Computational Methods for Transition States and Pathways in Rare Events

Edited by Shuting Gu

First published 2025

ISBN: 978-1-032-99647-9 (hbk)
ISBN: 978-1-032-99718-6 (pbk)
ISBN: 978-1-003-60565-2 (ebk)

Chapter 3

Variants of Gentlest Ascent Dynamics for Transition States

DOI: 10.1201/9781003605652-3

The funder for this chapter is The Natural Science Foundation of Top Talent of SZTU GDRC202137, the National Natural Science Foundation of China (NSFC) 11901211 and Shenzhen Technology University

CRC Press
Taylor & Francis Group
Boca Raton London New York

CRC Press is an imprint of the
Taylor & Francis Group, an **informa** business

CHAPTER 3

Variants of Gentlest Ascent Dynamics for Transition States

In this chapter, we explore several innovative extensions of the gentlest ascent dynamics (GAD), tailored to efficiently locate transition states in rare events. These variants address the complexities arising from multiscale systems, non-gradient dynamics, and energy functionals with higher-order spatial derivatives. Firstly, we introduce a multiscale gentlest ascent dynamics (MsGAD) for identifying transition states in the effective dynamics of slow-fast systems. By leveraging multiscale numerical techniques, MsGAD effectively resolves the dynamics while ensuring computational efficiency. Additionally, adaptive strategies are employed to accelerate convergence, optimizing the overall performance of the method. Secondly, for non-gradient systems, we propose a simplified (GAD) that uses only one direction variable, thereby avoiding the computational burden of Jacobian matrix transpose operations. This approach significantly reduces computational costs while maintaining accuracy. The theoretical connection between this simplified GAD and Hamiltonian dynamics is detailed in the Appendix, providing deeper insight into its mechanics. Thirdly, building upon the simplified GAD, we develop a variant tailored to multiscale models of non-gradient slow-fast systems. This adaptation further reduces computational overhead, making it feasible to tackle complex multiscale problems. Finally, to handle energy functionals in the H^{-1} metric, where higher-order spatial derivatives pose numerical challenges, we introduce a linear projection operator. This modification simplifies the (simplified) GAD framework, enabling efficient and accurate computation of saddle points for such functionals.

To validate the proposed methods, we present a series of numerical experiments involving stochastic ordinary differential equations and partial differential

DOI: 10.1201/9781003605652-3

equations. These examples showcase the effectiveness and computational advantages of the developed GAD variants across diverse scenarios.

3.1 MULTISCALE GENTLEST ASCENT DYNAMICS

The slow-fast system $(X(t), Y(t)) \in \mathcal{X} \times \mathcal{Y}$ represents a typical dynamical system characterized by two disparate time scales:

$$\begin{cases} \dot{X}^\epsilon(t) = f(X^\epsilon, Y^\epsilon), & \text{(3.1a)} \\ \dot{Y}^\epsilon(t) = \dfrac{1}{\epsilon} b(X^\epsilon, Y^\epsilon) + \dfrac{1}{\sqrt{\epsilon}} \sigma(X^\epsilon, Y^\epsilon)\eta(t), & \text{(3.1b)} \end{cases}$$

where ϵ is a small parameter. X^ϵ denotes the slow variable and Y^ϵ is the fast variable. For simplicity, we assume $\mathcal{X} = \mathbb{R}^n$ and $\mathcal{Y} = \mathbb{R}^m$. Here, $f(x, y) : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ and $b(x, y) : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^m$ are smooth vector fields. The $m \times m$ diffusion matrix $\sigma(x, y)$ is assumed non-degenerate for all x and y, and $\eta(t)$ is a zero-mean Gaussian noise in $\mathbb{R}^m$ with a specified covariance function $\mathrm{E}(\eta(t)\eta(t'))$.

As $\epsilon \to 0$, the fast dynamics can be effectively "slaved" and eliminated, while the slow dynamics converge to the effective dynamics of a deterministic function $\bar{X}(t)$. This result is grounded in the averaging principle, a well-established method in classical mathematical literature [57, 80, 58].

The averaging principle requires an ergodicity assumption for the fast process Y^ϵ. Let $\widetilde{Y}^x$, referred to as the virtually fast process, satisfy:

$$\dot{\widetilde{Y}}^x = b(x, \widetilde{Y}^x) + \sigma(x, \widetilde{Y}^x)\eta(t), \tag{3.2}$$

for a fixed parameter x. Assume $\widetilde{Y}^x$ is ergodic for each x, and its unique invariant measure $\mu_x(\mathrm{d}y)$ has a density function $\rho(x, y)$:

$$\mu_x(\mathrm{d}y) = \rho(x, y)\mathrm{d}y = \frac{1}{Z(x)} e^{-U(x,y)} \mathrm{d}y, \tag{3.3}$$

where the normalization factor $Z(x)$ is given by:

$$Z(x) \doteq \int_{\mathcal{Y}} e^{-U(x,y)} \mathrm{d}y. \tag{3.4}$$

By ergodicity, the expectation of any integrable function $u(y)$ with respect to μ_x can be approximated by the time average:

$$\int u(y)\mu_x(\mathrm{d}y) = \lim_{T\to\infty} \frac{1}{T} \int_0^T u(\widetilde{Y}(t))\mathrm{d}t.$$

The averaging principle [16, 31] then states that as $\epsilon \to 0$, the slow variable $X^\epsilon(t)$ converges to $\bar{X}(t)$, which satisfies the deterministic ordinary differential equation (ODE):

$$\dot{\bar{X}} = F(\bar{X}), \quad \text{where } F(x) \doteq \int f(x,y)\mu_x(dy). \tag{3.5}$$

In general, the averaging function $F(x)$ lacks a closed-form expression and must be approximated numerically to solve the ODE efficiently.

This section outlines the multiscale framework for analyzing slow-fast systems using the averaging principle. The effective dynamics $F(x)$ F(x) governs the evolution of the slow variables, enabling the reduction of computational complexity while retaining essential system behaviours. The next subsections will extend this framework to the Multiscale Gentlest Ascent Dynamics (MsGAD) for locating transition states in such systems.

3.1.1 GAD and MsGAD for Slow-Fast System

To locate the transition states of the effective dynamics (3.5), we can proceed in two ways: (1) directly applying the GAD to the effective dynamics; (2) starting from the full slow-fast system and deriving a multiscale extension, termed MsGAD (Multiscale Gentlest Ascent Dynamics). Both approaches are elaborated below.

Gentlest ascent dynamics for the effective dynamics By extending the GAD formulation to the effective dynamics (3.5), we obtain the following system:

$$\begin{cases} \dot{x}(t) = F(x) - 2\dfrac{\langle F(x), w\rangle}{\langle v, w\rangle} v, & \text{(3.6a)} \\ \gamma \dot{v}(t) = DF(x)v - \alpha v, & \text{(3.6b)} \\ \gamma \dot{w}(t) = DF(x)^{\mathsf{T}} w - \beta w, & \text{(3.6c)} \end{cases}$$

where $F(x) = \int f(x,y)\rho(x,y)\mathrm{d}y$ is the averaged drift vector field, and $DF(x)$ is its Jacobi matrix, and $\rho(x,y) = Z(x)^{-1}e^{-U(x,y)}$ is the invariant density of the fast process. The parameters γ, α and β retain the same roles as in (2.1) from section 2.1. The physical and geometrical interpretation of (3.6) remains consistent with the original GAD framework, albeit with the dynamics now governed by the averaged fields. Additionally, we define:

$$g(x,y) \doteq -\nabla_x U(x,y), \quad \text{and} \quad G(x) \doteq \int g(x,y)\rho(x,y)\mathrm{d}y. \tag{3.7}$$

Using the normalization factor $Z(x)$, the gradient of $\log Z(x)$ can be expressed as:

$$\nabla_x \log Z(x) = Z^{-1}(x)\nabla_x Z(x) = \int g(x,y)\rho(x,y)\mathrm{d}y = G(x). \tag{3.8}$$

The Jacobian of $F(x)$ can be expressed as:

$$DF(x) = \overline{D_x f}(x) + \overline{C}(x) \tag{3.9}$$
$$= \overline{(f - F) \otimes (g - G)}(x) \tag{3.10}$$

where the overline $\overline{\theta}(x)$ denotes the average with respect to μ_x, i.e., $\overline{\theta}(x) = \overline{\theta}(x) \doteq \int \theta(x, y)\mu_x(\mathrm{d}y)$, and $u \otimes v$ represents the tensor product of two vectors.

Multiscale Gentlest ascent dynamics (MsGAD) The MsGAD extends the GAD formulation to the original slow-fast dynamics (3.1). The governing equations are given by:

$$\begin{cases} \dot{x}^\epsilon(t) = f(x^\epsilon, y^\epsilon) - 2\dfrac{\langle f(x^\epsilon, y^\epsilon), w^\epsilon \rangle}{\langle w^\epsilon, v^\epsilon \rangle} v^\epsilon, & \text{(3.11a)} \\ \dot{y}^\epsilon(t) = \dfrac{1}{\epsilon} b(x^\epsilon, y^\epsilon) + \dfrac{\sigma(x^\epsilon, y^\epsilon)}{\sqrt{\epsilon}} \eta(t), & \text{(3.11b)} \\ \gamma \dot{v}^\epsilon(t) = \left(D_x f(x^\epsilon, y^\epsilon) + C(x^\epsilon, y^\epsilon)\right) v^\epsilon - \alpha^\epsilon v^\epsilon, & \text{(3.11c)} \\ \gamma \dot{w}^\epsilon(t) = \left(D_x f(x^\epsilon, y^\epsilon) + C(x^\epsilon, y^\epsilon)\right)^\mathsf{T} w^\epsilon - \beta^\epsilon w^\epsilon, & \text{(3.11d)} \end{cases}$$

where $C(x, y)$ captures the covariance effects and is given by

$$C(x, y) = (f(x, y) - F(x)) \otimes (g(x, y) - G(x)). \tag{3.12}$$

the Lagrangian multipliers α^ϵ and β^ϵ are defined as before to enforce certain normalization conditions.

While (3.11) provides a direct multiscale framework, the expressions for $F(x)$ and $G(x)$ involve averages over the invariant distribution μ_x, complicating their numerical implementation. This can be addressed by:

1. Heterogeneous Multiscale Method (HMM): Directly estimating F and G via sampling. Details are provided in section 3.1.1.

2. Seamless Coupling Method: Rewriting (3.11) in a standard slow-fast form similar to (3.1). This requires introducing an independent copy $\hat{Y}_t^x$ of the fast process for computational purposes.

   To reduce computational cost, we utilize the following reformulation for $C(x, y)$:

   $$C_2(x, y, z) \doteq f(x, y) \otimes g(x, y) - f(x, z) \otimes g(x, y), \tag{3.13}$$

   where y and z are independent samples from μ_x. Then $\overline{C}(x) = \mathrm{E}_y \mathrm{E}_z \left[C_2(x, y, z)\right]$, where E_y and E_z are the expectations w.r.t. y and z, respectively. Thus $\overline{C}(x)$ becomes

   $$\overline{C}(x) = \int \left[f(x, y) \otimes g(x, y) - \left(\int f(x, z) \hat{\mu}_x(\mathrm{d}z) \right) \otimes g(x, y) \right] \mu_x(\mathrm{d}y),$$

where $\hat{\mu}_x(=\mu_x)$ is the equilibrium distribution of the iid copy $\hat{Y}^x$. In the numerical accomplishment, we take the new process $\hat{Y}_t^x$ to calculate the expectation $F = \bar{f}$ w.r.t. z, and the old one $\widetilde{Y}_t^x$ to calculate the expectation $\overline{C} = \overline{f \otimes g - F \otimes g}$ w.r.t. y. This allows efficient evaluation of the covariance terms without introducing excessive complexity.

Heterogeneous multiscale method (HMM)

In this section, we present the framework of the *heterogeneous multiscale method* (HMM) [17, 18]) for addressing the averaged GAD system (3.6) and the MsGAD system (3.11). The key relation between these systems is that the MsGAD system (3.11) converges to the effective dynamics (3.6) as $\epsilon \to 0$.

To implement the HMM, we introduce distinct time step sizes for the macroscopic and microscopic scales:

- Macroscopic time step size: Δt for evolving x and $\Delta\tau$ for evolving v and w (usually $\Delta\tau = \Delta t$).
- Microscopic time step size: δt for evolving y.

The procedure for the HMM scheme using a forward Euler method for the MsGAD system is as follows.

1. **Macroscopic solver for x:** Evolve x using:

$$x_{n+1} = x_n + \left(F_n - 2v_n \left\langle F_n, w_n \right\rangle / \left\langle w_n, v_n \right\rangle\right) \Delta t,$$

   where x_n approximates $\bar{X}(n\Delta t)$ and F_n, v_n and w_n are estimated in subsequent steps.

2. **Microscopic solver for y:** Simulate the fast dynamics using δt for M micro-steps:

$$y_{n,m+1} = y_{n,m} + \frac{\delta t}{\epsilon} b(x_n, y_{n,m}) + \frac{\sigma(x_n, y_{n,m})}{\sqrt{\epsilon}} \sqrt{\delta t}\, \eta_{n,m},$$

   where $m = 0, 1, \ldots, M-1$ and $\{\eta_{n,m}\}$ are independent $\mathcal{N}(0,1)$ random variables. The initial is choose as a "warm start" : $y_{n,0} \doteq y_{n-1,M}$. Only the ratio $\delta t/\epsilon$ is required, which acts as the effective step size for the virtual fast process $\tilde{Y}^x$ in (3.2).

3. **Estimation of F_n, G_n and $(DF)_n$:** Compute:

$$F_n = \frac{1}{M} \sum_{m=1}^{M} f(x_n, y_{n,m}), \quad G_n = \frac{1}{M} \sum_{m=1}^{M} g(x_n, y_{n,m}),$$

$$(DF)_n = \frac{1}{M} \sum_{m=1}^{M} \Big(D_x f(x_n, y_{n,m}) + f(x_n, y_{n,m}) \otimes g(x_n, y_{n,m}) \Big) - F_n \otimes G_n.$$

4. **Evolution of v_n and w_n:** Solve for the right and left direction v_n and w_n iteratively over K steps using $\Delta\tau$ with initial conditions $v_{n,0} = v_{n-1}$, and $w_{n,0} = w_{n-1}$:

$$\hat{v}_{n,k+1} = v_{n,k} + \Delta\tau(DF)_n v_{n,k}, \quad \hat{w}_{n,k+1} = w_{n,k} + \Delta\tau(DF)_n^{\mathsf{T}} w_{n,k},$$
$$v_{n,k+1} = \frac{\hat{v}_{n,k+1}}{|\hat{v}_{n,k+1}|}, \quad w_{n,k+1} = \frac{\hat{w}_{n,k+1}}{\langle v_{n,k+1}, \hat{w}_{n,k+1}\rangle},$$

After K steps, set $v_n = v_{n,K}$ and $w_n = w_{n,K}$.

The microscopic time step δt must be much smaller than ϵ ensure sufficient resolution of the fast dynamics. The effective step size is $\delta t/\epsilon$. The total microscopic simulation time $M \times \delta t$ must be long enough to allow the fast process Y^ϵ to relax to its equilibrium distribution. A larger M also reduces statistical errors in estimating F, G, and DF. For evolving v_n and w_n, $K = 1$ is sufficient in principle, but a larger K can improve accuracy, especially for a small γ, albeit with increased computational cost.

Remark 3.1
- *For the macroscopic solver of x and v, w, explicit schemes with larger stability regions, such as the stabilized Runge-Kutta methods, are preferred.*
- *For the microscopic solver, higher-order weakly convergent schemes are recommended for efficient long-time integration of the fast dynamics. Examples include the stochastic Heun method [59] or the non-Markovian scheme proposed in [63], particularly when σ is constant:*

$$y_{n,m+1} = y_{n,m} + \frac{\delta t}{\epsilon} b(x_n, y_{n,m}) + \frac{\sigma\sqrt{\delta t}}{2\sqrt{\epsilon}}(\eta_{n,m} + \eta_{n,m+1}), \quad m = 0, 1, \ldots,$$

where $\{\eta_{n,m}\}$ are independent $\mathcal{N}(0,1)$ random variables.

Seamless coupling method (SCM)

The seamless coupling method (SCM), introduced in [23], eliminates the need for back-and-forth communication between the macro- and micro-states in slow-fast systems. This approach acts as a boosting algorithm by increasing the small parameter ϵ to a larger value $\epsilon' = \epsilon\lambda$ (where $\lambda > 1$ is a constant). The boosted system is then solved using a smaller time step size $\widetilde{\Delta}t$, which is smaller than the macroscopic time step size Δt of the HMM. This strategy can be more computationally efficient than HMM, particularly when solving the micro-model is costly. Following this idea, we modify the MsGAD system (3.11) by increasing ϵ to $\epsilon' = \epsilon\lambda$. To apply the seamless coupling, we must handle the double expectation in the covariance matrix (see (3.12)). This requires introducing an independent copy z^ϵ of the fast process. The resulting system is given as follows:

$$
\begin{cases}
\dot{x}^\epsilon(t) = f(x^\epsilon, y^\epsilon) - 2\dfrac{\langle f(x^\epsilon, y^\epsilon), w^\epsilon\rangle}{\langle w^\epsilon, v^\epsilon\rangle} v^\epsilon, & \text{(3.14a)}\\
\dot{y}^\epsilon(t) = \dfrac{1}{\epsilon\lambda} b(x^\epsilon, y^\epsilon) + \dfrac{1}{\sqrt{\epsilon\lambda}}\eta_1(t), & \text{(3.14b)}\\
\dot{z}^\epsilon(t) = \dfrac{1}{\epsilon\lambda} b(x^\epsilon, z^\epsilon) + \dfrac{1}{\sqrt{\epsilon\lambda}}\eta_2(t), & \text{(3.14c)}\\
\gamma\dot{v}^\epsilon(t) = (D_x f(x^\epsilon, y^\epsilon))\, v^\epsilon + f(x^\epsilon, y^\epsilon)\,\langle g(x^\epsilon, y^\epsilon), v^\epsilon\rangle \\
\qquad\qquad - f(x^\epsilon, z^\epsilon)\,\langle g(x^\epsilon, y^\epsilon), v^\epsilon\rangle - \alpha^\epsilon v^\epsilon, & \text{(3.14d)}\\
\gamma\dot{w}^\epsilon(t) = (D_x f(x^\epsilon, y^\epsilon))^{\mathsf{T}}\, w^\epsilon + g(x^\epsilon, y^\epsilon)\,\langle f(x^\epsilon, y^\epsilon), w^\epsilon\rangle \\
\qquad\qquad - g(x^\epsilon, z^\epsilon)\,\langle f(x^\epsilon, y^\epsilon), w^\epsilon\rangle - \beta^\epsilon w^\epsilon, & \text{(3.14e)}
\end{cases}
$$

where η_1 and η_2 are two independent copies of the stochastic noise process $\eta(t)$. The joint fast processes (y^ϵ, z^ϵ) correspond to the equilibrium distribution $\mu_x(\mathrm{d}y) \times \mu_x(\mathrm{d}z)$. As $\epsilon \to 0$, the effective dynamics of (3.14) converge to the averaged system (3.6). To solve (3.14), any standard ODE/SDE solver (e.g., the Euler scheme) can be employed. The same time step size $\widetilde{\Delta}t$ is used for all components of the system. According to [16], a reasonable choice for the step size is $\widetilde{\Delta}t = \Delta t/M$, where M is the number of micro-step used in the HMM.

Adaptive strategy

To reduce variance and improve computational efficiency, a hybrid approach can be adopted: initially applying the SCM and later transitioning to the HMM after an appropriate period. This strategy leverages the strengths of both methods. Below, we outline three additional adaptive methodologies, whose effectiveness is demonstrated in the numerical results section.

1. Averaging the Output
   Define the averaged solution as

   $$\tilde{x}(t) \doteq \frac{1}{t - t_0}\int_{t_0}^{t} x(s)\mathrm{d}s,$$

   where t_0 represents a "burn-in" period to allow the system to stabilize. This technique, widely used in Monte Carlo simulations and stochastic gradient dynamics, reduces the variance of noisy outputs without adding computational overhead. It can be applied to both the HMM (especially with a small sample size M) and SCM. However, its impact is particularly significant for the SCM, where noise tends to dominate.

2. Adaptive Sampling Size in HMM
   The sample size M in the HMM can be adjusted adaptively:

- Initial stage: Start with a small M to quickly locate the approximate solution region.
- Later stages:
  Gradually increase M to reduce variance as noise begins to dominate the error.

A simple adaptive rule for M could be scale it with time, such as $M \propto t^r$, where $r > 0$. This ensures a balance between computational cost and accuracy.

3. Adaptive Reduction of ϵ' in SCM
   In the SCM, the error depends on the small parameter of $\epsilon' := \epsilon\lambda$ and the step size $\widetilde{\Delta}t$. Reducing ϵ' over time helps decrease the deviation of x^ϵ from the true solution associated with F. However, as ϵ' decreases, the step size $\widetilde{\Delta}t$ must also decrease to maintain or improve accuracy. For example, in stochastic optimization, the step size typically decays as $\widetilde{\Delta}t \propto t^{-1}$, or equivalently $\widetilde{\Delta}t_k = \widetilde{\Delta}t_1/\sqrt{k}$ for the k-th iteration. Similarly, ϵ' can be set as $\epsilon'(t) \propto t^{-1+c}$, where $0 \leq c < 1$. This ensures the effective step size $\widetilde{\Delta}t/\epsilon'$ scales as t^{-c}.

To conclude this subsection, we compare the HMM and SCM methods, specifically in their application to find stationary points. For general discussions, refer to [16] and the references therein. Below are the key differences:

1. Different Target Systems
   The HMM and SCM address different systems:
   - HMM solves the effective dynamics (3.6) by simulating its multiscale system (3.11) as $\epsilon \to 0$.
   - SCM solves the pre-asymptotic system (3.14) with an effective parameter $\epsilon' = \epsilon\lambda > 0$.

   As ϵ' decreases towards zero, the SCM solution approaches the HMM solution. However, with a fixed $\epsilon' > 0$, the SCM trajectory may eventually transition between multiple saddle points (if they exist) over time due to its higher noise level.

2. Handling of the Covariance Matrix
   - HMM computes the two expectations in the covariance matrix $\overline{C}$ using a single stream of random samples from μ_x.
   - SCM approximates the double expectation through two independent streams of random samples for the virtual fast process.

   This dual-stream approach extends the traditional notion of seamless coupling, as classic SCM methods for slow-fast systems typically do not involve multiple streams.

3. Computational Costs and Variance

   - HMM requires averaging over a large sample size, which is computationally expensive but effectively reduces variance.
   - SCM does not compute sample averages at each iteration, making it faster but noisier.

3.1.2 MsGAD for Gradient Systems

In this section, we focus on slow-fast systems driven by a potential energy $U(x, y)$ defined over the extended space $\mathcal{X} \times \mathcal{Y}$. These dynamics exhibit gradient dynamics, describesd as:

$$\begin{cases} \dot{X}^\epsilon = -\nabla_x U(X^\epsilon, Y^\epsilon), & \text{(3.15a)} \\ \dot{Y}^\epsilon = -\frac{1}{\epsilon}\nabla_y U(X^\epsilon, Y^\epsilon) + \frac{1}{\sqrt{\epsilon}}\sigma\eta(t), & \text{(3.15b)} \end{cases}$$

where σ is a scalar constant and $\eta(t)$ represents standard white noise. The potential energy $U(x, y)$ governs the dynamics of (X^ϵ, Y^ϵ). The equilibrium measure of the fast process, given a fixed x is

$$\mu_x(\mathrm{d}y) = \rho(x, y)\mathrm{d}y = \frac{1}{Z(x)} e^{-\frac{2}{\sigma^2}U(x,y)}\mathrm{d}y, \tag{3.16}$$

where the normalization factor $Z(x)$ is defined as:

$$Z(x) = \int e^{-\frac{2}{\sigma^2}U(x,y)}\mathrm{d}y.$$

As $\epsilon \to 0$, the dynamics of the slow-fast variable X^ϵ converge to the averaged equation:

$$\dot{\bar{X}} = F(\bar{X}), \tag{3.17}$$

where the effective force $F(x)$ is given by:

$$F(x) = -\int \nabla_x U(x, y)\rho(x, y)\mathrm{d}y.$$

Using the definition of $Z(x)$, the gradient of $\log Z(x)$ becomes:

$$\nabla_x \log Z(x) = \frac{2}{\sigma^2}\int -\nabla_x U(x, y)\rho(x, y)\mathrm{d}y = \frac{2}{\sigma^2}F(x).$$

Thus, the averaged dynamics can be rewritten as a gradient system:

$$\dot{\bar{X}} = -\nabla_x W(\bar{X}), \tag{3.18}$$

where the effective potential $W(x)$ is

$$W(x) = -\frac{\sigma^2}{2}\log Z(x) = -\frac{\sigma^2}{2}\log\left(\int e^{-2U(x,y)/\sigma^2}\mathrm{d}y\right). \tag{3.19}$$

The Hessian matrix of $W(x)$ can be derived as:

$$\nabla_x^2 W(x) = -DF(x), \tag{3.20}$$

which expands to:

$$\nabla_x^2 W(x) = \overline{\nabla_x^2 U}(x) - \frac{2}{\sigma^2}\overline{\nabla_x U \otimes \nabla_x U}(x) + \frac{2}{\sigma^2}\overline{\nabla_x U}(x) \otimes \overline{\nabla_x U}(x). \tag{3.21}$$

The term involving the Fisher information matrix of the invariant measure $\mu_x = \rho(x,y)\mathrm{d}y$ is:

$$-\mathrm{E}_{\mu_x}\left[\nabla_x^2 \log\rho\right] = \frac{4}{\sigma^2}\left(\overline{\nabla_x U \otimes \nabla_x U}(x) - \overline{\nabla_x U}(x) \otimes \overline{\nabla_x U}(x)\right).$$

The general framework for GAD and MsGAD for slow-fast systems simplifies significantly for gradient systems due to the underlying potential-driven dynamics. The reduction in complexity arises from the explicit gradient structure, which naturally governs both the effective dynamics and the associated potential $W(x)$.

3.2 SIMPLIFIED GENTLEST ASCENT DYNAMICS

The GAD for non-gradient systems of the form $\dot{x} = b(x)$, where $b \in \mathbb{R}^d$, relies on two direction variables v and w. These variables are associated with the eigenvectors of the Jacobian matrix $J(x) = Db(x)$ and its transpose $J(x)^\mathsf{T}$. In particular, the dynamics require:

1. The computation of $J(x)v = \lim_{h\to 0}(b(x+hv) - b(x))/h$, which is straightforward using finite difference methods.

2. The computation of $J(x)^\mathsf{T}w$, which lacks a directional derivative interpretation and poses challenges for large-scale problems.

To address these issues, we introduce the Simplified GAD. This approach eliminates the need to compute $J(x)^\mathsf{T}w$ by using only a single direction variable (v or w). This reduction maintains the convergence properties of the original GAD while significantly improving computational efficiency.

The new GAD has two equivalent formulations (not used simultaneously):
Formulation 1: Using v

$$\begin{cases} \dot{x} = b(x) - 2\left\langle b(x), v(t)\right\rangle v(t)/\left\|v(t)\right\|^2, & \text{(3.22a)} \\ \dot{v} = J(x)v - \left\langle v, Jv\right\rangle v, & \text{(3.22b)} \end{cases}$$

Formulation 2: Using w

$$\begin{cases} \dot{x} = b(x) - 2\langle b(x), w(t)\rangle w(t)/\|w(t)\|^2, & \text{(3.23a)} \\ \dot{w} = J^{\mathsf{T}}(x)w - \left\langle w, J^{\mathsf{T}}w \right\rangle w. & \text{(3.23b)} \end{cases}$$

In both cases, the dynamics evolve in $\mathbb{R}^{2d}$, and the direction variable is normalized such that $\|v_0\| = 1$ or $\|w_0\| = 1$, ensuring v and w are remains unit vectors.

Comparison of Formulations The difference between (3.22) and (3.23) lies in the matrix-vector multiplication:

- Formulation (3.22) involves $J(x)v$, which can be computed using finite differences.
- Formulation (3.23) requires $J(x)^{\mathsf{T}}w$, making it computationally less desirable for large systems.

However, formulation (3.23) has theoretical significance, as shown in the Appendix. It can be interpreted as a modified Hamiltonian system (3.61) where flipping the sign of the normalized momentum stabilizes the saddle point.

The simplified GAD converges to the index-1 saddle point of the original dynamics $\dot{x} = b(x)$, as shown in the following theorem.

Theorem 3.1 *(a) Let (x_*, v_*) be a fixed point of the simplified GAD (3.22), where v_* is a normalized vector, $\|v_*\| = 1$. Then:*

- *x_* is a fixed point of the original system, $b(x_*) = 0$.*
- *v_* is an eigenvector of $J(x_*)$ associated with an eigenvalue λ_*:*

$$J(x_*)v_* = \lambda_* v_*.$$

(b) Suppose x_s is a fixed point of $\dot{x} = b(x)$, and $J(x_s)$ has n distinct real eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_n$ with corresponding eigenvectors v_i. Then:

- *Each (x_s, v_i) is a fixed point of the simplified GAD (3.22).*
- *Among these, exactly one fixed point $(x_s, v_{i'})$ is linearly stable if and only if x_s is an index-1 saddle point of $\dot{x} = b(x)$ and $\lambda_{i'} > 0$ is the only positive eigenvalue of $J(x_s)$.*

The proof, presented in Appendix A, follows a linear analysis at the fixed points. While the eigenvalues of the simplified GAD are identical to those of the original GAD, the latter has eigenvalues with multiplicity 2.

Remark 3.2 *1. Time-Scale Adjustment: A positive constant τ can be introduced to scale the dynamics of the direction variable, as in the original GAD. This allows for flexible control over the convergence speed, though τ is omitted here for simplicity.*

2. *For gradient systems where* $\dot{x}(t) = -\nabla V(x)$, $J = -H$ *(the Hessian matrix), formulations* (3.22) *and* (3.23) *are identical. They reduce to the standard GAD for gradient systems* (2.3). *Thus,* (3.22) *is always preferred for both gradient and non-gradient systems.*

3.3 SIMPLIFIED MULTISCALE GAD

The concept of GAD has been extended to the slow-fast stochastic systems, as presented in the MsGAD framework (section 3.1). As a corollary of the simplified GAD discussed earlier, it can be directly applied to such multi-scale models. For background details, readers are encouraged to refer to the original MsGAD paper [41]. In this section, We present the simplified multiscale GAD (MsGAD) scheme, which is designed for identifying saddle points in the slow-fast system:

$$\begin{cases} \dot{X}^\epsilon(t) = f(X^\epsilon, Y^\epsilon), & \text{(3.24a)} \\ \dot{Y}^\epsilon(t) = \dfrac{1}{\epsilon} b(X^\epsilon, Y^\epsilon) + \dfrac{1}{\sqrt{\epsilon}} \sigma(X^\epsilon, Y^\epsilon)\eta(t), & \text{(3.24b)} \end{cases}$$

where ϵ is a small parameter, η represents noisy perturbations, X^ϵ is the slow variable, and Y^ϵ is the fast variable. As $\epsilon \to 0$, the effective dynamics of the slow variable can be expressed as

$$\dot{\bar{X}} = F(\bar{X}), \quad \text{where } F(x) \doteq \int f(x, y)\mu_x(\mathrm{d}y), \tag{3.25}$$

and $\mu_x(\mathrm{d}y)$ is the invariant measure of the fast process, with its density function denoted by $\rho(x, y)$. The simplified MsGAD for finding saddle points in (3.25) can be expressed using either of the following formulations:

1. Formulation 1: Using v^ϵ

$$\begin{cases} \dot{x}^\epsilon(t) = f(x^\epsilon, y^\epsilon) - 2\dfrac{\langle f(x^\epsilon, y^\epsilon), v^\epsilon\rangle}{\langle v^\epsilon, v^\epsilon\rangle} v^\epsilon, & \text{(3.26a)} \\ \dot{y}^\epsilon(t) = \dfrac{1}{\epsilon} b(x^\epsilon, y^\epsilon) + \dfrac{\sigma}{\sqrt{\epsilon}}\eta(t), & \text{(3.26b)} \\ \dot{v}^\epsilon(t) = \left(D_x f(x^\epsilon, y^\epsilon) + C(x^\epsilon, y^\epsilon)\right) v^\epsilon - \alpha^\epsilon v^\epsilon, & \text{(3.26c)} \end{cases}$$

2. Formulation 2: Using w^ϵ

$$\begin{cases} \dot{x}^\epsilon(t) = f(x^\epsilon, y^\epsilon) - 2\dfrac{\langle f(x^\epsilon, y^\epsilon), w^\epsilon\rangle}{\langle w^\epsilon, w^\epsilon\rangle} w^\epsilon, & \text{(3.27a)} \\ \dot{y}^\epsilon(t) = \dfrac{1}{\epsilon} b(x^\epsilon, y^\epsilon) + \dfrac{\sigma}{\sqrt{\epsilon}}\eta(t), & \text{(3.27b)} \\ \dot{w}^\epsilon(t) = \left(D_x f(x^\epsilon, y^\epsilon) + C(x^\epsilon, y^\epsilon)\right)^{\mathsf{T}} w^\epsilon - \beta^\epsilon w^\epsilon. & \text{(3.27c)} \end{cases}$$

Here, $D_x f(x,y)$ is the Jacobian matrix of $f(x,y)$ with respect to x. Other terms are defined as follows:

$$\begin{aligned}
\alpha^\epsilon &= \langle v^\epsilon, (D_x f + C)\, v^\epsilon \rangle\,, \quad \beta^\epsilon = \left\langle w^\epsilon, (D_x f + C)^\mathsf{T}\, w^\epsilon \right\rangle, \\
C(x,y) &= (f(x,y) - F(x)) \otimes (g(x,y) - G(x))\,, \\
g(x,y) &= -\nabla_x U(x,y), \quad U(x,y) = -\log \rho(x,y), \\
G(x) &= \int g(x,y)\mu_x(dy).
\end{aligned}$$

In general, the covariance function C lacks a closed form and must be computed dynamically using methods such as the heterogeneous multiscale method (HMM) or the seamless coupling method[41].

Consider the following example:

$$\begin{cases}
\dot{X}_i = -\sum_{j=1}^{2} A_{ij} X_j + Y_i^2, \quad i = 1,2, & \text{(3.28a)} \\
\dot{Y}_i = -\frac{1}{\epsilon}\frac{Y_i}{\Gamma_i(X)} + \frac{\sigma}{\sqrt{\epsilon}}\eta_i(t), & \text{(3.28b)}
\end{cases}$$

where A is a 2 by 2 matrix, $\Gamma_i(x)$ are positive-valued functions, and σ is a constant. For $\epsilon \to 0$, the effective dynamics are given by:

$$\dot{\bar{X}}_i = -\sum_{j=1}^{2} A_{ij}\bar{X}_j + \frac{\sigma^2}{2}\Gamma_i(\bar{X}) =: F_i(\bar{X}), i = 1,2. \tag{3.29}$$

The Jacobian matrix $J(x) = DF(x)$ has the form:

$$J(x) = -A + \frac{1}{2}\sigma^2 \mathrm{diag}\left\{\Gamma_i'(x) : i = 1,2\right\}.$$

The covariance $C(x,y)$ for (3.26) is given by:

$$C_{ij}(x,y) = \frac{\Gamma_j'(x)}{\Gamma_j^2(x)}\left(\frac{y_i^2 y_j^2}{\sigma^2} - \frac{1}{2}y_i^2\Gamma_j(x) - \frac{1}{2}y_j^2\Gamma_i(x) + \frac{\sigma^2}{4}\Gamma_i(x)\Gamma_j(x)\right).$$

By averaging $C(x,y)$ with respect to $\mu_x(dy)$, we find:

$$\bar{C} = \int C(x,y)\mu_x(dy) = \frac{1}{2}\sigma^2 \mathrm{diag}\left\{\Gamma_i'(x) : i = 1,2\right\}.$$

and

$$\int D_x f(x,y)\mu_x(dy) = -A.$$

This demonstrates that the effective dynamics of MsGAD (3.26) reduce to the GAD for the effective equation (3.29).

3.4 PROJECTED GAD

Fixed points of a system vary across different potential energy surfaces and metrics. For example, the Ginzburg-Landau free energy leads to the Allen-Cahn equation in the L^2 metric and the Cahn-Hilliard equation in the H^{-1} metric. Notably, dynamics in the H^{-1} metric involve two higher orders of spatial derivatives compared to L^2 metric, causing severe numerical challenges due to ill-conditioning.

When applying GAD or MsGAD for gradient systems in the H^{-1} metric, similar difficulties arise. A key distinction between these metrics lies in mass conservation: the H^{-1} metric preserves mass, whereas L^2 does not. To address this, we incorporate a projection operator to enforce mass conservation in the L^2 metric dynamics, bridging the relationship between fixed points in the two metrics.

This section introduces a projected GAD method for efficiently locating saddle point in the H^{-1} metric. The GAD for extended gradient systems in the H^{-1} metric is given as:

$$\begin{cases} \dfrac{\partial \phi}{\partial t} = \Delta \delta_\phi F(\phi) + 2\dfrac{\langle \delta_\phi F(\phi), v\rangle_{L^2}}{\langle v, v\rangle_{L^2}} v, & \text{(3.30a)} \\ \gamma \dfrac{\partial v}{\partial t} = \widetilde{\mathbf{H}} v + \left\langle v, \delta_\phi^2 F(\phi) v \right\rangle_{L^2} v, & \text{(3.30b)} \end{cases}$$

where $\widetilde{\mathbf{H}} = \Delta \delta_\phi^2 F(\phi)$. We define the projection operator $\mathbf{P}$ to impose the mass conservation constraint:

$$\mathbf{P}u := u - \frac{1}{|\Omega|}\int_\Omega u(x)\,\mathrm{d}x, \tag{3.31}$$

where $|\Omega|$ is the domain volume. The operator $\mathbf{P}$ projects u onto the subspace $H_0 = \{u \in L^2 : \int_\Omega u(x)\mathrm{d}x = 0\}$. Its key properties include:

1. $\mathbf{P}^2 = \mathbf{P}$;
2. $\mathbf{P}u \in H_0, \forall u \in L^2$;
3. $\mathbf{P}v = v, \forall v \in H_0$;
4. $\langle v, \mathbf{P}w\rangle_{L^2} = \langle \mathbf{P}v, w\rangle_{L^2}, \forall v \in L^2$ and $\forall w \in H_0$.

In fact, for $\forall u$,

$$\begin{aligned} \mathbf{P}^2 u &= \mathbf{P}(u - \frac{1}{|\Omega|}\int_\Omega u(x)\,dx) \\ &= u - \frac{1}{|\Omega|}\int_\Omega u(x)\,dx - \frac{1}{|\Omega|}\int_\Omega (u(x) - \frac{1}{|\Omega|}\int_\Omega u(x)\,dx)\,dx \\ &= \mathbf{P}u. \end{aligned}$$

Besides, $\mathbf{P}u \in H_0$ since

$$\int_\Omega \mathbf{P}u \, \mathrm{d}x = \int_\Omega (u - \frac{1}{|\Omega|} \int_\Omega u(x) \, \mathrm{d}x) \, \mathrm{d}x = 0, \quad \forall u \in L^2. \tag{3.32}$$

The third property is obvious. For the last one, when $w \in H_0, v \in L^2$, we have

$$\langle v, \mathbf{P}w \rangle_{L^2} - \langle \mathbf{P}v, w \rangle_{L^2} = \langle v, w \rangle_{L^2} - \langle \mathbf{P}v, w \rangle_{L^2} = \langle v - \mathbf{P}v, w \rangle_{L^2}.$$

Since $\mathbf{P}v \in H_0, v \in L^2$ and $\mathbf{P}$ is the projection from L^2 to H_0, we obtain

$$v - \mathbf{P}v \in H_0^\perp \quad \Rightarrow \quad \langle v - \mathbf{P}v, w \rangle_{L^2} = 0,$$

thus,

$$\langle v, \mathbf{P}w \rangle_{L^2} = \langle \mathbf{P}v, w \rangle_{L^2}.$$

With the projection operator $\mathbf{P}$, the following equivalence can be get

$$\left\langle v, \widetilde{\mathbf{H}}v \right\rangle_{H^{-1}} = \langle v, -\Delta(\mathbf{H}v) \rangle_{H^{-1}} = \left\langle v, -(-\Delta)^{-1}\Delta(\mathbf{H}v) \right\rangle_{L^2} = \langle v, \mathbf{H}v \rangle_{L^2}.$$

By the last two properties of $\mathbf{P}$, we have

$$\langle v, \mathbf{H}v \rangle_{L^2} = \langle \mathbf{P}v, \mathbf{H}\mathbf{P}v \rangle_{L^2} = \langle v, \mathbf{P}\mathbf{H}\mathbf{P}v \rangle_{L^2}, \quad v \in H_0. \tag{3.33}$$

By incorporating $\mathbf{P}$, the GAD dynamics in the H^{-1} metric can be equivalently reformulated in L^2 metric with projection. The projected GAD equations are:

$$\begin{cases} \dfrac{\partial \phi}{\partial t} = -\mathbf{P}\delta_\phi F(\phi) + 2\dfrac{\langle \delta_\phi F(\phi), v \rangle_{L^2}}{\langle v, v \rangle_{L^2}} \mathbf{P}v, & \text{(3.34a)} \\ \gamma \dfrac{\partial v}{\partial t} = -\mathbf{P}\delta_\phi^2 F(\phi)\mathbf{P}v + \left\langle v, \delta_\phi^2 F(\phi) v \right\rangle_{L^2} v. & \text{(3.34b)} \end{cases}$$

The projection $\mathbf{P}$ ensures the dynamics preserve mass conservation by confining ϕ and v to H_0. Moreover, it simplifies the computation of saddle points by avoiding higher-order spatial derivatives.

To confirm the equivalence of fix points between the H^{-1} and L^2 metrics, Consider $\mu_1 = \delta_\phi F(\phi)$ and $\mu_2 = \Delta\mu_1$. With periodic or zero-flux boundary conditions:

$$\mu_2 = 0 \Leftrightarrow \mathbf{P}\mu_1 = 0.$$

Thus, the projected dynamics in L^2 metric correctly capture the fixed points of the original H^{-1} metric dynamics.

3.5 NUMERICAL EXAMPLES

In this subsection, we validate the variants of the gentlest ascent dynamics (GAD) through several numerical examples.

Numerical Validation of MsGAD

To demonstrate the functionality of the MsGAD, we present two numerical examples, which were originally published in [41]. The first example involves a system with both a two-dimensional ordinary differential equation (ODE) and a two-dimensional stochastic ODE, which does not exhibit a gradient structure. The HMM is used within the MsGAD framework for this system. The second example features an Allen–Cahn partial differential equation (PDE) coupled with a stochastic Allen–Cahn PDE, which is associated with an extended potential functional. In this case, we compare the performance of both the HMM and the SCM within the MsGAD.

A two-dimensional example Consider the following system on $\mathcal{X} \times \mathcal{Y} = \mathbb{R}^2 \times \mathbb{R}^2$

$$\begin{cases} \dot{X}_i = -\sum_j D_{ij} X_j + Y_i^2, & \text{(3.35a)} \\ \dot{Y}_i = -\frac{1}{\epsilon} \frac{Y_i}{\Gamma_i(X)} + \frac{1}{\sqrt{\epsilon}} \sigma \eta(t), & \text{(3.35b)} \end{cases}$$

where the vector field $\Gamma(x) = (\Gamma_1(x), \Gamma_2(x)) : \mathcal{X} \to \mathcal{Y}$ is given. The matrix $D = (D_{ij})$ is a symmetric 2×2 matrix, and σ is a constant. The processes $\{Y_i(t)\}$ are independent Ornstein-Uhlenbeck processes parameterized by $X = x$. The equilibrium distribution of $Y = (Y_1, Y_2)$ is the product measure of $\mathcal{N}(0, \sigma^2 \Gamma_i(x)/2)$. The limit equation of this system has a closed-form solution:

$$\dot{\bar{X}}_i = -\sum_j D_{ij} \bar{X}_j + \frac{\sigma^2}{2} \Gamma_i(\bar{X}). \tag{3.36}$$

For the case where D is positive-definite, we can rewrite (3.35) as

$$\begin{cases} \dot{X} = -\nabla_X (\frac{1}{2} X^\mathsf{T} D X - \sum_i Y_i^2 X_i) & \text{(3.37a)} \\ \dot{Y} = -\nabla_Y (\sum_i \frac{Y_i^2}{2\Gamma_i(X)}), & \text{(3.37b)} \end{cases}$$

however, even in this positive-definite case, there is no single potential for the system (3.35) in the extended space $\mathcal{X} \times \mathcal{Y}$ for any choice of Γ. This means that we cannot determine *a priori* whether the averaging system is gradient-based. The analytical form of the averaged system (3.36) (with a positive-definite D) reveals that it is a gradient system if and only if the vector field Γ is a gradient itself, i.e., there exists a scalar function R such that

$$\Gamma(x) = (\Gamma_1(x), \ldots, \Gamma_n(x)) = \nabla_x R(x). \tag{3.38}$$

This implies that while the existence of an extended potential function is a sufficient condition for averaged dynamics to be gradient-based, it is not a necessary condition.

We now proceed to present the numerical results for the MsGAD applied to this example. To begin, we must solve for both directions $v(t)$ and $w(t)$ in the MsGAD scheme because, as noted, we cannot assume that the Jacobian of the potential function, DF, is symmetric *a priori*. To validate our approach, we not only observe the convergence to the saddle point but also compare the trajectory $x(t)$ in the MsGAD with that of the classic GAD applied directly to the known limit equation (3.36). For the numerical tests, we use the following parameters:

$$\sigma^2 = 10, \quad D = \begin{bmatrix} 0.8 & -0.2 \\ -0.2 & 0.5 \end{bmatrix},$$

and for $x = (x_1, x_2)$,

$$R(x) = \sum_i \arctan(x_i - 5), \quad \Gamma_i(x) = \left(1 + (x_i - 5)^2\right)^{-1}, \ i = 1, 2.$$

The averaged equation (3.36) then becomes

$$\dot{\bar{X}} = -\nabla W(\bar{X}), \quad \text{where } W(x) = \frac{1}{2} x^{\mathsf{T}} D x - \frac{\sigma^2}{2} R(x). \tag{3.39}$$

This system has three local minima: $m_1 = (0.4643, 0.6985), m_2 = (2.2038, 5.9804)$ and $m_3 = (5.7109, 6.2369)$, and two saddle points: $s_1 = (1.2841, 3.4483)$ and $s_2 = (3.5689, 6.0735)$, shown as in Fig. 3.1.

In the HMM scheme of the MsGAD, we use the forward Euler solver for the entire system. The macro-time step size is $\Delta t = 0.01$ for both x^ϵ and v^ϵ, w^ϵ, and we set $\tau = 1.0$ and $K = 1$ for this example. The micro-time step size is $\delta t = \epsilon \times 0.01$, and the total sampling time is $T = 10$ to estimate the effective force and the Jacobian matrix. The initial values for x are set at the three local minima, and the initial values for the directions (v, w) and the fast processes are chosen arbitrarily. Fig. 3.1 displays the four GAD trajectories of the $x-$component (dashed lines) starting from the three local minima. This result has been published in [41]. Depending on the initial conditions for x, these trajectories converge to different neighboring saddle points. Two of the trajectories, starting from m_2, converge to the saddle points s_1 and s_2, respectively, due to different initial values for the directions v and w.

A coupled Allen-Cahn system The second example examines a system of stochastic partial differential equations (SPDEs) governing $(u^\epsilon(x,t), \phi^\epsilon(x,t))$ in the Hilbert space $L^2([0,1])$. The system satisfies coupled Allen-Cahn-type equations with Neumann boundary conditions:

$$\begin{cases} \partial_t u^\epsilon = \kappa^2 \Delta u^\epsilon + u^\epsilon - (u^\epsilon)^3 + \mu \phi^\epsilon, & \text{(3.40a)} \\ \partial_t \phi^\epsilon = \dfrac{1}{\epsilon}[\Delta \phi^\epsilon - \phi^\epsilon + \mu u^\epsilon] + \dfrac{\sigma}{\sqrt{\epsilon}} \dot{W}_t(t). & \text{(3.40b)} \end{cases}$$

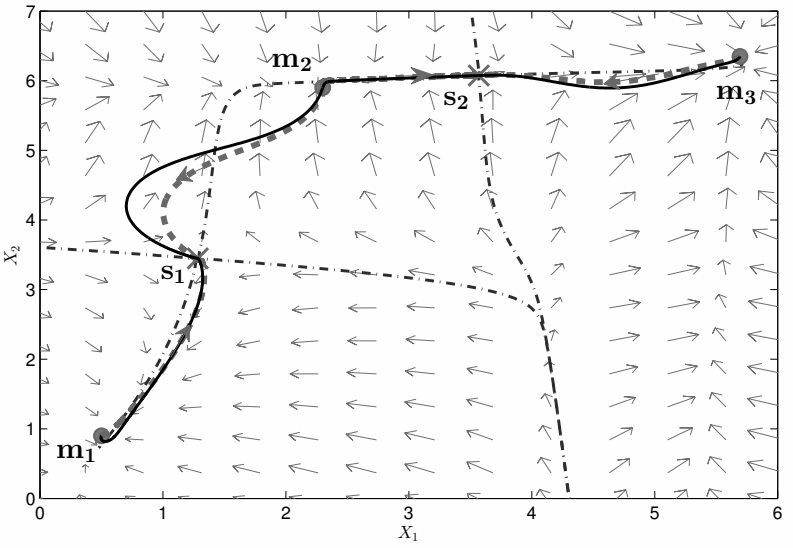

Figure 3.1 GAD trajectories from three different local minima (m_1, m_2, and m_3) to two different saddle points (s_1 and s_2). The flow indicated by the arrows corresponds to the averaged gradient system (3.39). The dash-dotted curves represent the stable/unstable manifolds of the two saddle points, determining the basin boundaries of the three local wells. The thick dashed curves with arrows marked represent the trajectories of MsGAD by the HMM. For comparison, the thin solid curves show the trajectories of the GAD directly applied to the closed form (3.39) of the averaged system.

Here κ is the diffusion coefficient for the slow dynamics, μ is the coupling constant between the slow and the fast dynamics, and $W_t(t)$ is an $L^2([0,1])$-valued Wiener process with a positive-definite (spatial) covariance operator Q. The term $\dot{W}_t$ represents white noise in time, and σ is the noise intensity. The Laplacian operator is given by $\Delta = \partial_x^2$. For any fixed $u^\epsilon = u$, the equilibrium distribution of the SPDE (3.40b) is the Gaussian measure $\mathcal{N}(\mu(I-\Delta)^{-1}u, \sigma^2(I-\Delta)^{-1}Q/2)$ on the Cameron–Martin space. Taking Q as the identity operator (formally treating $\dot{W}_t$ as spatio-temporal white noise), the averaged equation for the limiting solution $\bar{u}$ becomes

$$\begin{cases} \partial_t \bar{u} = \kappa^2 \Delta \bar{u} + \bar{u} - \bar{u}^3 + \mu^2 (I-\Delta)^{-1} \bar{u}, & \text{(3.41a)} \\ \left.\dfrac{\partial \bar{u}}{\partial \vec{n}}\right|_{x=0} = 0, \quad \left.\dfrac{\partial \bar{u}}{\partial \vec{n}}\right|_{x=1} = 0. & \text{(3.41b)} \end{cases}$$

An energy functional $U(u,\phi)$ jointly defined for the pair (u,ϕ) is given by:

$$U(u,\phi) = \int_\Omega \frac{\kappa^2}{2} u_x^2 + \frac{1}{4}(u^2-1)^2 - \mu u \phi + \frac{1}{2}\phi_x^2 + \frac{1}{2}\phi^2 \, \mathrm{d}x, \qquad (3.42)$$

indicating that (3.40) is a gradient system. Therefore, the MsGAD formulation only modifies one direction:

$$\begin{cases} \partial_t u^\epsilon = -\delta_u U(u^\epsilon, \phi^\epsilon) + 2\dfrac{\langle \delta_u U(u^\epsilon, \phi^\epsilon), v^\epsilon \rangle}{\langle v^\epsilon, v^\epsilon \rangle} v^\epsilon, & \text{(3.43a)} \\ \partial_t \phi^\epsilon = -\dfrac{1}{\epsilon}\delta_\phi U(u^\epsilon, \phi^\epsilon) + \dfrac{\sigma}{\sqrt{\epsilon}}\dot{W}_t(t), & \text{(3.43b)} \\ \partial_t v^\epsilon = -\delta_u^2 U(u^\epsilon, \phi^\epsilon) v^\epsilon + C v^\epsilon - \alpha^\epsilon v^\epsilon. & \text{(3.43c)} \end{cases}$$

Here $\delta_u U$ and $\delta_\phi U$ are the first-order Fréchet derivatives of $U(u, \phi)$ with respect to u and ϕ, respectively, while $\delta_u^2 U$ is the Hessian matrix. The operator C is defined as: $C = -\frac{2}{\sigma^2}\delta_u U \otimes \delta_u U + \frac{2}{\sigma^2}\overline{\delta_u U} \otimes \overline{\delta_u U}$.

In the numerical tests, parameters are set to $\kappa = 0.01$, $\sigma = 0.3$, and $\mu = 1$. In this configuration, $u \equiv \pm 1.4142$ are two local minima of the effective dynamics. When $\mu = 0$, the minima shift to $u \equiv \pm 1$. The corresponding saddle points for these cases are shown in Fig. 3.2 [41].

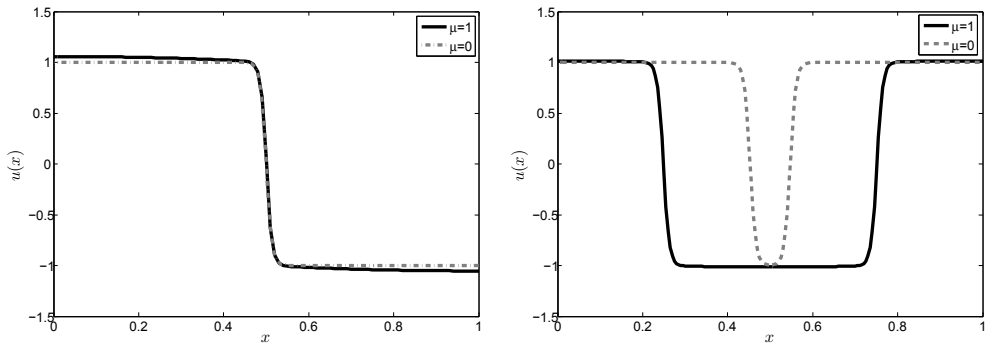

Figure 3.2 Two types of index-1 saddle points for $\mu = 0$ (dashed lines) and $\mu = 1$ (solid lines). The saddle point in **a** has a lower energy than the saddle point in **b**.

The MsGAD system (3.43) is solved using both the HMM and SCM. The HMM applies the convex splitting method for time discretization of (3.43a), ensuring a large time step Δt. Spatial discretization uses the central finite difference method with a uniform mesh size $\Delta x = 1/200$. The term $\delta_u^2 U(u, \phi)v$ in (3.43) is approximated using finite differences:

$$\delta_u^2 U(u, \phi)v \approx \frac{1}{h}[\delta_u U(u + hv, \phi) - \delta_u U(u, \phi)],$$

where $h = 0.001$; For SCM, the boosted system modifies (3.43) by coupling additional variables and noise terms:

$$
\begin{cases}
\partial_t u^\epsilon = -\delta_u U(u^\epsilon, \phi^\epsilon) + 2\dfrac{\langle \delta_u U(u^\epsilon, \phi^\epsilon), v^\epsilon \rangle}{\langle v^\epsilon, v^\epsilon \rangle} v^\epsilon, & \text{(3.44a)} \\
\partial_t \phi^\epsilon = -\dfrac{1}{\epsilon\lambda}\delta_\phi U(u^\epsilon, \phi^\epsilon) + \dfrac{\sigma}{\sqrt{\epsilon\lambda}}\dot{W}_{t1}(t), & \text{(3.44b)} \\
\partial_t \psi^\epsilon = -\dfrac{1}{\epsilon\lambda}\delta_\psi U(u^\epsilon, \psi^\epsilon) + \dfrac{\sigma}{\sqrt{\epsilon\lambda}}\dot{W}_{t2}(t), & \text{(3.44c)} \\
\partial_t v^\epsilon = -\delta_u^2 U(u^\epsilon, \phi^\epsilon) v^\epsilon + f(u^\epsilon, \phi^\epsilon)\langle g(u^\epsilon, \phi^\epsilon), v^\epsilon \rangle \\
\qquad\qquad - f(u^\epsilon, \psi^\epsilon)\langle g(u^\epsilon, \phi^\epsilon), v^\epsilon \rangle - \alpha^\epsilon v^\epsilon. & \text{(3.44d)}
\end{cases}
$$

Here W_{t1} and W_{t2} are two independent and identically distributed copies of W_t, and $\langle \cdot, \cdot \rangle$ denotes the inner product in $L^2([0,1])$.

Define the errors as follows,

$$\mathrm{err}_H(t) = \|u_H(x,t) - u^*(x)\|_{L^2}, \qquad \mathrm{err}_S(t) = \|u_S(x,t) - u^*(x)\|_{L^2},$$

where $u_H(x,t)$ is the solution of (3.43) obtained using the HMM, and $u_S(x,t)$ is the solution of (3.44) obtained using the SCM. The true solution, $u^*(x)$, is computed using the classic GAD applied to the closed form of the averaged system (3.41), with a very fine time step size and a sufficiency small tolerance.

To evaluate the efficiency of the HMM and SCM, we tested the saddle point with the lowest energy (as shown in Fig. 3.2) using the initial condition $u_0(x) = \cos(\pi x)$.

In the HMM, the macro-time step size for u^ϵ and v^ϵ is set to $\Delta t = 0.025$, while the micro-time step size for ϕ^ϵ is $\delta t = 0.01 \times \epsilon$. To examine the effect of the sample size M, simulation were performed with varying M values, running until the error reduced to 10^{-4}. The errors are plotted against both physical time and the number of force evaluations (including direction calculations) in Fig. 3.3 [41]. A smaller sample size M leads to larger and earlier fluctuations in the error, as seen in Fig. 3.3**a**. However, smaller M significantly reduces computational costs, as evident from Fig. 3.3**b**. While further reductions in M are possible to minimize costs, achieving an error as small as 10^{-4} requires either switching to a larger M near the saddle point or employing time-averaging to reduce variance.

For the standard SCM, we used $\widetilde{\Delta}t = 1.0 \times 10^{-2}$ for all components, with $\epsilon = 1.0 \times 10^{-4}$ and $\lambda = 10$ (effectively, $\epsilon' = \epsilon\lambda = 1.0 \times 10^{-3}$). In the adaptive SCM, these parameters ($\widetilde{\Delta}t$ and ϵ') remained constant until time t_a, when the error err_S reached a threshold of 10^{-2}. From t_a, the time step size $\widetilde{\Delta}t = k^{-1/2}\widetilde{\Delta}t$ and $\epsilon' = k^{-1/2}\epsilon'$ were progressively reduced, where k denotes the number of time steps since t_a. As described in section 3.1.1, the SCM's final output is the time-averaged solution:

$$\tilde{u}(x,t) \doteq \frac{1}{t - t_0}\int_{t_0}^{t} u(x,s)\mathrm{d}s,$$

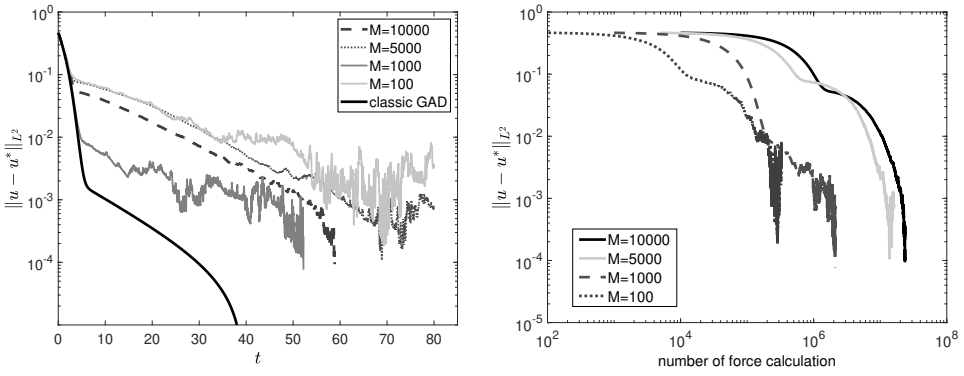

Figure 3.3 **a** shows the evolution of the error with the physical time for the classic GAD applied to the effective dynamics (3.41) and for the HMM to (3.43) with various sample sizes $M = 1000$ and 10000; **b** is the decay of the errors with the number of force calculations.

where $t_0 = 10$. This continuous-time integral is implemented iteratively using discrete time points.

Fig. 3.4 demonstrates the evolution of the error in the SCM [41]. For a fixed $\epsilon' = 10^{-3}$, the error plateaued near 10^{-3}. However, when ϵ' was reduced gradually, the error decreased further, eventually reaching 10^{-5}. This improvement,

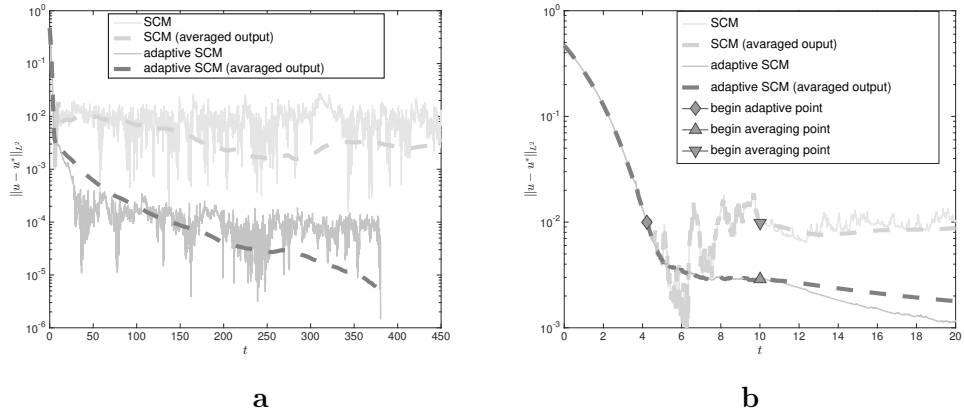

Figure 3.4 The evolution of the error with the time in the seamless coupling method. **b** is the same plot as **a** in the early stage before time $t = 20$. The solutions are averaged w.r.t. time after the burn-in time $t_0 = 10$, which are shown by the two triangular symbols in the plot. The resulted new outputs are shown in dashed thick lines. The value of ϵ' and the step size in the adaptive SCM start to decay when the error decreases to 0.01, which is shown by the diamond symbol in the right plot. The "SCM" means the standard SCM with a fixed ϵ' and a constant time step size.

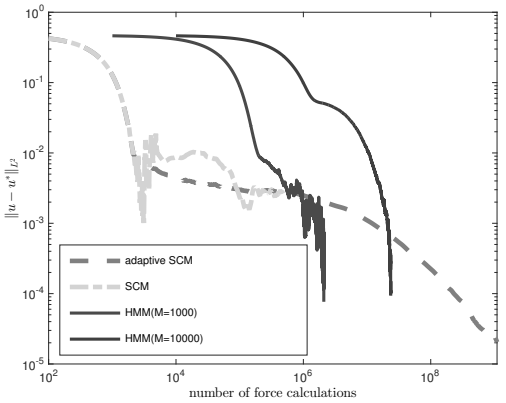

Figure 3.5 The illustration of the computational costs for the SCM, the adaptive version of SCM (both with averaged output), and the HMM (the line with noticeable oscillations corresponds to $M = 1000$; the other line represents $M = 10000$). The plot is presented on a log-log scale.

however, comes at the cost of increased stiffness, necessitating smaller time steps. The log-log plot in Fig. 3.5 highlights the rapid growth in computational cost as ϵ' decreases.

Fig. 3.5 also compares the computational performance of the SCM, adaptive SCM, and HMM [41]. While the SCM is cost-effective in quickly reaching regions of interest, it becomes computationally expensive for small ϵ'. Conversely, the HMM is more suited for precise computations near the saddle point. Based on these findings, a practical strategy might involve starting with a boosted ϵ' in the adaptive SCM to efficiently locate the region of interest, followed by switching to the HMM with an adaptive choice of M. The optimal selection of parameters for both methods remains an open question of practical and theoretical interest, warranting further investigation.

Numerical validation for simplified GAD

A two-dimensional example The first test aims to locate the saddle point of the following two-dimensional non-gradient system:

$$\dot{x}_i = -\frac{1}{2}\sum_{j=1} A_{ij}x_j + \frac{\sigma^2}{2}\Gamma_i(x), \quad i = 1, 2, \tag{3.45}$$

where

$$\sigma^2 = 10, A = \begin{bmatrix} 0.8 & -0.3 \\ -0.2 & 0.5 \end{bmatrix}, \Gamma_i(x) = \left(1 + (x_i - 5)^2\right)^{-1}, \ i = 1, 2.$$

This system has two stable fixed points:

$$m_1 = (0.5931, 0.7655), m_2 = (5.8770, 6.2507)$$

and a unique saddle point $s = (1.7954, 3.3088)$. The simplified GAD trajectories of the $x-$component (solid lines) starting from m_1 and m_2, are shown in Fig. 3.6 [43]. The solid lines represent the trajectories, computed using initial direction vectors $v = [1, 0]$ and $[0, 1]$, respectively.

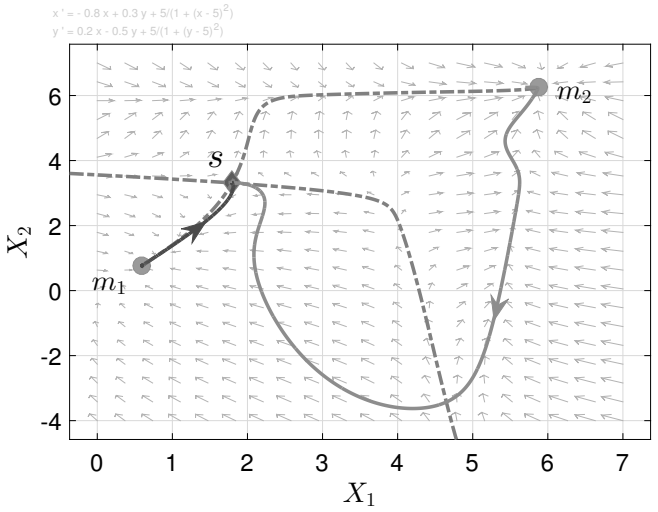

Figure 3.6 Simplified GAD trajectories of the $x-$ component from two locally stable fixed points (m_1 and m_2) to the saddle point s. The dark and the grey solid curves represent the trajectories with initial vector $v = [1, 0]$ and $[0, 1]$, respectively. The arrows indicate the flow of the non-gradient system (3.45). The dash-dotted curves represent the stable/unstable manifolds of the saddle point, where the stable manifold is the basin boundary between m_1 and m_2.

Equation (3.45) corresponds to the effective dynamics of the slow-fast system (3.28), as discussed in section 3.3. The second test applies the simplified MsGAD (3.26) to locate the saddle point of (3.45), using the same setting for A, Γ_i and σ. When the HMM is employed, the resulting trajectories closely match those in Fig. 3.6. Therefore, we focus here on the results obtained using the seamless coupling method, which requires selecting a sufficiently small number ϵ. Fig. 3.7 illustrates the x component trajectories computed with the seamless coupling method, starting from m_1 and m_2 and converging to the saddle point s [43]. The initial direction vectors v are identical to those used in Fig. 3.6.

Nucleation in the presence of shear flow In this example, we tackle a more challenging problem: nucleation in a reaction-diffusion system under the influence of shear flow. Nucleation is a significant physical phenomenon [52, 79, 69, 84, 109], often characterized by the saddle point of the Ginzburg-Landau free energy. In gradient systems driven solely by free energy, the string method [19, 22] is typically used to compute the minimum energy path. However, when shear flow is introduced, the system becomes non-gradient,

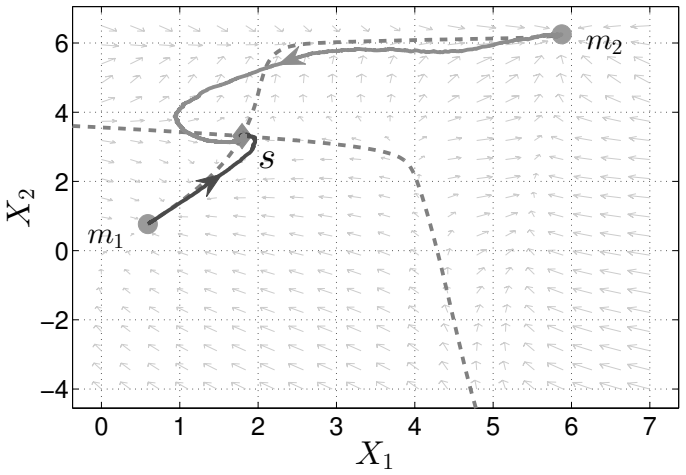

Figure 3.7 Simplified MsGAD trajectories computed using the seamless coupling method. The small parameter ϵ is 10^{-5}. The initial direction vectors v and annotations follow the same conventions as in Fig. 3.6.

necessitating the use of the minimum action method [20] to compute the minimum action path and identify the saddle point along it. By contrast, our simplified GAD directly calculates the saddle point without requiring the entire path.

The Ginzburg-Landau free energy of the order parameter $\phi(x, y)$ is given by:

$$E(\phi) = \int_\Omega \frac{\kappa}{2}|\nabla\phi|^2 + \frac{1}{4}(1-\phi^2)^2 \, \mathrm{d}x\mathrm{d}y, \tag{3.46}$$

where $\kappa = 0.01$, the domain $\Omega = [0,1]\times[0,1]$ has periodic boundary conditions. We consider two cases of the shear flow, illustrated in Fig. 3.8 [43], leading to the following dynamics of the Allen-Cahn equation:

1. Shear flow in the $x-$ direction:

$$\partial_t\phi = -\frac{\delta E}{\delta\phi} + \gamma\sin(2\pi y)\partial_x\phi, \tag{3.47}$$

2. Shear flow in both direction:

$$\partial_t\phi = -\frac{\delta E}{\delta\phi} + \gamma\sin(2\pi y)\partial_x\phi + \gamma\sin(2\pi x)\partial_y\phi, \tag{3.48}$$

where γ is the shear rate, and the Fréchlet derivative of E is $\delta_\phi E = -\kappa\Delta\phi - \phi + \phi^3$.

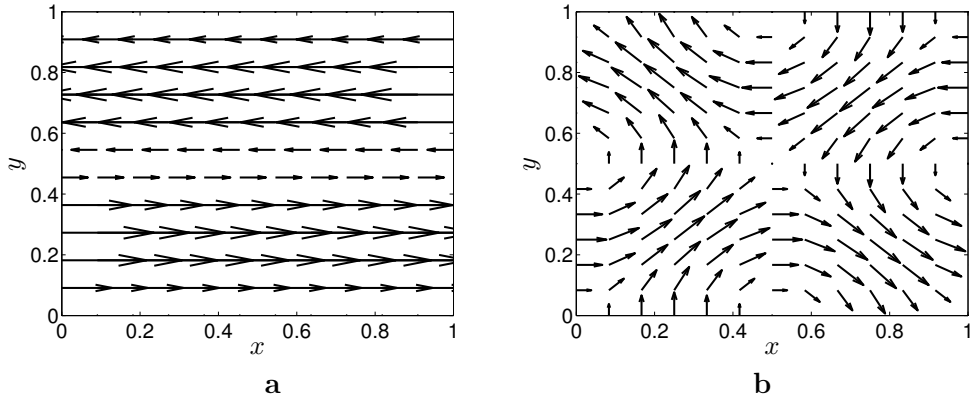

Figure 3.8 Shear flow vector fields used in equations (3.47) (left) and (3.48) (right).

We aim to locate the index-1 saddle point $\phi(x, y)$ of the dynamics (3.47) and (3.48) using the simplified GAD. Denoting the right-hand side of these equations by $b(\phi)$, the corresponding GAD system is:

$$\begin{cases} \partial_t \phi = b(\phi) - 2\langle b(\phi), v\rangle v / \|v\|^2, & \text{(3.49a)} \\ \partial_t v = Db(\phi)v - \langle v, (Db)v\rangle v / \|v\|^2, & \text{(3.49b)} \end{cases}$$

where $v = v(x, t)$, $\langle \cdot, \cdot \rangle$ is the L^2 inner product, and $\|\cdot\|$ is the L^2 norm.

Remark 3.3 *Here the dynamics is a PDE model and we can have the analytical expression for the Jacobian and its transpose. We take the case in equation* (3.47)*, for example, to show Db and its adjoint* $(Db)^T$*.* $b(\phi) = \kappa\Delta\phi + \phi - \phi^3 + \gamma \sin(2\pi y)\partial_x\phi$*.* $Db(\phi)v = \kappa\Delta v + v - 3\phi^2 v + \gamma \sin(2\pi y)\partial_x v$*. Then* $(Db(\phi))^T w = \kappa\Delta w + w - 3\phi^2 w - \gamma \sin(2\pi y)\partial_x w$ *since the adjoint of* ∂_x *is* $-\partial_x$*. This example shows that when b is a differential operator, one may obtain the "transpose" (adjoint) of the Jacobian analytically. Then the two forms of the simplified GAD* (3.22) *and* (3.23) *are both applicable in such cases.*

In numerical tests, we use the mesh points with $N_x = N_y = 128$ in the finite difference method for spatial discretization. The two metastable states are $\phi \equiv 1$ and $\phi \equiv -1$ regardless of the shear flow. By solving the simplified GAD (3.49), we shall get different index-1 saddle points for various γ. We are interested in the effect of the shear rate on the profiles of saddle points. It is noted that the steady states for any shear flow preserve the symmetry $\phi \to -\phi$ and equation (3.48) additionally preserves another symmetry $\phi(x, y) \to \phi(y, x)$. So there are multiple symmetric images of one steady state. All of our plots below display only one of the symmetric images.

For the dynamics in (3.47), the shear force acts only in the $x-$direction. As the shear rate γ increases, the saddle point profiles are shown in Fig. 3.9 [43]. We have the following interesting observation from this figure: the profiles

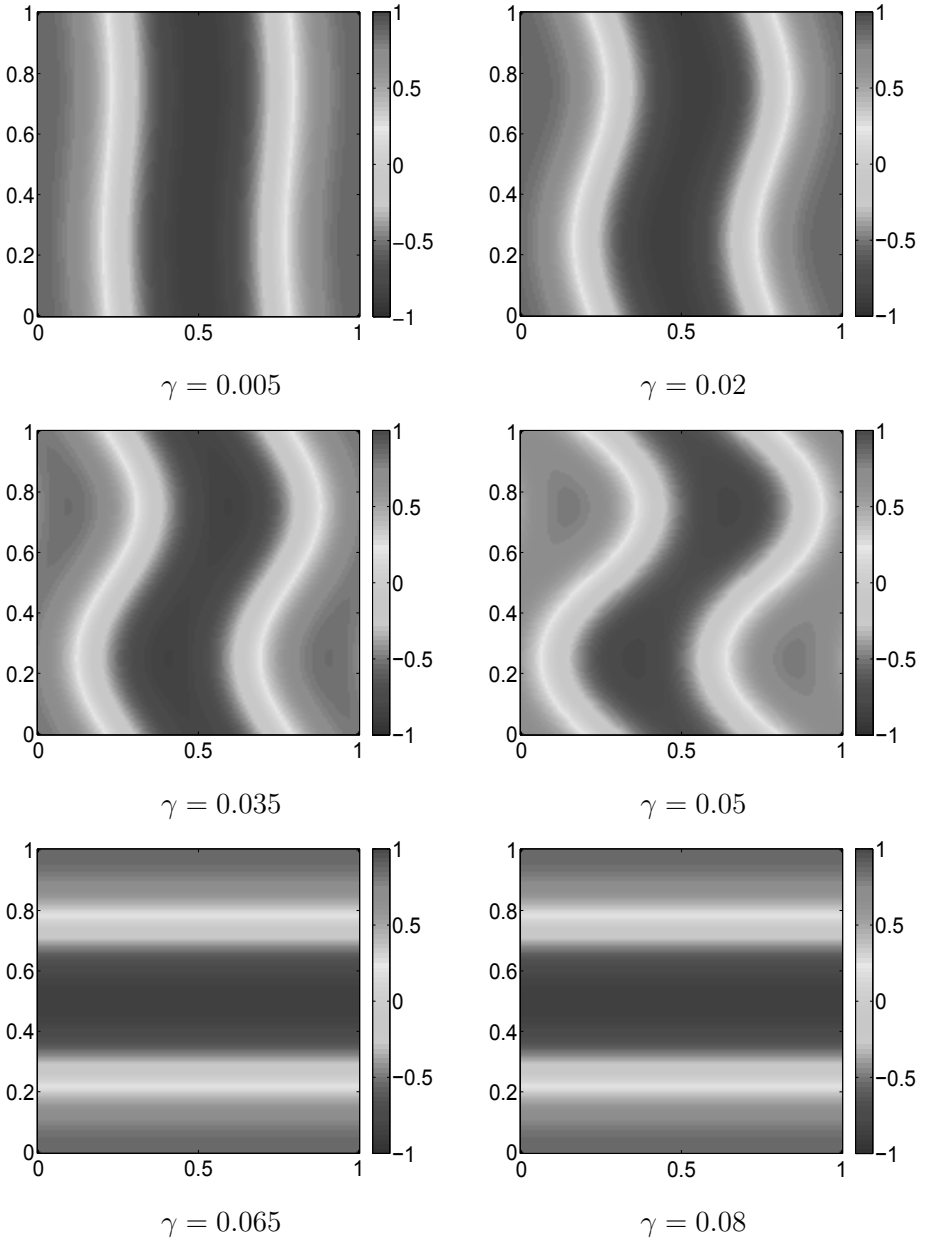

Figure 3.9 The contour plots of the saddle points $\phi(x, y)$ of the model (3.47) at various shear rates γ. The profile of ϕ becomes stretched horizontally as γ gets larger. But it becomes lamellar phase and does not change any more when γ is sufficiently large. The colourbar on the right of each figure shows the values of ϕ.

of the saddle points get more and more stretched along the shear direction until a lamellar phase is attained for γ large enough. In fact, the lamellar phase shown in the last two subfigures (Fig. 3.9(e) and Fig. 3.9(f)) is always a saddle point for γ. It seems to have a critical γ_* between 0.05 and 0.065 such that for $\gamma < \gamma_*$, there are two saddle points: one is twisted and the other is lamellar, and for $\gamma > \gamma_*$, it seems only one index-1 saddle point, the lamellar phase. To determine which saddle point corresponds to the minimal action for escaping at $\gamma < \gamma_*$, additional computations using the minimum action method [52] are required.

For the dynamics in (3.48), the shear flow introduces forces in both $x-$ and $y-$ directions. The saddle point profiles for various γ are shown in Fig. 3.10 [43]. The shear "twists" the profiles again but in different patterns. Similarly to the first case, the saddle point is finally unchanged when γ is sufficiently large. And eventually, the saddle point forms an "X" shape. For small shear rate, the "X" shaped saddle point, the last image in Fig. 3.10 does not exit, unlike the lamellar phase in the previous shear flow case. Thus, there seems to have only one index-1 saddle point at any γ, except for the symmetric images. In summary, the shear flow acting on the Ginzburg-Landau energy landscape induces a variety of different patterns of the saddle points and transition mechanisms. Our simplified GAD offers a useful tool to locate these saddle points with economic computational costs.

APPENDIX

The proof of Theorem 3.1

Proof Theorem 3.1 *(a) By the condition that (x_*, v_*) is a fixed point of the simplified GAD* (3.22)*, we have*

$$\begin{cases} b(x_*) - 2\langle b(x_*), v_*\rangle v_* = 0, & \text{(3.50a)} \\ J(x_*)v_* = \langle v_*, J(x_*)v_*\rangle v_*. & \text{(3.50b)} \end{cases}$$

Equation (3.50b) *implies that v_* is the eigenvector of $J(x_*)$ corresponding to the eigenvalue $\lambda_* \doteq \langle v_*, J(x_*)v_*\rangle$. Making inner product with v_* on both sides of* (3.50a)*, we get*

$$\langle b(x_*), v_*\rangle - 2\langle b(x_*), v_*\rangle \langle v_*, v_*\rangle = 0.$$

Since $\|v_*\| = 1$, we have $\langle b(x_*), v_*\rangle = 0$. Thus $b(x_*) = 0$ by (3.50a)*.*

(b) Since x_s is a fixed point of the system $\dot{x} = b(x)$, we have $b(x_s) = 0$, thus

$$b(x_s) - 2\langle b(x_s), v_i\rangle v_i / \|v_i\|^2 = 0, \quad i = 1, 2, \ldots, n. \tag{3.51}$$

Since $J(x_s)v_i = \lambda_i v_i$, by taking inner product with v_i on both sides of (3.50b) *and using the condition $\|v_i\| = 1$, we get $\lambda_i = \langle J(x_s)v_i, v_i\rangle$, and*

$$J(x_s)v_i - \langle J(x_s)v_i, v_i\rangle v_i = 0, \quad i = 1, 2, \ldots, n. \tag{3.52}$$

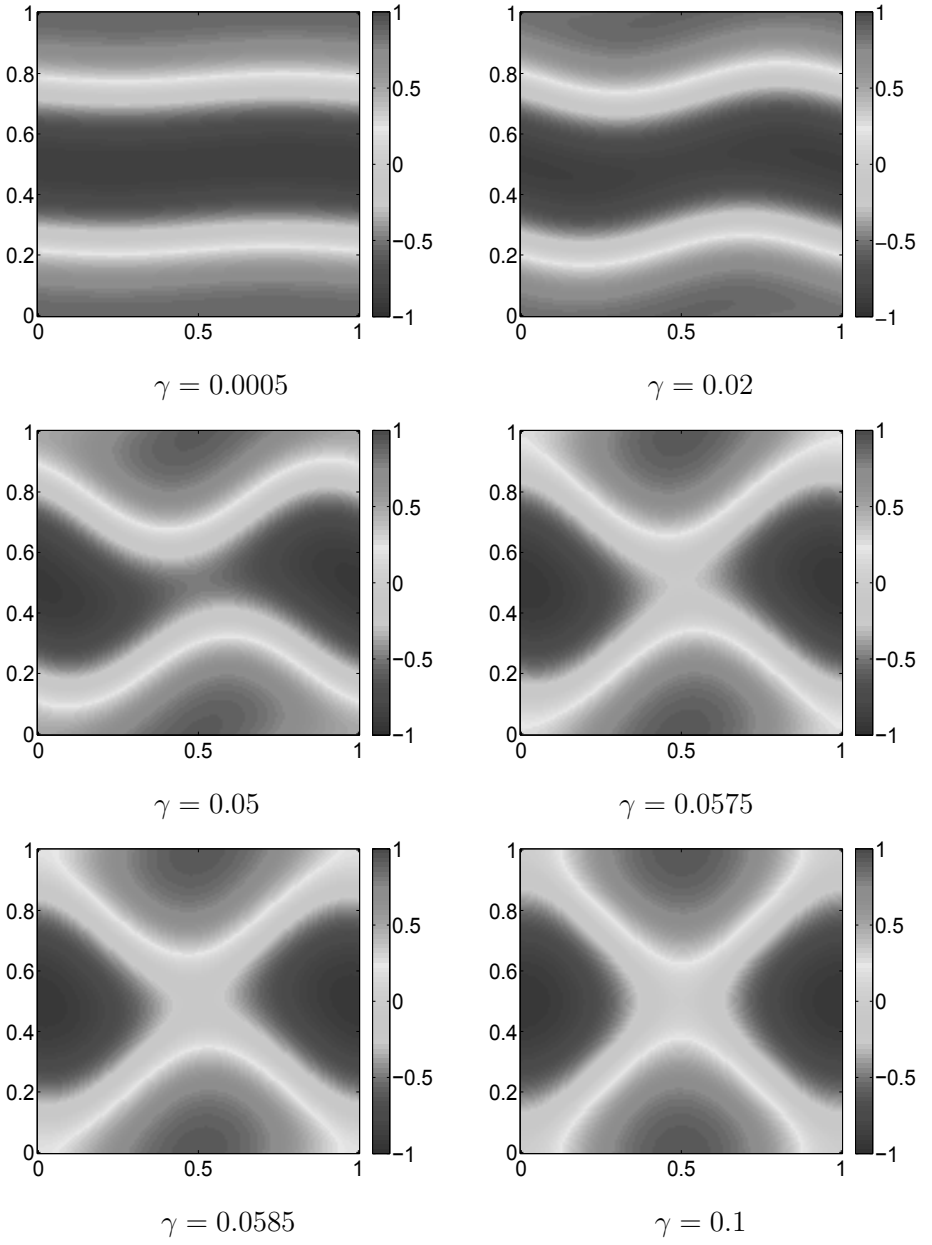

Figure 3.10 The contour plots of the saddle points $\phi(x, y)$ of the model (3.48) at various shear rates γ. The profile of ϕ gets twisted more and more as γ increases and forms an "X" shape in the end. The colourbar on the right of each figure also shows the values of ϕ.

Equation (3.51) *and* (3.52) *imply that* (x_s, v_i) *is the fixed point of the simplified GAD* (3.22) *for all* $i = 1, 2, \cdots, n$.

Next, we write down the eigenvalues and corresponding eigenvectors of the Jacobian matrix of the simplified GAD at any fixed points (x_s, v_i). *First, the Jacobian matrix of the simplified GAD* (3.22) *has the expression:*

$$\tilde{\mathbb{J}}(x_s, v_i) = \begin{bmatrix} \mathbb{N}_1, & 0 \\ *, & \mathbb{M}_1 \end{bmatrix}, \tag{3.53}$$

where

$$\mathbb{N}_1 := J - 2\lambda_i v_i v_i^T, \quad \mathbb{M}_1 := J - \lambda_i - v_i v_i^T(\lambda_i + J).$$

The eigenvalues of $\tilde{\mathbb{J}}$ *can be obtained from the eigenvalues of its two diagonal blocks* $\mathbb{N}_1$ *and* $\mathbb{M}_1$. *It is verified that*

$$\begin{aligned} \mathbb{N}_1 v_i &= Jv_i - 2\lambda_i v_i v_i^T v_i = -\lambda_i v_i, \\ \mathbb{N}_1 v_j &= Jv_j - 2\lambda_i v_i v_i^T v_j = \lambda_j v_j, \\ \mathbb{M}_1 v_i &= Jv_i - \lambda_i v_i - v_i v_i^T(\lambda_i + J)v_i = -2\lambda_i v_i, \end{aligned}$$

and

$$\begin{aligned} &\mathbb{M}_1(v_j - (v_j^T v_i)v_i) = \mathbb{M}v_j - (v_j^T v_i)\mathbb{M}v_i \\ &= (J - \lambda_i - v_i v_i^T(\lambda_i + J))v_j + 2\lambda_i(v_j^T v_i)v_i \\ &= (\lambda_j - \lambda_i)v_j - v_i(\lambda_i + \lambda_j)v_i^T v_j + 2\lambda_i(v_j^T v_i)v_i \\ &= (\lambda_j - \lambda_i)v_j - v_i(\lambda_j - \lambda_i)v_i^T v_j \\ &= (\lambda_j - \lambda_i)(v_j - (v_i^T v_j)v_i). \end{aligned}$$

Hence the eigenvalues of the Jacobian matrix $\tilde{\mathbb{J}}$ *at any fixed points* $(x_s, v_i), i = 1, 2, \ldots, n$ *are*

$$-2\lambda_i, -\lambda_i, \{\lambda_j : j \neq i\}, \{\lambda_j - \lambda_i : j \neq i\}. \tag{3.54}$$

The linear stability condition is that all the above eigenvalues of $\tilde{\mathbb{J}}$ *are negative. Thus one fixed point* $(x_s, v_{i'})$ *is linearly stable if and only if* $\lambda_{i'} > 0$ *and all other eigenvalues* $\lambda_j < 0$ *for* $j \neq i'$. *In this case, the fixed point* x_s *is an index-1 saddle point of the system* $\dot{x} = b(x)$.

Remark 3.4 *Theorem 3.1 also holds for the simplified GAD* (3.23). *In this case, the Jacobian matrix of the simplified GAD* (3.23) *becomes*

$$\tilde{\mathbb{J}}(x_s, w_i) = \begin{bmatrix} \mathbb{N}_2, & 0 \\ *, & \mathbb{M}_2 \end{bmatrix},$$

where

$$\mathbb{N}_2 := J - 2\lambda_i w_i w_i^T, \quad \textit{and} \quad \mathbb{M}_2 := J^T - \lambda_i - w_i w_i^T(\lambda_i + J^T).$$

This Jacobian matrix has the same eigenvalues (3.54) *as the Jacobian matrix* (3.53) *of the simplified GAD* (3.22).

Relation with the Hamilton's equation In this part, we discuss the Hamilton's equation associated with the rare event study of the equation (1.1). According to the Freidlin-Wentzell large deviation principle (LDP) [31], as $\epsilon \to 0$, the most probable transition path over the time interval $[0, T]$ of the system (1.1) is a minimizer of the following Freidlin-Wentzell action functional

$$S[\phi] = \int_0^T L(\phi, \dot{\phi}) \mathrm{d}t, \tag{3.55}$$

where the Lagrangian $L(x, y)$ is defined as

$$L(x, y) := \frac{1}{2} \langle y - b(x), y - b(x) \rangle . \tag{3.56}$$

$\langle \cdot, \cdot \rangle$ is the inner product in $\mathbb{R}^d$. The Hamiltonian $H(x, p)$, as the conjugate of the Lagrangian $L(x, y)$, is

$$H(x, p) = \langle b(x), p \rangle + \langle p, p \rangle /2. \tag{3.57}$$

It is well-known that the minimizer of $S[\phi]$, denoted as $x(t)$, satisfies the Hamilton's equations

$$\begin{cases} \dot{x} = H_p = b(x) + p(t), & \text{(3.58a)} \\ \dot{p} = -H_x = -J(x)^{\mathsf{T}} p(t), & \text{(3.58b)} \end{cases}$$

where $p(t)$ is the (generalized) momentum. $J(x) = Db(x)$ is the Jacobian matrix we have used before in the GAD. The eigenvalues of $J(x)$ are denoted as $\{\lambda_i\}$. Equation (3.58) looks superficially similar to equation (3.23) with two differences: (1) the signs before $J(x)^{\mathsf{T}}$ are the opposite and (2) the momentum p is not a unit vector as the direction variable w. In fact, the critical point of (3.58) is (x_*, p_*) where $b(x_*) = 0$ and $p_* = 0$ by assuming that $J(x_*)$ is non-degenerate. Assume $J(x)$ has the right-eigenvectors v_i and the left-eigenvectors w_i:

$$Jv_i = \lambda_i v_i, \text{ and } J^{\mathsf{T}} w_i = \lambda_i w_i, \quad 1 \le i \le d,$$

where all eigenvalues are assumed distinctive and the left or right eigenvectors both form a basis of $\mathbb{R}^d$. We introduce the normalized unit vector u to represent the direction of p. Define the scalar $l \doteq \|p\|^2$, then $u = p/\sqrt{l}$ and $\dot{l} = 2 \langle p, \dot{p} \rangle = -2 \left\langle p, J^{\mathsf{T}} p \right\rangle = -2l \left\langle u, J^{\mathsf{T}} u \right\rangle$. So,

$$\dot{u} = \frac{d}{dt} \left(\frac{p}{\sqrt{l}} \right) = -J^{\mathsf{T}}(x) u + \left\langle u, J^{\mathsf{T}} u \right\rangle u. \tag{3.59}$$

By the important zero-Hamiltonian condition[31] $H \equiv 0$, we have

$$l = \|p\|^2 = -2 \langle b, p \rangle = -2\sqrt{l} \langle b, u \rangle ,$$

that is,

$$l = 0, \quad \text{or} \quad \sqrt{l} = -2\langle b(x), u\rangle .$$

$l = 0$ means $p = 0$, which corresponds to the original dynamics $\dot{x} = b(x)$. l is not always zero for the exit dynamics, then $\sqrt{l} = -2\langle b(x), u\rangle$ and the equation (3.58a) becomes

$$\dot{x} = b(x) + \sqrt{l}u = b(x) - 2\langle b(x), u\rangle u. \tag{3.60}$$

So far, by (3.59) and (3.60), we get the momentum-normalized version for the Hamilton's equations (3.58) restricted on the zero-H hypersurface:

$$\begin{cases} \dot{x} = b(x) - 2\langle b(x), u(t)\rangle u(t)/\|u(t)\|^2, & \text{(3.61a)} \\ \dot{u} = -J^{\mathsf{T}}(x)u + \left\langle u, J^{\mathsf{T}}u\right\rangle u. & \text{(3.61b)} \end{cases}$$

$\|u_0\| = 1$ is assumed. Note that this dynamics (3.61) is not exactly identical to the original Hamilton's equation (3.58) since the branch of $p \equiv 0$ has been discarded.

Now, the only difference between the Hamilton's equation (3.61) and the simplified GAD (3.23) is the opposite sign on the right hand sides of (3.61b) and (3.23b). By Remark 3.4, the Jacobian matrix of (3.61) is $\begin{bmatrix} \mathbb{N}_2, & 0 \\ *, & -\mathbb{M}_2 \end{bmatrix}$, whose eigenvalues are $-\lambda_i, 2\lambda_i, \{\lambda_j, j \neq i\}, \{\lambda_i - \lambda_j, j \neq i\}$. The position dynamics in (3.23) and (3.61) have the same form of applying the projection matrix $I - 2ww^{\mathsf{T}}$ or $I - 2uu^{\mathsf{T}}$ in front of the original force $b(x)$. The difference is which direction they select. If x were frozen, the w dynamics in equation (3.23b) picks up the least stable direction while the momentum direction in Hamilton's equations uses the most stable direction. Thus the GAD (3.23) can converge to the saddle point of the vector field $b(x)$ while the Hamiltonian dynamics (3.61) has no stable steady state. So one may view the simplified GAD as a modification of the Hamilton's equation by flipping the sign of the (normalized) momentum to stabilize the saddle point. Note that although we can introduce a factor γ for (3.23b), as shown in Remark 3.2, to speed up the clock for the direction dynamics, there is no such a freedom for the Hamilton's equation (3.61b).

In conclusion, several innovative extensions of Gentlest Ascent Dynamics (GAD) have been introduced to enhance the identification of transition states in complex rare event systems. The multiscale Gentlest Ascent Dynamics (MsGAD) was presented as an effective tool for slow-fast systems, ensuring computational efficiency while resolving multiscale dynamics. The simplified GAD, designed for non-gradient systems, significantly reduces computational costs by eliminating the need for Jacobian matrix transpose operations, while maintaining accuracy. An adaptation of this simplified approach for multiscale non-gradient slow-fast systems further optimizes performance for complex problems. Additionally, a linear projection operator was introduced to

address numerical challenges arising from higher-order spatial derivatives in energy functionals within the H^{-1} metric. The numerical validation of these methods demonstrated their effectiveness and computational advantages, confirming the potential of these extensions to tackle complex transition state identification problems efficiently. These novel variants of GAD offer significant improvements in both accuracy and computational efficiency, paving the way for more scalable and practical applications in rare event studies.