omega \setminus A \cup B)}} \quad (33)$$

computed on a validation dataset generated according to the distribution ρ . In all the experiments, 2000 samples are used for the boundary measures m_A and m_B .

4.1. The double-well potential

In this experiment, consider the committor function in the following double-well potential:

$$V(\mathbf{x}) = (x_1^2 - 1)^2 + 0.3 \sum_{i=2}^d x_i^2, \quad (34)$$

with $d = 10$ and the regions A and B defined as

$$A = \{x \in \mathbb{R}^d \mid x_1 \leq -1\}, \quad B = \{x \in \mathbb{R}^d \mid x_1 \geq 1\}. \quad (35)$$

The true solution can be easily obtained by letting $q(\mathbf{x}) = \tilde{q}(x_1)$, such that it satisfies the one-dimensional ordinary differential equation (ODE)

$$\frac{d^2 \tilde{q}(x_1)}{dx_1^2} - 4\beta x_1(x_1^2 - 1) \frac{d\tilde{q}(x_1)}{dx_1} = 0, \quad \tilde{q}(0) = 0, \quad \tilde{q}(1) = 1. \quad (36)$$

By solving this ODE numerically, we obtain a highly accurate approximation of the exact solution q^* .

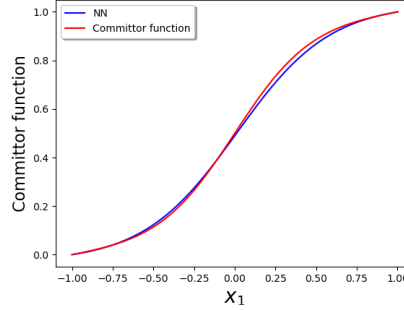
Since the solution of this problem does not exhibit any singular behavior at the boundaries ∂A and ∂B , $S_A(\cdot) = S_B(\cdot) = 0$ and the training is only performed for the component $n_{\theta_0}(\cdot)$ in (26). The problem is solved for the temperatures $T = 0.5$ and $T = 0.2$. Table 1 summarizes the numerical results and the training parameters. Notice that when the temperature T is lower, the distribution ρ is sparser in $\Omega \setminus (A \cup B)$. Therefore, more samples are used for $T = 0.2$ than for $T = 0.5$ in

T	E	δ	c	No. training samples	Batch size	No. testing samples
0.5	0.014	0.003	15	1.5×10^5	1000	4.0×10^5
0.2	0.011	0.003	15	8.0×10^5	1000	8.0×10^5

Table 1: Results for the double-well potential problem.

order to achieve a comparable precision. For example, for $T = 0.5$, 1.5×10^5 samples are used for $\Omega \setminus (A \cup B)$ and the relative error is $E = 0.014$. On the other hand, for $T = 0.2$, 8.0×10^5 samples are used and the relative error of the neural network solution is $E = 0.011$.

The numerical solution for $T = 0.5$ is shown in Fig. 2. The committor function q_θ represented by a neural network and the committor function q^* obtained via solving (36) are plotted along the x_1 dimension for a fixed (x_2, \dots, x_d) . The plot demonstrates that the NN committor function gives a satisfactory approximation to the true solution. We comment here that the final error is not sensitive to the parameter δ . For example, for the case with $T = 0.5$, $\delta = 0.003$ is used. If $\delta = 0.01, 0.03, 0.05$ are chosen instead, the corresponding final errors are $E = 0.013, 0.013, 0.013$, respectively. We also comment that using the multi-step sampling method (23) can slightly improve the final accuracy. For example, for the case with $T = 0.5$ and $\delta = 0.05$, if the multiple-step scheme (23) with $M = 10$ is used, the final error is $E = 0.012$.

Figure 2: The committor function for the double-well potential along x_1 dimension when $T = 0.5$ for an arbitrarily chosen (x_2, \dots, x_d) with $d = 10$.

4.1.1. COMPARISON WITH THE METHOD OF (KHOO ET AL., 2019)

In this section, we compare the proposed method with the one in (Khoo et al., 2019) in terms of speed, accuracy, and robustness on the double-well potential problem with $T = 0.5$. We use the same number of training samples (1.5×10^5) and testing samples (4.0×10^5), NN architectures, and hyperparameters in the implementation of both methods. The numerical tests are carried out on 4 N1 virtual CPUs on the Google Cloud platform with altogether 26 GB memory and a Tesla K80 GPU. In order to compare penalty coefficients on the same scale, a normalized penalty coefficient c_{norm} is used: it is defined via $c_{norm} = c/\delta$ for the new method with c given in (18) and via $c_{norm} = \tilde{c}$ for the old method with \tilde{c} given in (6).

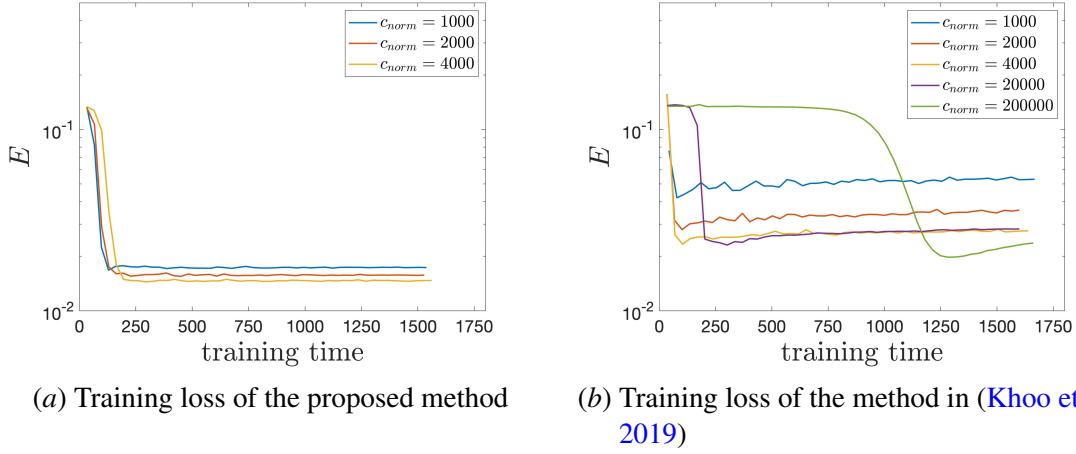


Figure 3: Comparison of the training process of the proposed method and (Khoo et al., 2019)’s method. (a): The training loss versus the training time used for the proposed method. (b): The training loss versus the training time used for the method in (Khoo et al., 2019). We report the error E (defined in (33)) versus the training time used. Here, c_{norm} stands for the normalized penalty coefficient defined in Section 4.1.1.

Fig. 3 demonstrates a clear difference between the behavior of the two methods during the training. As shown in Fig. 3(a), when using the method proposed in this paper, the approximate solution converges quickly and the final relative error is rather small, regardless of the choice of penalty coefficients. In contrast, as shown in Fig. 3(b), when using the method proposed in (Khoo et al., 2019), different penalty coefficients lead to different training behaviors. When the penalty parameter c_{norm} is small, the time used to reach convergence is short but the final relative error is relatively large. When a large penalty parameter is used, the relative error is reduced but it is still higher than the proposed method. Moreover, the time for training to converge is long when using a large penalty parameter. In conclusion, when using the method in (Khoo et al., 2019), the penalty coefficient needs to be carefully tuned in order to have a performance close to the proposed method.

4.2. The rugged-Mueller potential

In this example, we consider the committor function corresponding to the following rugged-Mueller potential:

$$V(\mathbf{x}) = \tilde{V}(x_1, x_2) + \frac{1}{2\sigma^2} \sum_{i=3}^d x_i^2, \quad (37)$$

where

$$\tilde{V}(x_1, x_2) = \sum_{i=1}^4 D_i e^{a_i(x_1 - X_i)^2 + b_i(x_1 - X_i)(x_2 - Y_i) + c_i(x_2 - Y_i)^2} + \gamma \sin(2k\pi x_1) \sin(2k\pi x_2) \quad (38)$$

is the 2-dimensional rugged-Mueller potential with the parameters

$$\begin{aligned}
 [a_1, a_2, a_3, a_4] &= [-1, -1, -6.5, 0.7], & [b_1, b_2, b_3, b_4] &= [0, 0, 11, 0.6], \\
 [c_1, c_2, c_3, c_4] &= [-10, -10, -6.5, 0.7], & [D_1, D_2, D_3, D_4] &= [-200, -100, -170, 15], \\
 [X_1, X_2, X_3, X_4] &= [1, 0, -0.5, -1], & [Y_1, Y_2, Y_3, Y_4] &= [0, 0.5, 1.5, 1], \\
 [\gamma, k, \sigma, d] &= [9, 5, 0.05, 10].
 \end{aligned} \tag{39}$$

The domain of interest Ω of this example is $[-1.5, 1] \times [-0.5, 2] \times \mathbb{R}^{d-2}$ and the regions A and B are the following two cylinders:

$$\begin{aligned}
 A &= \left\{ \mathbf{x} \in \mathbb{R}^d \mid \sqrt{(x_1 + 0.57)^2 + (x_2 - 1.43)^2} \leq 0.3 \right\}, \\
 B &= \left\{ \mathbf{x} \in \mathbb{R}^d \mid \sqrt{(x_1 - 0.56)^2 + (x_2 - 0.044)^2} \leq 0.3 \right\}.
 \end{aligned} \tag{40}$$

In order to compute the error, q^* is solved approximately within the x_1x_2 -plane. More precisely, we first apply finite element method on uniform grid to (3) in 2 dimensions with the potential \tilde{V} , the domain $\tilde{\Omega} = [-1.5, 1] \times [-0.5, 2]$, and the regions \tilde{A} and \tilde{B} being the projection of the A and B defined in (40) onto the x_1x_2 -plane. Once the 2-dimensional committor function \tilde{q} is available, the approximation is $q^*(\mathbf{x}) = \tilde{q}(x_1, x_2)$. The code of the finite element method is provided by the authors of (Lai and Lu, 2018).

As mentioned in Section 2.3, the singularity functions S_A and S_B should be set as the fundamental solutions of (3) when taking $T \rightarrow \infty$. In this case, the limiting committor function as $T \rightarrow \infty$ has the form $q(\mathbf{x}) = \tilde{q}(x_1, x_2)$ that satisfies a 2-dimensional Laplace equation, the fundamental solution to which has the form

$$\Phi(x_1, x_2) = -\frac{1}{4\pi} \log((x_1 - a)^2 + (x_2 - b)^2).$$

Therefore, we set $S_A = \log((x_1 + 0.57)^2 + (x_2 - 1.43)^2)$ and $S_B = \log((x_1 - 0.56)^2 + (x_2 - 0.044)^2)$.

Numerical experiments are carried out when $T = 22$ and $T = 40$. In both situations, 6.0×10^5 samples are used in $\Omega \setminus (A \cup B)$. The relative errors are $E = 0.024$ when $T = 22$, and $E = 0.023$ when $T = 40$. Table 2 summarizes the numerical error and the parameters used in the experiments. The numerical solution is plotted in Fig. 4, where the exact solutions q^* are shown on the left and

(T, σ)	E	δ	c	No. training samples	Batch size	No. testing samples
(22, 0.05)	0.024	0.001	500	6.0×10^5	5000	1.0×10^6
(40, 0.05)	0.023	0.001	500	6.0×10^5	5000	1.0×10^6

Table 2: Results for the rugged-Mueller potential problem.

the committor functions q_θ represented by neural network are plotted on the right. The plots show that the NN approximation shows good agreement with the true solution.

4.3. The Ginzburg-Landau model

The Ginzburg-Landau theory is developed to give a mathematical description of superconductivity (Hoffmann and Tang, 2012). In this example, we discuss a simplified Ginzburg-Landau phase transition model. The Ginzburg-Landau energy in one dimension is defined as:

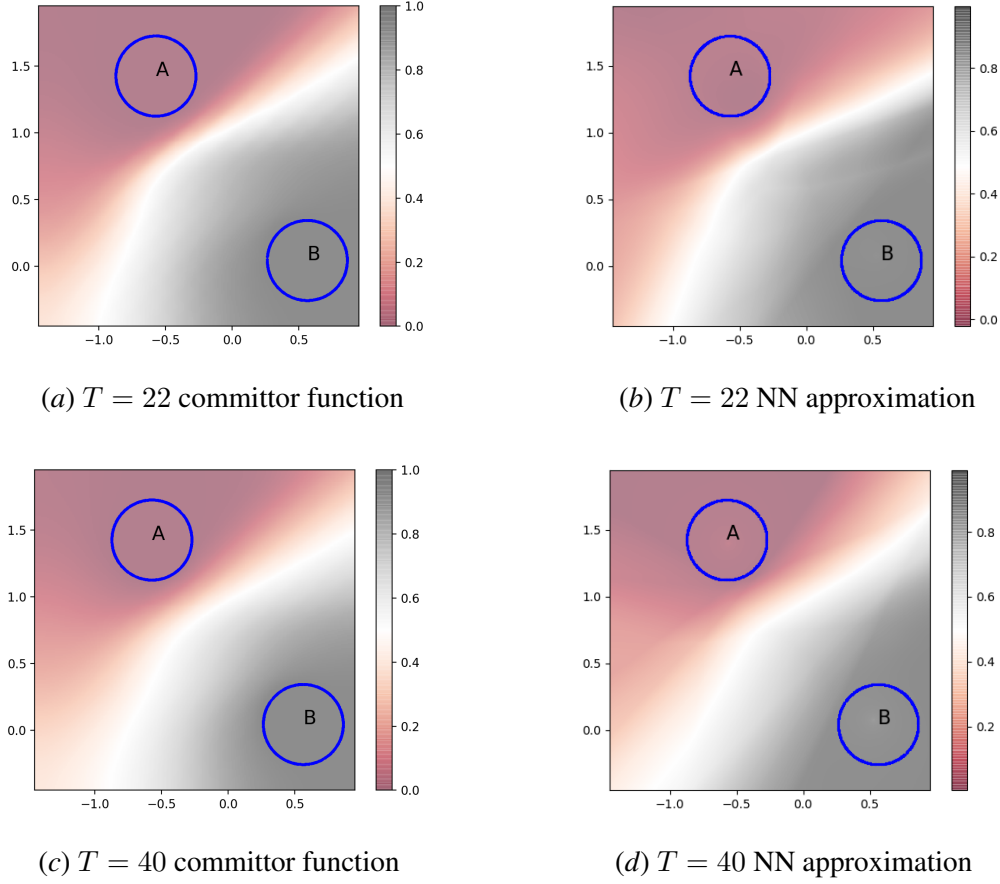


Figure 4: Comparisons between the NN represented committor functions and the ground truths. (a): The ground truth committor function for $T = 22$. (b): The NN parameterized committor function for $T = 22$. (c): The ground truth committor function for $T = 40$. (d): The NN parameterized committor function for $T = 40$.

$$\tilde{V}[u] = \int_0^1 \frac{\lambda}{2} u_x^2 + \frac{1}{4\lambda} (1 - u^2)^2 dx, \quad (41)$$

where λ is a small positive parameter and u is a sufficiently smooth function on $[0, 1]$ with boundary conditions $u(0) = u(1) = 0$.

The high-dimensionality nature of the committor functions is a direct result of the discretization of u . With a numerical discretization, $u(x)$ is uniformly discretized by $U = (U_1, \dots, U_d)$ defined on a uniform grid on $[0, 1]$ with the boundary conditions $U_0 = U_{d+1} = 0$. Then the continuous Ginzburg-Landau energy is approximated by a discrete one:

$$V(U) := \tilde{V}_h[U] = \sum_{i=1}^{d+1} \frac{\lambda}{2} \left(\frac{U_i - U_{i-1}}{h} \right)^2 + \frac{1}{4\lambda} (1 - U_i^2)^2, \quad (42)$$

where the grid size $h = 1/(d + 1)$. In this experiment we use $h = 1/50$ and the dimension $d = 49$. $V(U)$ has two local minima $u_{\pm}(\cdot)$ shown in Fig. 5. The regions A and B are taken as the spheres $\{U : \|U - u_{\pm}\| \leq r\}$, where $\|\cdot\|$ is the Euclidean norm, and the radius r is chosen to be 3.

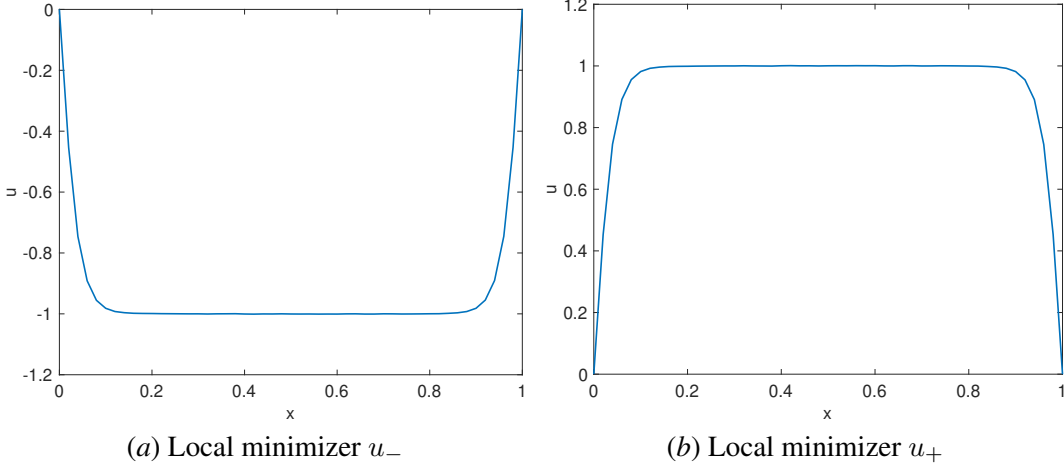


Figure 5: Two local minima of the energy (42) with $\lambda = 0.03$. (a): u_- , (b): u_+ .

Based on the discretization used, $d = 49$. According to Section 2.3, the singularities S_A and S_B are set to $S_A = |U - u_-|^{2-d}$, $S_B = |U - u_+|^{2-d}$. The numerical results are reported for the temperatures $T = 30$ and $T = 20$. For both cases 2.0×10^5 samples are used in $\Omega \setminus (A \cup B)$. In Table 3 we summarize the parameters used in the experiments with $T = 30$ and $T = 20$.

T	δ	c	No. training samples	Batch size
20	0.002	200	2.0×10^5	5000
30	0.001	200	2.0×10^5	5000

Table 3: Parameters for the Ginzburg-Landau problem.

In this problem, it is intractable to obtain the exact q^* due to the high dimensionality and therefore we are not able to estimate the relative error E directly. Instead, we study the region near the $\frac{1}{2}$ -isosurface of committor function q_θ , which is defined as $\Gamma_{\frac{1}{2}, \epsilon} = \{U : |q_\theta(U) - \frac{1}{2}| < \epsilon\}$. If q_θ is indeed a satisfactory approximation of q^* , then for a trajectory given by (7) starting from an arbitrary point $x_0 \in \Gamma_{\frac{1}{2}, \epsilon}$, the probability of entering B before A should be close to $\frac{1}{2}$.

More precisely, we first identify m states $\{\tilde{x}_j\}_{j=1}^m$ on $\Gamma_{\frac{1}{2}, \epsilon}$. From each \tilde{x}_j , N trajectories are generated according to (7). Let us denote the number of trajectories reaching B before A as n . If the NN committor function is accurate, then by the central limit theorem, when N is large, the distribution of n/N should be approximately $\mathcal{N}(\frac{1}{2}, (4N)^{-1})$, i.e. the normal distribution with mean $\frac{1}{2}$ and variance $(4N)^{-1}$. In the actual experiment with $\epsilon = 0.01$, $m = 120$, and $N = 100$, the resulting statistics contain n_j/N for $j = 1, 2, \dots, 120$.

The numerical results we get when $T = 20$ and $T = 30$ are illustrated in Fig. 6 and Fig. 7, respectively. The histogram of $\{n_j/N\}_{j=1}^{120}$ is compared with the normal distribution $\mathcal{N}(\frac{1}{2}, 1/400)$ on the left, and the Q-Q (quantile-quantile) plot of the distribution of $\{n_j/N\}_{j=1}^{120}$ versus $\mathcal{N}(\frac{1}{2}, 1/400)$ is shown on the right. These figures demonstrate that the distribution of $\{n_j/N\}_{j=1}^{120}$ is in good

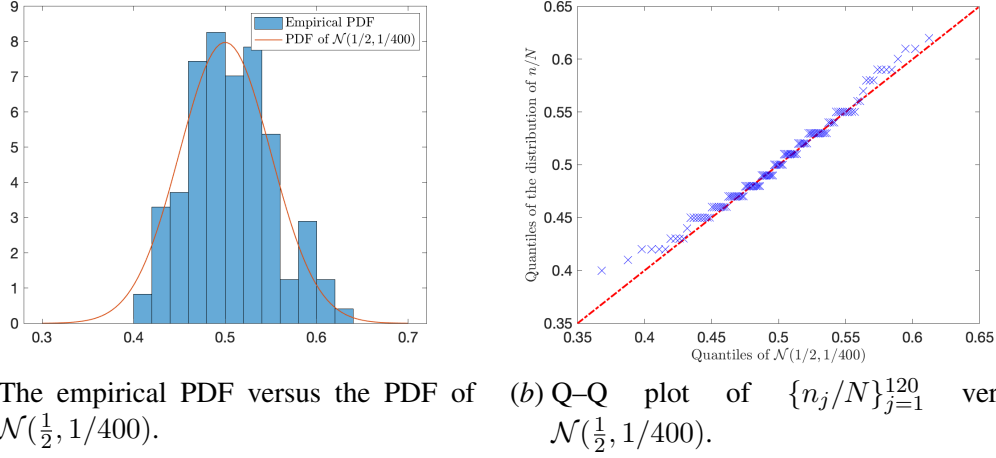


Figure 6: Numerical results of the Ginzburg-Landau model when $T = 20$. (a): Empirical probability density function (PDF) of $\{n_j/N\}_{j=1}^{120}$ compared with the PDF of $\mathcal{N}(\frac{1}{2}, 1/400)$. (b): Q-Q (quantile-quantile) plot of $\{n_j/N\}_{j=1}^{120}$ compared with the Q-Q plot of $\mathcal{N}(\frac{1}{2}, 1/400)$.

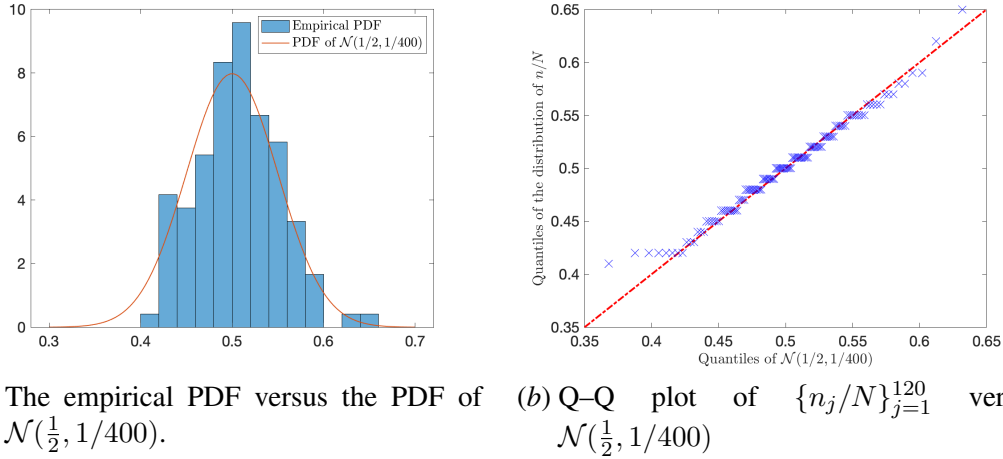


Figure 7: Numerical results of the Ginzburg-Landau model when $T = 30$. (a): Empirical PDF of $\{n_j/N\}_{j=1}^{120}$ compared with the PDF of $\mathcal{N}(\frac{1}{2}, 1/400)$. (b): Q-Q (quantile-quantile) plot of $\{n_j/N\}_{j=1}^{120}$ versus $\mathcal{N}(\frac{1}{2}, 1/400)$.

agreement with the normal distribution $\mathcal{N}(\frac{1}{2}, 1/400)$. As mentioned previously in Section 2.2, we can integrate importance sampling (Li et al., 2019; Rotskoff and Vanden-Eijnden, 2020) when dealing with metastability, especially when the temperature is relatively low. For instance, we can generate the initial state \mathbf{x} in (22) using importance sampling and then obtain \mathbf{x}_δ via (22).

5. Conclusion

In this paper, we improve the method in (Khoo et al., 2019) that solves for the neural network parameterized committor function. In particular, we show that the committor function satisfies an integral equation based on the semigroup of the Fokker-Planck operator. This integral formulation allows us to remove the explicit gradient and handle the boundary conditions naturally. The integrals in the variational form of this new equation can be conveniently approximated via sampling, and the committor function can be solved for using a neural network parameterization and stochastic gradient descent. The resulting algorithm is shown to be faster and less sensitive to the penalty parameter when compared with the approach in (Khoo et al., 2019). The convergence of the training process is guaranteed in the lazy training regime.

This work suggests a few directions of future research. First on the numerical side, the SDE (7) is currently integrated with the Euler-Maruyama method. It will be useful to explore higher order integration schemes. Our approximation for the stopping time is also rather primitive and it will be beneficial to explore better decision rules. Second, the proposed method can be readily applied to other high-dimensional partial differential equations that possess probabilistic interpretations.

References

- Andrea Agazzi and Jianfeng Lu. Temporal-difference learning with nonlinear function approximation: lazy training and mean field regimes, 2020.
- Lenaïc Chizat, Edouard Oyallon, and Francis Bach. On lazy training in differentiable programming. In *Advances in Neural Information Processing Systems*, pages 2937–2947, 2019.
- Ronald R Coifman, Ioannis G Kevrekidis, Stéphane Lafon, Mauro Maggioni, and Boaz Nadler. Diffusion maps, reduction coordinates, and low dimensional representation of stochastic systems. *Multiscale Modeling & Simulation*, 7(2):842–864, 2008.
- Nelson Dunford and Jacob T Schwartz. Linear operators. part ii. spectral theory. *Bull. Amer. Math. Soc*, 2(9904):11348–9, 1965.
- EB Dynkin. Markov processes. vols. i, ii. publishers. *New York: Academic Press Inc. Translated with the authorization and assistance of the author by J. Fabius, V. Greenberg, A. Maitra, G. Majone. Die Grundlehren der Mathematischen Wissenschaften, Bände*, 121:122, 1965.
- Weinan E and Eric Vanden-Eijnden. Towards a theory of transition paths. *Journal of statistical physics*, 123(3):503, 2006.
- Weinan E, Weiqing Ren, and Eric Vanden-Eijnden. Finite temperature string method for the study of rare events. *J. Phys. Chem. B*, 109(14):6688–6693, 2005.
- K-H Hoffmann and Qi Tang. *Ginzburg-Landau phase transition theory and superconductivity*, volume 134. Birkhäuser, 2012.
- Chi Jin, Praneeth Netrapalli, Rong Ge, Sham M Kakade, and Michael I Jordan. On nonconvex optimization for machine learning: Gradients, stochasticity, and saddle points. *arXiv preprint arXiv:1902.04811*, 2019.
- Olav Kallenberg. *Foundations of modern probability*. Springer Science & Business Media, 2006.
- Yuehaw Khoo, Jianfeng Lu, and Lexing Ying. Solving for high-dimensional committor functions using artificial neural networks. *Research in the Mathematical Sciences*, 6(1):1, 2019.
- Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- Peter E Kloeden and Eckhard Platen. *Numerical solution of stochastic differential equations*, volume 23. Springer Science & Business Media, 2013.
- Harold Kushner and G George Yin. *Stochastic approximation and recursive algorithms and applications*, volume 35. Springer Science & Business Media, 2003.
- Rongjie Lai and Jianfeng Lu. Point cloud discretization of fokker–planck operators for committor functions. *Multiscale Modeling & Simulation*, 16(2):710–726, 2018.
- Qianxiao Li, Bo Lin, and Weiqing Ren. Computing committor functions for the study of rare events using deep learning. *The Journal of Chemical Physics*, 151(5):054112, 2019.

- Jianfeng Lu and James Nolen. Reactive trajectories and the transition path process. *Probability Theory and Related Fields*, 161(1):195–244, 2015.
- Yann Ollivier. Approximate temporal difference learning is a gradient descent for reversible policies. *arXiv preprint arXiv:1805.00869*, 2018.
- Grigorios A Pavliotis. *Stochastic processes and applications: diffusion processes, the Fokker-Planck and Langevin equations*, volume 60. Springer, 2014.
- Grant M. Rotskoff and Eric Vanden-Eijnden. Learning with rare data: Using active importance sampling to optimize objectives dominated by rare events, 2020.
- Eric Vanden-Eijnden and Maddalena Venturoli. Revisiting the finite temperature string method for the calculation of reaction tubes and free energies. *The Journal of chemical physics*, 130(19):05B605, 2009.
- Eric Vanden-Eijnden et al. Transition-path theory and path-finding algorithms for the study of rare events. *Annual review of physical chemistry*, 61:391–420, 2010.

Appendix A. Proof of Proposition 1

Recall that the Markov semigroup is defined as follows.

Definition 6 *The Markov semigroup $(P_t)_{t \geq 0}$ associated with a Markov process $(\mathbf{x}_t)_{t \geq 0}$ is defined as*

$$P_t f(\mathbf{x}) = \mathbb{E}^{\mathbf{x}}(f(\mathbf{x}_t)) = \mathbb{E}(f(\mathbf{x}_t) \mid \mathbf{x}_0 = \mathbf{x}), \quad t \geq 0, \quad \mathbf{x} \in \Omega, \quad (43)$$

where $f : \mathbb{R}^d \rightarrow \mathbb{R}$ is bounded continuous.

Recall that $\tau = \tau_{A \cup B}$. While $\tau \wedge \delta$ cannot be directly plugged into Definition 6 since it is a random variable instead of a constant, the operator $P_{\tau \wedge \delta} \equiv P$ can still be defined by (8), and by the strong Markov property of the Langevin process $(\mathbf{x}_t)_{t \geq 0}$, $(\mathbf{x}_{\tau \wedge \delta})_{\delta \geq 0}$ is also a Markov process, thus $(P_{\tau \wedge \delta})_{\delta \geq 0}$ is a Markov semigroup. Now we proceed to the proof with the help of Dynkin's formula described in the following theorem:

Theorem 7 (Dynkin, 1965)

Consider an Itô diffusion $\{\mathbf{x}_t\}_{t \geq 0}$ defined by the following d -dimensional stochastic differential equation (SDE)

$$d\mathbf{x}_t = b(\mathbf{x}_t)dt + \sigma(\mathbf{x}_t)d\mathbf{w}_t. \quad (44)$$

Let \mathcal{A} be the infinitesimal generator of $\{\mathbf{x}_t\}_{t \geq 0}$, which is defined by

$$\mathcal{A}f(\mathbf{x}) = \lim_{t \rightarrow 0+} \frac{P_t f(\mathbf{x}) - f(\mathbf{x})}{t}, \quad (45)$$

where $f \in D(\mathcal{A}) := \{f \in C_0(\mathbb{R}^d) \text{ such that the limit (45) exists}\}$, then for a stopping time τ such that $\mathbb{E}^{\mathbf{x}}\tau < \infty$,

$$\mathbb{E}^{\mathbf{x}}[f(\mathbf{x}_\tau)] = f(\mathbf{x}) + \mathbb{E}^{\mathbf{x}} \left[\int_0^\tau \mathcal{A}f(\mathbf{x}_s) ds \right], \quad (46)$$

for any $f \in C_0^2(\mathbb{R}^d)$. Moreover, when τ is the first exit time for a bounded set, (46) holds for any $f \in C^2(\mathbb{R}^d)$

For any fixed $\delta > 0$, $\tau \wedge \delta$ is a stopping time, and is bounded. Although the committor function q may not be compactly supported, we can still use the formula (46) with the stopping time $\tau \wedge \delta$ replaced by $\tau \wedge \delta \wedge \tau_{B_r}$, where B_r is the ball with the radius being r and the center being the origin. When δ is sufficiently small and r sufficiently large, the difference between $P_{\tau \wedge \delta}$ and $P_{\tau \wedge \delta \wedge \tau_{B_r}}$ is negligible in practice, as long as V is a confining potential. For this reason, we continue to use the notation $P = P_{\tau \wedge \delta}$ in the problem formulation instead of the more cumbersome $P_{\tau \wedge \delta \wedge \tau_{B_r}}$. Therefore for the solution q of the equation (3), we have

$$Pq(\mathbf{x}) = q(\mathbf{x}) + \mathbb{E}^{\mathbf{x}} \left[\int_0^{\tau \wedge \delta} -Lq(\mathbf{x}_s) ds \right] = q(\mathbf{x}), \quad \forall \mathbf{x} \in \Omega \setminus A \cup B, \quad (47)$$

as long as we can prove $\mathcal{A} = -L$. In order to show this, we introduce a result in (Kallenberg, 2006).

Theorem 8 (Kallenberg, 2006) *Assume that \mathbf{x}_t satisfies the SDE (44). When $a(\mathbf{x}) = \sigma(\mathbf{x})\sigma(\mathbf{x})^\top$ and b are bounded and Lipschitz continuous, then the generator \mathcal{A} of the semigroup associated with (44) has the following form.*

$$\mathcal{A} = \frac{1}{2} \sum_{i,j=1}^d a_{ij}(\mathbf{x}) \frac{\partial^2}{\partial x_i \partial x_j} + b(\mathbf{x}) \cdot \nabla. \quad (48)$$

Now we are ready to justify (47).

For the Langevin process (1), the coefficients are $a_{ij}(\mathbf{x}, t) = \mathbf{1}_{\{i=j\}}1/\beta$, and $b = \nabla V$, so the conditions in Theorem 8 on a are naturally satisfied. Since ∇V is bounded and Lipschitz on \mathbb{R}^d , the conditions in Theorem 8 on b are satisfied as well. Thus by Theorem 8, the generator \mathcal{A} of $(P_t)_{t \geq 0}$ is

$$\mathcal{A} = \frac{1}{2} \sum_{i,j=1}^d \frac{1}{\beta} \mathbf{1}_{\{i=j\}} \frac{\partial^2}{\partial x_i \partial x_j} + \nabla V(\mathbf{x}) \cdot \nabla = -L.$$

Plugging this into Dynkin's formula (46) yields (9), which is what we want to prove.

Appendix B. Proof of Proposition 2

Consider the stationary Langevin process, i.e. $\{\mathbf{x}_t\}_{t \geq 0}$ such that $\mathbf{x}_0 \sim \rho$ and \mathbf{x}_t satisfies (1). It is known that this process is reversible (see for example (Pavliotis, 2014)). In other words, we have

$$\mathbb{E}_{\mathbf{x} \sim \rho} u(\mathbf{x}) v(\mathbf{x}_\delta) = \mathbb{E}_{\mathbf{x} \sim \rho} u(\mathbf{x}_\delta) v(\mathbf{x}),$$

for any $u, v \in L^2_\rho(\Omega \setminus A \cup B)$. Now since the law of the trajectory $\{\mathbf{x}_{\tau \wedge t}\}_{t < \delta}$ is the same as $\{\mathbf{x}_t\}_{t < \delta}$ as long as $\delta < \tau$, we have

$$\mathbb{E}_{\mathbf{x} \sim \rho} u(\mathbf{x}) v(\mathbf{x}_\delta) \mathbf{1}_{\{\delta < \tau\}} = \mathbb{E}_{\mathbf{x} \sim \rho} u(\mathbf{x}_\delta) v(\mathbf{x}) \mathbf{1}_{\{\delta < \tau\}},$$

which is exactly $\langle u, P^i v \rangle_\rho = \langle P^i u, v \rangle_\rho$.

Appendix C. Proof of Proposition 4

(31) can be rewritten as

$$P^i f(\mathbf{x}) = \int_{\Omega \setminus A \cup B} \frac{p^i(\mathbf{x}, \mathbf{y})}{\rho(\mathbf{y})} f(\mathbf{y}) \rho(\mathbf{y}) d\mathbf{y},$$

and when (32) holds, we have

$$\begin{aligned} & \iint_{\Omega \setminus A \cup B \times \Omega \setminus A \cup B} \left(\frac{p^i(\mathbf{x}, \mathbf{y})}{\rho(\mathbf{y})} \right)^2 \rho(\mathbf{x}) \rho(\mathbf{y}) d\mathbf{x} d\mathbf{y} \\ &= \iint_{\Omega \setminus A \cup B \times \Omega \setminus A \cup B} \frac{p^i(\mathbf{x}, \mathbf{y}) \rho(\mathbf{x}) p^i(\mathbf{x}, \mathbf{y})}{\rho(\mathbf{y})} d\mathbf{x} d\mathbf{y} \\ &= \iint_{\Omega \setminus A \cup B \times \Omega \setminus A \cup B} p^i(\mathbf{x}, \mathbf{y}) p^i(\mathbf{y}, \mathbf{x}) d\mathbf{x} d\mathbf{y} < \infty, \end{aligned}$$

where in the second equality we used the symmetry of P^i . Thus P^i is a Hilbert-Schmidt operator on $L^2_\rho(\Omega \setminus A \cup B)$ (Dunford and Schwartz, 1965). Thus it is compact, and every point in its spectrum is an eigenvalue. Since P^i is contractive, all of its eigenvalues are less or equal than 1, and it suffices to prove that 1 is not an eigenvalue for P^i . If 1 was an eigenvalue of P^i , then there exists a nonzero

function $f \in L^2_\rho(\Omega \setminus A \cup B)$ such that $P^i f = f$, and by using (31) n times, we get

$$\begin{aligned}
 f(\mathbf{x}) &= \int_{\Omega \setminus A \cup B} p^i(\mathbf{x}, \mathbf{y}) f(\mathbf{y}) d\mathbf{y} \\
 &= \int \int_{\Omega \setminus A \cup B \times \Omega \setminus A \cup B} p^i(\mathbf{x}, \mathbf{y}) p^i(\mathbf{x}, \mathbf{z}) f(\mathbf{z}) d\mathbf{y} d\mathbf{z} \\
 &= \dots \\
 &= \mathbb{E}^{\mathbf{x}} \left(f(\mathbf{x}_{n\delta}) \mathbf{1}_{\{n\delta < \tau\}} \right), \quad \forall n > 0.
 \end{aligned}$$

But the Langevin process (1) is ergodic, and the equilibrium measure ρ is positive, so $\mathbf{1}_{\{n\delta < \tau\}} \rightarrow 0$ almost surely when $n \rightarrow \infty$, resulting in $f = 0$, which contradicts with the assumption that f is nonzero. This contradiction shows that 1 is not an eigenvalue of P^i and closes the proof.