Learning Most Probable Transition Pathway for Stochastic Dynamical Systems

Ting Gao

Center for Mathematical Sciences, School of Mathematics and Statistics
Huazhong University of Science and Technology
Collaborators: Weiwei, Xiaoli Chen, Jianyu Chen, Jin Guo, Peng Zhang, Jinqiao Duan

July. 17 - 21, 2023
International Conference Dynamical Systems and Semi-algebraic geometry:
interations with Optimization and Deep Learning
Da Lat University

Outline

SDE v.s. Data Science

Explanations
- Training Dynamics
- Convergence
- Stability

non-Gaussian Stochastic Dyn. Sys.

Deep Learning

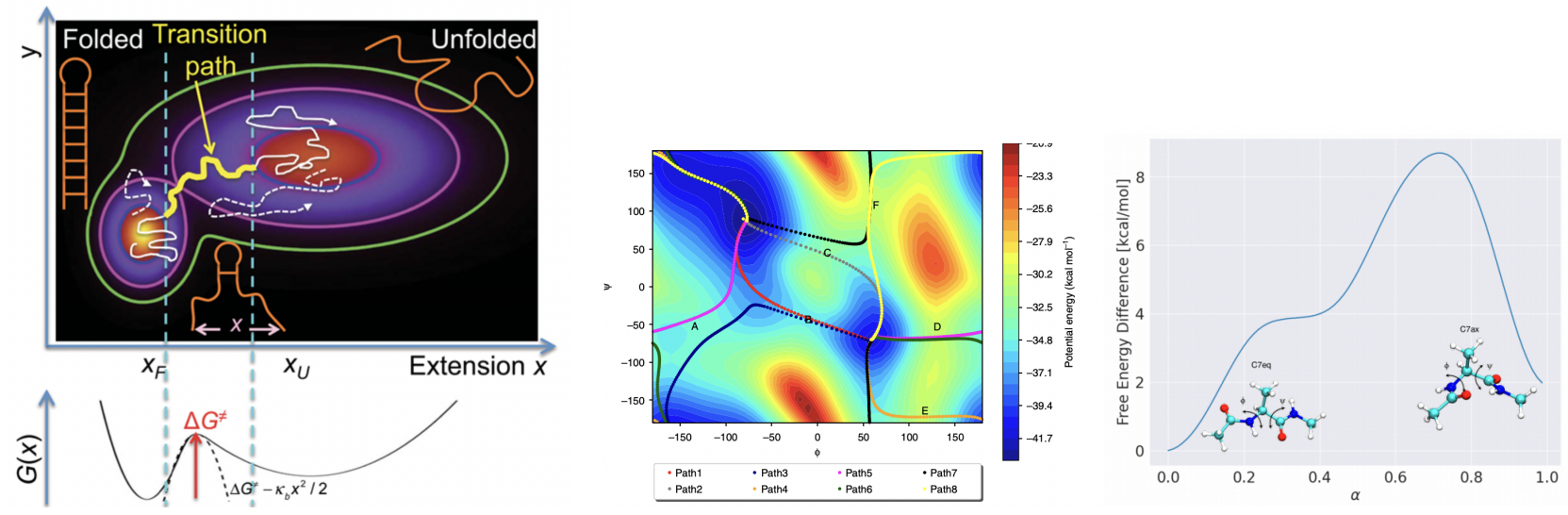
Applications: Identification, Prediction, Control

- **Extracting Governing Laws from Data**
  - Unknown Mechnisms
  - Latent Dynamical System
  - High-dim Linear Embedding
- **Learning Dynamics of the SDEs**
  - Evolutionary pdf
  - Invariant Manifold
  - Transition Pathway
  - Exit/Transition Time

- **Methods**
  - SINDY/MLE/Krammers-Moyal/...
  - Manifold Learning
  - Reservior Computing
  - Generative Models
  - Meta Learning
  - Reinforcement Learning
  - GCN / SNN

- **High-dim Spatial-Temporal data**
  - Mathematical Finance
  - Brain Science
  - Wireless Communication
  - Privacy Computing

Motivation Background



(a) (b)

Figure: (a)Protein folding/unfolding along energy landscape; (b)Conformational transition pathways of alanine dipeptide.

Goal: how to find transition pathways (rare events) in a fast and accurate way?

How to get the most probable transition pathway?

Minimizing action functional for SDE

$$dX_t = f(X_t)dt + \varepsilon\, dL_t$$

.

• Finite noise: Onsager-Machlup action functional with $\varepsilon > 0$.

$\min_z \int_0^T OM(\dot{z}(t), z(t))dt$:

$z_m(0) = x_0, \quad z_m(T) = x_1$ (Two metastable states: x_0, x_1)

• Small noise perturbation: Freidlin-Wentzell action functional, with $0 < \varepsilon \ll 1$.

Compensated Poisson process

Consider d-dimensional SDE under a pure Lévy jump process:

$$dX_t^\varepsilon = b(X_t^\varepsilon)\,dt + \varepsilon d\tilde{L}_t^\varepsilon, \quad X_0^\varepsilon = x_1. \tag{1}$$

Function $b : \mathbb{R}^d \to \mathbb{R}^d$ is the drift term. For every $\varepsilon > 0$, the non-Gaussian Lévy process $(\tilde{L}_t^\varepsilon)_{t\geq 0}$ is given by

$$\tilde{L}_t^\varepsilon = \int_0^t \int_{\mathbb{R}^d\setminus\{0\}} z\tilde{N}^{\frac{1}{\varepsilon}}(ds, dz), \tag{2}$$

where $\tilde{N}^{\frac{1}{\varepsilon}}$ is a compensated Poisson random measure defined on a given complete probability space $(\Omega, \mathscr{F}, \mathbb{P})$. The measure ν has the form

$$\nu(dz) = e^{-|z|^\gamma} dz, \quad \gamma > 1. \tag{3}$$

Freidlin-Wentzell action functional

Consider

$$dX_t^\varepsilon = b(X_t^\varepsilon)\,dt + \varepsilon d\tilde{L}_t^\varepsilon, \quad X_0^\varepsilon = x_1. \tag{4}$$

where $\tilde{L}_t^\varepsilon$: Lévy process with compensated Poisson random measure.

Define $S_T : C[0,T] \longrightarrow [0,\infty]$

$$S_T(\varphi) \triangleq \inf_{\varphi = F(g)} \left\{ \int_0^T \int_{\mathbb{R}^d \setminus \{0\}} (g(s,z)\ln g(s,z) - g(s,z) + 1)\nu(dz)ds \right\} \tag{5}$$

where reference trajectory $\varphi(t)$ satisfies

$$\varphi(t) = x + \int_0^t b(\varphi(s))ds + \int_0^t \int_{\mathbb{R}^d \setminus \{0\}} z(g(s,z) - 1)\nu(dz)ds. \tag{6}$$

and Eq.(6) has a unique solution if every positive measure function g satisfying (A. Budhiraja et al. 2013)

$$\int_0^T \int_{\mathbb{R}^d \setminus \{0\}} (g(s,z)\ln g(s,z) - g(s,z) + 1)\nu(dz)ds < \infty.$$

Reformulate constrained Minimization as an optimal control problem

$$\begin{cases} \inf\limits_{g\in\mathscr{U}} & \mathscr{J}[\varphi;g]=\int_0^T \mathscr{L}(g(s,\cdot))dt+\eta(\varphi(T)), \\ \text{subject to} & \dot{\varphi}(t)=b(\varphi(t))+\mathscr{Q}(g(t,\cdot)), \\ & \varphi(0)=x_1. \end{cases} \tag{7}$$

where

$$\mathscr{L}(g(s,\cdot))=\int_{\mathbb{R}^d\setminus\{0\}}\big(g(s,z)\ln g(s,z)-g(s,z)+1\big)\nu(dz), \tag{8}$$

and

$$\mathscr{Q}(g(t,\cdot))=\int_{\mathbb{R}^d\setminus\{0\}}(g(t,z)-1)z\nu(dz). \tag{9}$$

Function η is the terminal cost, and it is defined by

$$\begin{cases} \eta(x)=0, & x=x_2, \\ \eta(x)=\infty, & \text{otherwise}. \end{cases} \tag{10}$$

Solution: a supervised learning way

Idea: Construct two neural networks:

$$\varphi_{NN}(t; w_\varphi, b_\varphi), \quad g_{NN}(x, t; w_g, b_g).$$

$$\begin{aligned}
&\text{loss}_\varphi = \\
&\frac{1}{N_T}\sum_{i=1}^{N_T}\left(\dot{\varphi}_{NN}(t_i) - b(\varphi_{NN}(t_i)) - \textbf{Int}_z\left(z(g_{NN}(t_i,z)-1)e^{-|z|^\gamma}\right)\right)^2 + \\
&\tau_1(\varphi_{NN}(0)-x_1)^2, \\
&\text{loss}_g = \\
&\frac{1}{N_T}\sum_{i=1}^{N_T}\textbf{Int}_z\left(\left(g_{NN}(t_i,z)\ln g_{NN}(t_i,z) - g_{NN}(t_i,z)+1\right)e^{-|z|^\gamma}\right) + \\
&\tau_2(\varphi_{NN}(T)-x_2)^2,
\end{aligned}$$

$$\text{loss} = \tau\text{loss}_\varphi + \text{loss}_g, \tag{11}$$

where τ is the weight to balance loss_φ and loss_g.

Experiment

Example

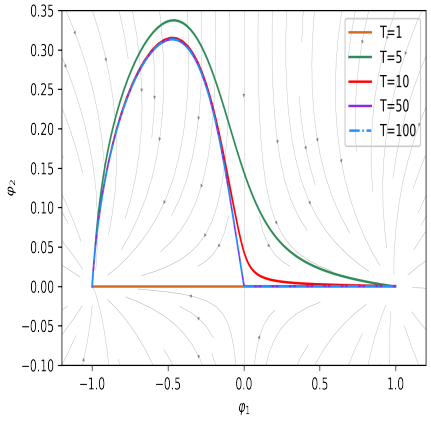
Consider the Maier Stein system under non-Gaussian Lévy noise:

$$dX_t^\varepsilon = b(X_t^\varepsilon)\,dt + \varepsilon d\tilde{L}_t^\varepsilon, \quad X_0^\varepsilon = x_1, \tag{12}$$

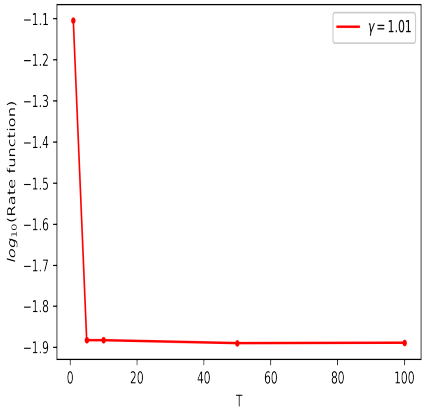
where b(x,y)=$\begin{pmatrix} x - x^3 - \beta x y^2 \\ -(1+x^2)\,y \end{pmatrix}$ .and $\nu(dx) = \exp(-|x|^\gamma)dx$.

This system has two metastable points $(\pm 1, 0)$ and one saddle point $(0,0)$. When $\beta = 1$, the system is a gradient system with potential $V(x,y) = -\frac{1}{2}x^2 + \frac{1}{4}x^4 + \frac{1}{2}y^2 + \frac{1}{2}x^2y^2$.

Experiment results



(a)



(b)

Figure: (a) The most likely transition path for the Maier Stein system under non-Gaussian Lévy noise from the metastable point $(-1,0)$ to the metastable point $(1,0)$ within time $T = 1,5,10,50,100$. (b) The value of rate function for the most likely transition path within the time $T = 1,5,10,50,100$.

Maximum Principle

We consider a stochastic differential equation with multiplicative Gaussian noise in $\mathbb{R}^d$

$$dX(t) = \tilde{b}(X(t))dt + \sigma(X(t))dB(t),\ t \in [0, t_f], \tag{13}$$

with initial condition $X(0) = x_0 \in \mathbb{R}^d$, where $\tilde{b} : \mathbb{R}^d \to \mathbb{R}^d$ is a regular function ('drift' or 'vector field'), $\sigma : \mathbb{R}^d \to \mathbb{R}^{d\times k}$ is a $d \times k$ matrix-valued function ('noise intensity'), and B is a Brownian motion in $\mathbb{R}^k$.

Onsager-Machlup action functional

The Onsager-Machlup action functional is

$$S(z,\dot{z}) = \frac{1}{2}\int_0^{t_f} [(\dot{z}-b(z))\,V(z)(\dot{z}-b(z))^T + \operatorname{div} b(z) - \frac{1}{6}R(z)]dt, \tag{14}$$

where $V(z) = (\sigma(z)\sigma^*(z))^{-1}$, $R(z)$ is the scalar curvature with respect to the Riemannian metric induced by $V(z)$, and $b^i(z) = \tilde{b}^i(z) - \frac{1}{2}\sum_{l,j}(V^{-1}(z))^{lj}\Gamma^i_{lj}$ is the i component of b. Γ^i_{lj} is the Christoffel symbols associated with this Riemannian metric, which satisfies

$$\Gamma^i_{lj} = \frac{1}{2}\sum_m g^{im}\left(\frac{\partial}{\partial x^j}g_{lm} + \frac{\partial}{\partial x^l}g_{jm} - \frac{\partial}{\partial x^m}g_{lj}\right), \tag{15}$$

where $(g^{ij})(z)$ is the inverse of the Riemannian metric $(g_{ij})(z) = V(z)$.

Onsager-Machlup action functional

The divergence $\mathrm{div}\, b(z)$ is defined as

$$\mathrm{div} b(z) = \frac{1}{\sqrt{|V(z)|}} \sum_i \frac{\partial}{\partial z_i} \left(b^i(z) \sqrt{|V(z)|} \right), \tag{16}$$

where $|V(z)|$ is the determinate of Riemann metric.

The Onsager-Machlup action functional can be considered as the integral of a Lagrange

$$L(z, \dot{z}) = \frac{1}{2}[(\dot{z} - b(z))\, V(z) (\dot{z} - b(z))^T + \mathrm{div}\, b(z) - \frac{1}{6} R(z)]. \tag{17}$$

Pontryagin's Maximum Principle

Define the Hamiltonian

$$H : \mathbb{R} \times \mathbb{R}^d \times \mathbb{R}^d \times \Theta \longrightarrow \mathbb{R},$$

$$(t, x, p, \theta) \longmapsto H(t, x, p, \theta) := p^\top f(t, x, \theta) - L(t, x, \theta).$$

We can rewrite the cost functional in terms of the Hamiltonian:

$$J(\theta) = \int_{t_0}^{t_f} (\langle p(t), \dot{x}(t) \rangle - H(t, x(t), \theta(t), p(t)))dt + \Phi(x(t_f)). \quad (18)$$

Recall Pontryagin's Maximum Principle:
1) The joint evolution of x^* and p^* are governed by:

$$\begin{cases} \dot{x}^*(t) = \nabla_p H(t, x^*(t), p^*(t), \theta^*(t)), & x^*(t_0) = x_0, \\ \dot{p}^*(t) = -\nabla_x H(t, x^*(t), p^*(t), \theta^*(t)), & p^*(t_f) = -\nabla_x \Phi(x^*(t_f)). \end{cases} \quad (19)$$

2) For each t, Hamiltonian has a global maximum at $\theta = \theta^*$, i.e.,

$$H(t, x^*(t), p^*(t), \theta^*(t)) \geq H(t, x^*(t), p^*(t), \theta(t)), \quad \forall\, \theta \in \Theta \text{ and } t \in [t_0, t_f] \quad (20)$$

Reformulate as a deterministic optimal control problem

$$\begin{cases} \underset{\theta\in\Theta}{\text{minimize}} & \frac{1}{2}\int_0^{t_f}[\theta^2 + \operatorname{div} b(z) - \frac{1}{6}R(z)]dt + \Phi(z(t_f)), \\ \text{subject to} & \dot{z}(t) = \tilde{b}(z(t)) + \sigma(z(t))\theta(t), \\ & z(0) = x_0. \end{cases} \tag{21}$$

Here $z : [0, t_f] \to \mathbb{R}^d$ is the state and $\theta \in \Theta$ is the feedback control. The function Φ is the terminal cost, and it is defined by

$$\Phi(x) = \frac{(x - x_{t_f})^2}{(x - x_{t_f})^2 + 1}. \tag{22}$$

Solution: Solve Eq.(21, 22) through Maximum Principle!

Algorithm:

Algorithm 1 Back Propagation Based Extended MSA

1: **Input:** $\theta^0 \in \Theta$, x_0;
2: **Iterations:** for k=0 to K, do;
3: build neural network for controller θ;
4: forward solve $\dot{x}_t^{\theta^k} = \nabla_p H\left(t, x_t^{\theta^k}, p_t^{\theta^k}, \theta_t^k\right) = f(t, x_t^{\theta^k}, \theta_t^k), \quad x_0^{\theta^k} = x_0$ with second-order Runge-Kuta;
5: backward solve $\dot{p}_t^{\theta^k} = -\nabla_x H\left(t, x_t^{\theta^k}, p_t^{\theta^k}, \theta_t^k\right), \quad p_T^{\theta^k} = -\nabla_x \Phi(x_T^{\theta^k})$ with second-order Runge-Kuta;
6: compute the loss function $L = -\tilde{H}$;
7: update θ_t^k to $\theta_t^{k+1} = \arg\max_{\theta \in \Theta} \tilde{H}\left(t, X_t^{\theta^k}, P_t^{\theta^k}, \theta, \dot{X}_t^{\theta^k}, \dot{P}_t^{\theta^k}\right)$;
8: train the network for θ_t^k according to back propagation gradient decent;
9: after K iterations, the optimal solution is obtained when the loss function converges.
10: **Output:** x_t^*, p_t^*, θ_t^* .

Convergence Study

Theorem

There exist a hyper parameter $\rho > 0$ *, a constant* $M > 0$ *and a constant* $C > 0$ *satisfying* $\rho < 2C$*. If the above two assumptions (* **H1**(*Lipschitz*) $-$ **H2**(*Regularity*) *) are satisfied, we summarize the convergence of neural network and Algorithm in the following two results:*

($\mathscr{A}$ *) When the iteration times* $k \to +\infty$, $\theta^k \to \theta^*$.

($\mathscr{B}$ *) When the iteration times* $k \to +\infty$, $J(\theta^{k+1}) - J(\theta^k)$ *as defined in* (23) *approaches* 0*, and for every* $k \geq 0$, $\Delta t > 0$*, the loss function* $J(\theta^k)$ *satisfies,*

$$J(\theta^{k+1}) - J(\theta^k) \leq -\Delta t \sum_{t=0}^{T-1} H_t\left(x_t^\theta, p_{t+1}^\theta, \theta_t\right) + M\left(\left\|\theta_t^{k+1} - \theta_t^k\right\|^2\right) \left\|x_t^\theta\right\| \tag{23}$$

A nutrient-phytoplankton-zooplankton system

Example

$$\begin{cases} \dfrac{dN}{dt} = D(N_0 - N) - f(N)P, \\ \dfrac{dP}{dt} = \alpha f(N)P - d_1 P - g(P)Z - w_1 P, \\ \dfrac{dZ}{dt} = \beta g(P)Z - d_2 Z - w_2 Z, \end{cases} \tag{24}$$

where

$$f(N) = \begin{cases} \frac{b}{a}N, & 0 \leqslant N \leqslant a, \\ b, & N > a, \end{cases}$$

$$g(P) = \frac{cP}{1 + dP}.$$

N_0 is constant input rate of nutrient, D, w_1 and w_2 are the washout rates for nutrient, d_1 and d_2 are the death rates, a and b are the conversion rates. Let $d_1 + w_1 = D_1$, $d_2 + w_2 = D_2$.

Pontryagin's Maximum Principle associated with problem

(i) $0 \leqslant x \leqslant a$

$$
\begin{cases}
\dot{x}^* = D\left(N_0 - x^*\right) - \dfrac{b}{a}x^*y^* + \sigma_1\theta_1^* \\
\dot{y}^* = \alpha\dfrac{b}{a}x^*y^* - \dfrac{cy^*}{1+dy^*}z^* - D_1y^* + \sigma_2\theta_2^* \\
\dot{z}^* = \beta\dfrac{cy^*}{1+dy^*}z^* - D_2z^* + \sigma_3\theta_3^* \\
\dot{p_1}^* = p_1^*(D + \dfrac{b}{a}y^*) + p_2^*\alpha\dfrac{b}{a}y^* + \dfrac{\alpha b}{2a} \\
\dot{p_2}^* = p_1^*\dfrac{b}{a}x^* + p_2^*(D_1 + z^*\dfrac{c}{(1+dy^*)^2} - \alpha\dfrac{b}{a}x^*) \\
\qquad - p_3^*\beta z^*\dfrac{c}{(1+dy^*)^2} + \dfrac{1}{2}(\beta\dfrac{c}{(1+dy^*)^2} - \dfrac{b}{a} + \dfrac{2z^*cd(1+dy^*)}{(1+dy^*)^4}) \\
\dot{p_3}^* = p_2^*\dfrac{cy^*}{1+dy^*} - p_3^*(\beta\dfrac{cy^*}{1+dy^*} - D_2) - \dfrac{c}{2(1+dy^*)^2}
\end{cases}
$$

(ii) $x > a$

$$
\begin{cases}
\dot{x}^* = D\left(N_0 - x^*\right) - by^* + \sigma_1\theta_1^* \\
\dot{y}^* = \alpha by^* - \dfrac{cy^*}{1+dy^*}z^* - D_1y^* + \sigma_2\theta_2^* \\
\dot{z}^* = \beta\dfrac{cy^*}{1+dy^*}z^* - D_2z^* + \sigma_3\theta_3^*. \\
\dot{p_1}^* = p_1^*D \\
\dot{p_2}^* = p_1^*b + p_2^*(D_1 + z^*\dfrac{c}{(1+dy^*)^2} - \alpha b) \\
\qquad - p_3^*\beta z^*\dfrac{c}{(1+dy^*)^2} + \dfrac{1}{2}(\beta\dfrac{c}{(1+dy^*)^2} + \dfrac{2z^*cd(1+dy^*)}{(1+dy^*)^4}) \\
\dot{p_3}^* = p_2^*\dfrac{cy^*}{1+dy*} + p_3^*(D_2 - \beta\dfrac{cy^*}{1+dy^*}) - \dfrac{c}{2(1+dy^*)^2}
\end{cases}
$$

Experiments

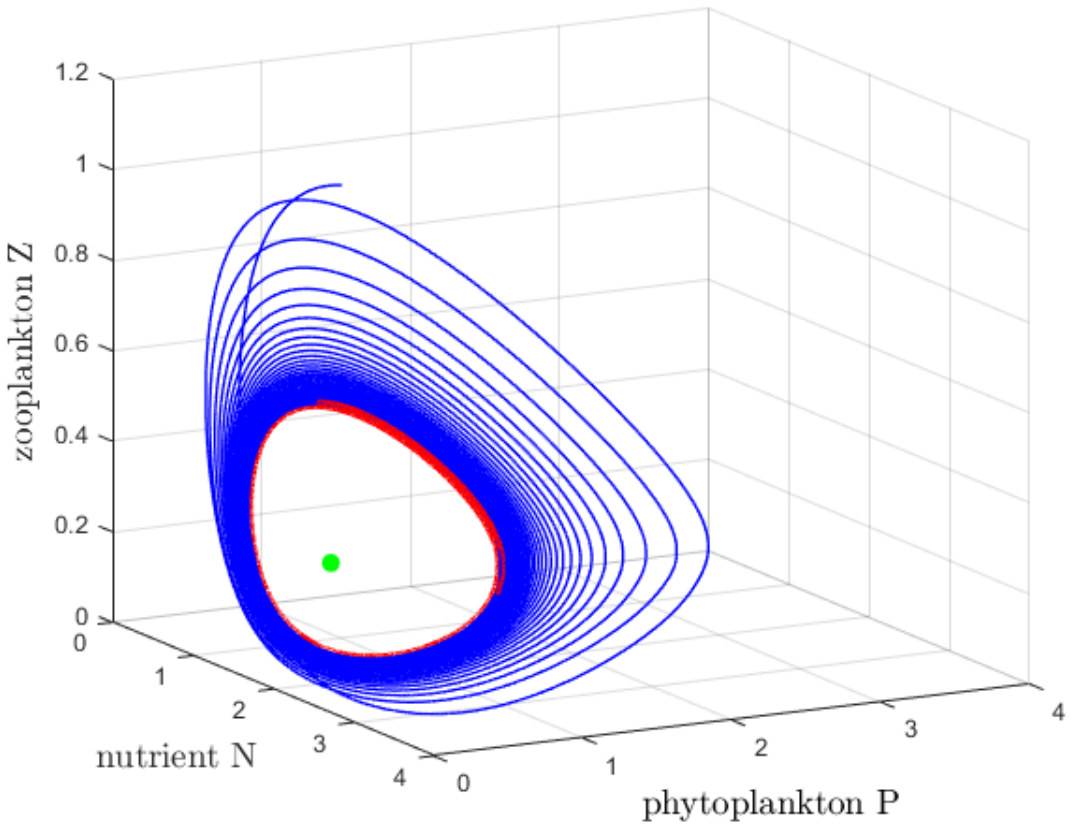


Figure: Nutrient-Plankton system–Bistable phenomenon: (1) a stable equilibrium (green point in the center) and a stable limit cycle (red curve). (2) The circular curve in blue is the asymptotic solution of this system.

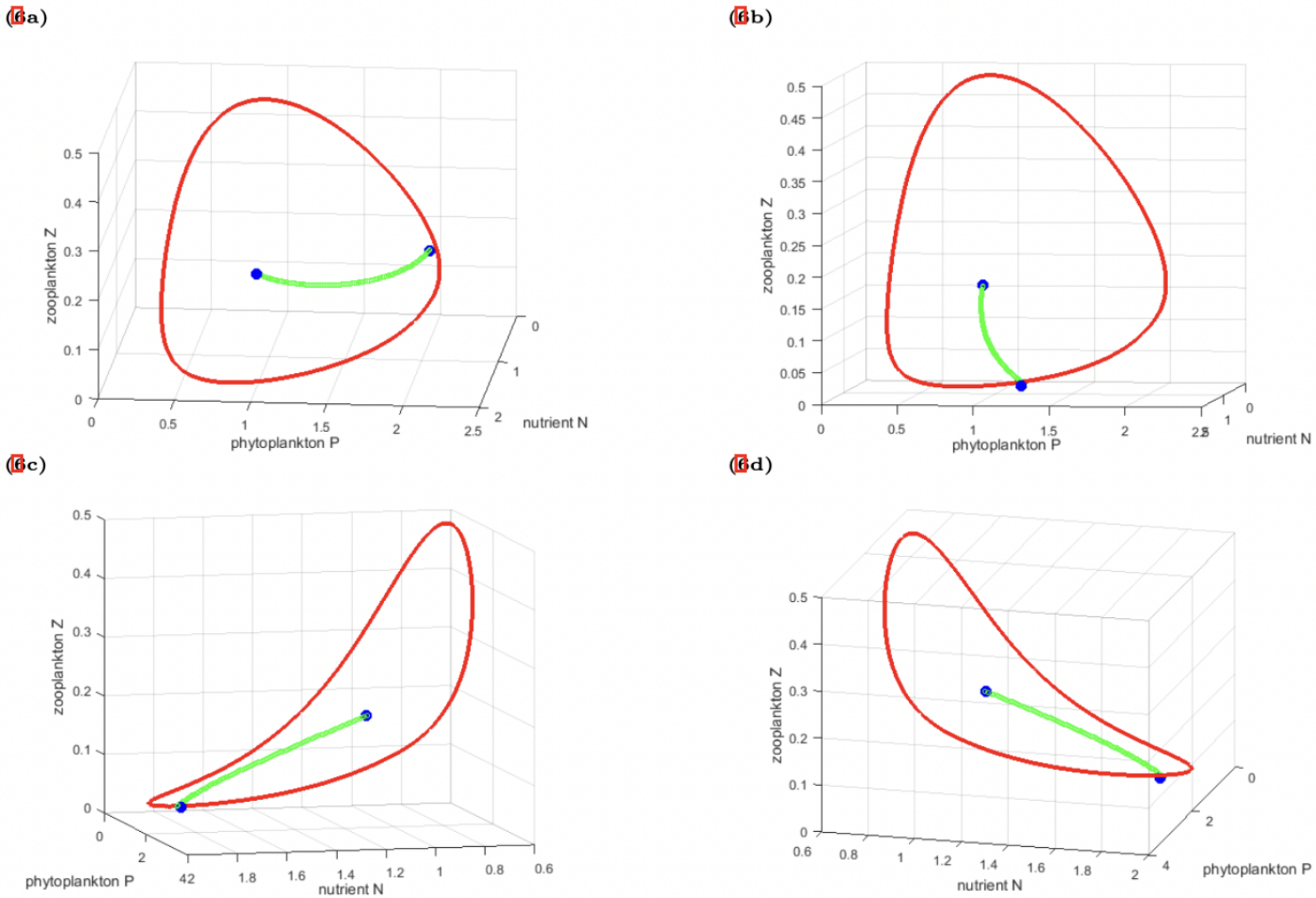


Figure 6: Nutrient-Plankton system–The most probable transition pathway in the sense of Onsager-Machlup: Transition time $T = 1$. The blue point is the coexisting equilibrium, the green line is the most probable transition pathway, and the red circle

Deep Reinforcement Learning in Finite-Horizon to Explore the Most Probable Transition Pathway

Consider the following stochastic differential equation

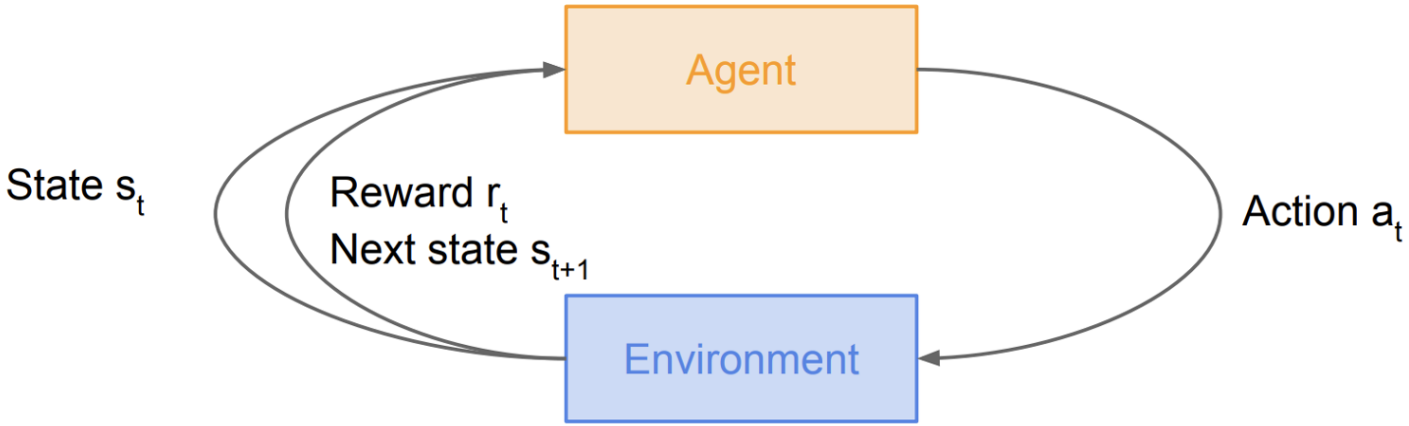
$$\begin{cases} dX(t) = b(X(t))dt + \sigma(X(t))dB(t), \\ X(0) = X_0. \end{cases} \tag{25}$$

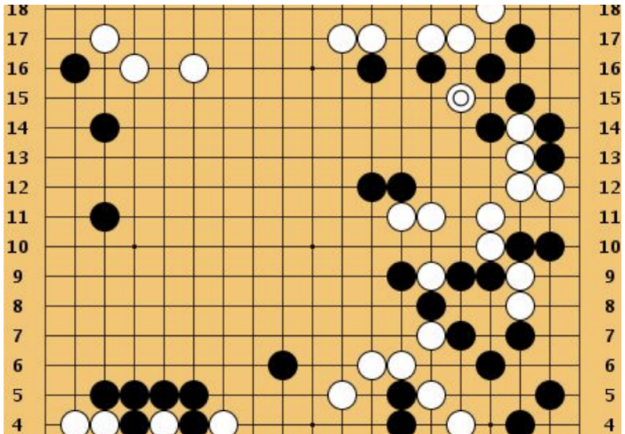
The minimization for the Onsager–Machlup action functional can be reformulated as the following constrained optimal control problem:

$$\begin{cases} \inf\limits_{u \in \mathbb{A}} \ \mathscr{J}[X,u] = \int_0^T f(t,X(t),u(t))dt + g(X(T)), \\ \text{subject to } \dot{X}(t) = b(X(t)) + \sigma(X(t))u(t), \\ \qquad\qquad X(0) = X_0, \end{cases} \tag{26}$$

Main Idea

Problems: an agent interacting with an environment, which provides numeric reward signals.
Goal: to learn how to take actions in order to maximize total reward.





Objective: Win the game!

State: Position of all pieces
Action: Where to put the next piece down
Reward: 1 if win at the end of the game, 0 otherwise

Definition

- **State space**: The state s_t is defined as the coordinates of $X(t) \in \mathbb{R}^d$ at timestep t.
- **Action space**: The action space is defined as $U \subset \mathbb{R}^u (a_t \in U)$. At timestep k ($k = 0, 1, \cdots, N$), a_k corresponds to the control term $u(k\Delta t)$ in (26).
- **Cost (Reward)**: For consistency with the optimal control problem (26), we will call the cost as running cost in the following. The running cost r_t at timestep t is defined as

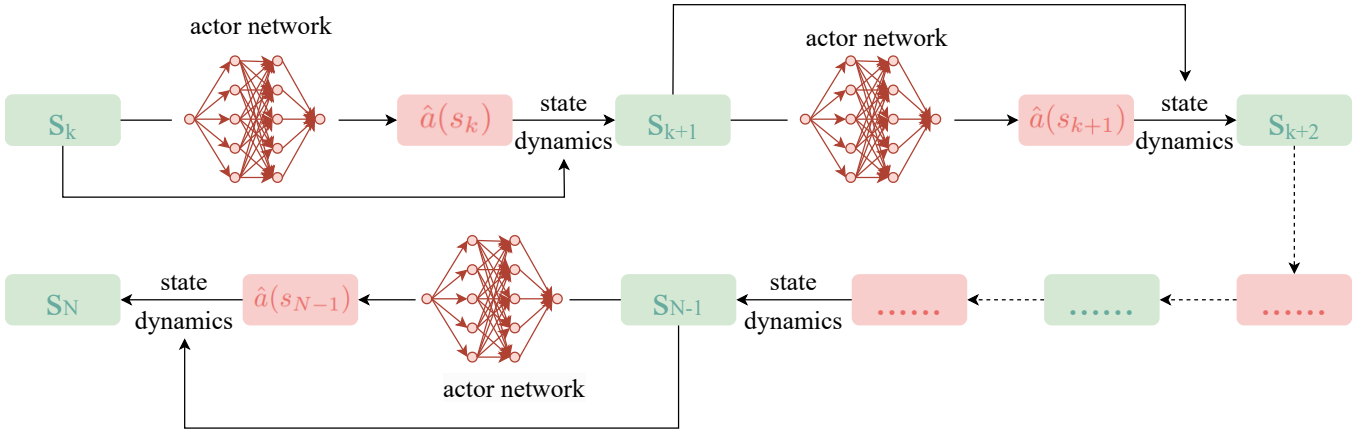
$$r_t = f(t, s(t), a(t))\Delta t.$$

  Our goal is to minimize the summation of running cost and terminal cost.
- **Transition dynamics**: The transition dynamics corresponding to the optimal control problem (26) can be expressed as

$$s_{t+1} = s_t + b(s_t)\Delta t + \sigma(s_t)a_t\Delta t\,.$$

Terminal Prediction

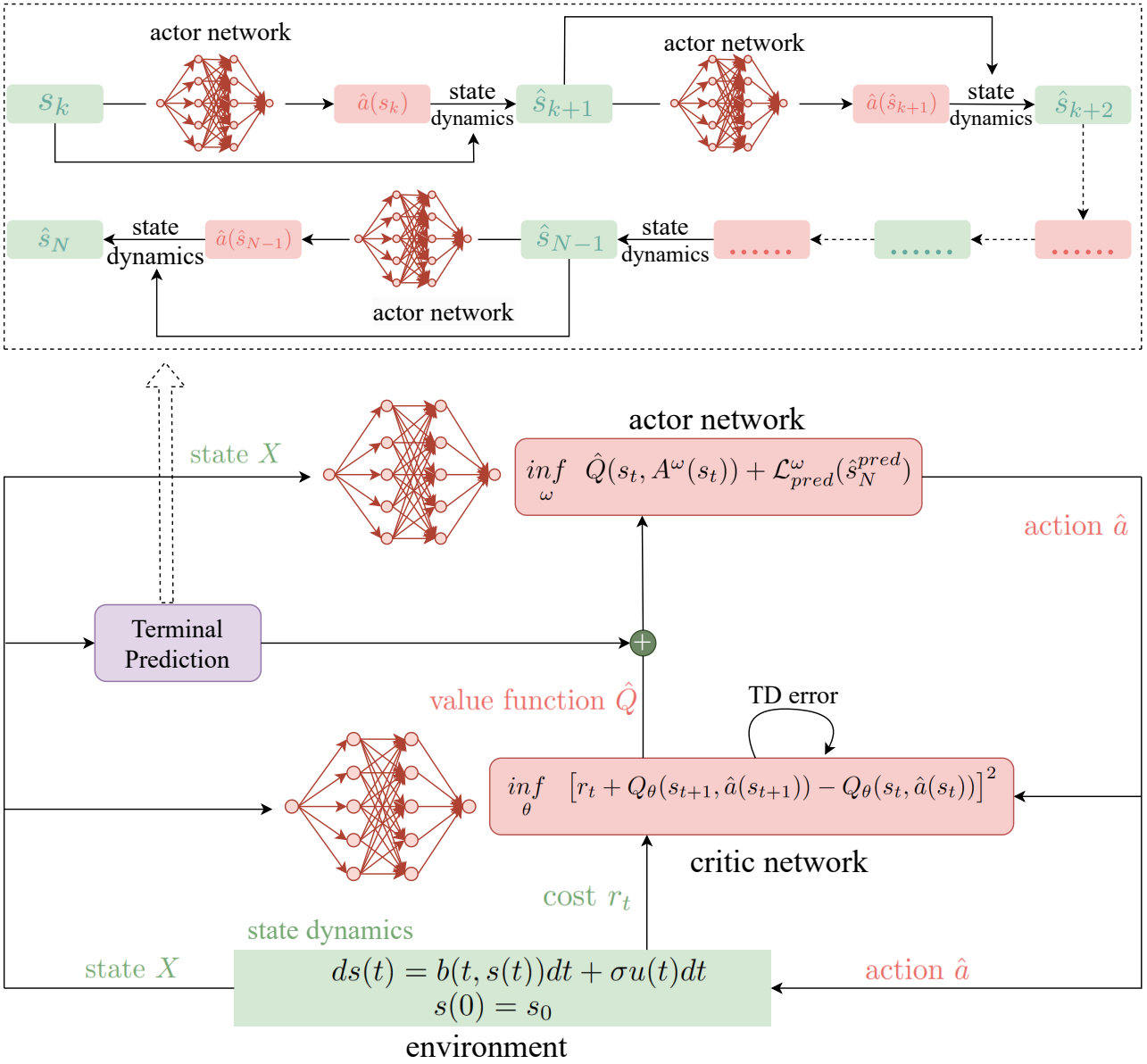
- How can the terminal constraints ensure that our agent reaches the target terminal state?



By the chain rule, we have

$$\frac{\partial \mathscr{L}_{pred}(\hat{s}_N^{k,\hat{a}})}{\partial \omega} = \frac{\partial g(\hat{s}_N^{k,\hat{a}})}{\partial \omega} = \frac{\partial g(\hat{s}_N^{k,\hat{a}})}{\partial \hat{a}} \cdot \frac{\partial \hat{a}}{\partial \omega}.$$

Terminal Prediction Deep Deterministic Policy Gradien



Algorithm

Algorithm 1: Terminal prediction Deep Deterministic Policy Gradient

Input: Randomly initialize critic network $Q_\theta(s,a)$ and actor network $A_\omega(s)$ with weights θ and ω.

Output: Updated network Q_θ and network A_ω after K episodes of the algorithm

Random run M trajectories, collect samples $\{\mathcal{D}\}$;

for $n \leftarrow 0$ **to** *episode* **do**

  $s_0 = s(0)$;

  Set $Q_\theta(s_N) = 0$;

  for $t \leftarrow 0$ **to** N **do**

    Select action with exploration noise $a_t \leftarrow A_\omega(s_t) + \epsilon,\ \epsilon \sim \mathcal{N}(0, \sigma^2_{act})$ *according to the current policy;*

    Execute action a_t *and observe running cost* $r_t = \frac{1}{2}[a_t^2 + \nabla \cdot b(s_t)]\Delta t$, *then get new state* $s_{t+1} \leftarrow F(s_t, a_t)$;

    Store transition (s_t, a_t, r_t, s_{t+1}) *in* $\mathcal{D}_{[t]}$;

    Sample a minibatch of M transitions $\{(s_t^i, a_t^i, r_t^i, s_{t+1}^i)\}_{i=1}^M$;

    Set $y_t^i \leftarrow r_t^i + Q_\theta(s_{t+1}^i, A_\omega(s_{t+1}^i))$;

    Update critic by minimizing the TD loss:

$$\mathcal{L}_Q \leftarrow \frac{1}{M}\sum_{i=1}^{M}(y_t^i - (Q_\theta^i(s_t^i, a_t^i)))^2.$$

    Terminal prediction $\mathcal{L}_{pred} \leftarrow g(\hat{s}_N^{i,pred})$;

    Update actor policy using the sampled policy gradient:;

$$\nabla_\omega \mathcal{L}_{act} \approx \frac{1}{M}\sum_{i=1}^{M}\left[\nabla_a Q_\theta(s,a)|_{s=s_t^i, a=a_t^i} \cdot \nabla_\omega A(s)|_{s=s_t^i} + \nabla_a \mathcal{L}_{pred} \cdot \nabla_\omega A(s)|_{s=s_t^i}\right].$$

  end

end

Convergence Analysis

The total error: $f^* - \hat{f}$, where $\hat{f} =$ the output of our model. Define:

- $f_m =$ best approximation to f^* in $\mathscr{A}_M$
- $\tilde{f}_{n,m} =$ "when using only the dataset S, the best approximation to f^* in $\mathscr{A}_M$"

Decomposition of the error

$$f^* - \hat{f} = \underbrace{f^* - f_m}_{appr.} + \underbrace{f_m - \tilde{f}_{n,m}}_{estim.} + \underbrace{\tilde{f}_{n,m} - \hat{f}}_{optim.} \tag{27}$$

- $f^* - f_m$= approximation error, due entirely to the choice of the hypothesis space
- $f_m - \tilde{f}_{n,m}$ = estimation error - sample error due to the fact that we only have a finite dataset
- $\tilde{f}_{n,m} - \hat{f}$ = optimization error - training error caused by the choice of optimizer

Convergence Analysis

The estimation error:
For $n = 0, 1, \cdots, N-1$, the following holds

$$\begin{aligned}\mathbb{E}\varepsilon_{n+1}^{esti} \leq & (\sqrt{2}+16)\frac{(N-n+1)\|f\|_\infty + \|g\|_\infty}{\sqrt{M}} \\ & + 16\left([f]_L + \eta_M\gamma_M + [g]_L\frac{\rho_M - \rho_M^{N-n+1}}{1-\rho_M}\right)\frac{\gamma_M}{\sqrt{M}}.\end{aligned} \tag{28}$$

The estimation error measures how closely the selected estimator (e.g., the mean square estimate) approximates a certain quantity (e.g., the conditional expectation). Obviously, we anticipate that the estimation to become more accurate when the size of the training set is sufficiently large.

Convergence Analysis

Since the class of neural networks $\mathscr{A}_M$ is not dense in the set $\mathbb{A}^{\mathscr{X}}$ of all Borelian functions, we consider the approximation error as a measure of how accurately the neural network function in set $\mathscr{A}_M$ approximates the regression function.
The approximation error:
For $n = 0, 1, \cdots, N-1$, the following holds

$$\begin{aligned}\varepsilon_{n+1}^{approx} \leq \inf_{A \in \mathscr{A}_M} \Big\{ & 2[f]_L \Delta t \mathbb{E}_M \Big[\Big| A(X_n) - a^{opt}(X_n) \Big| \Big] \\ & + 2\mathbb{E}_M \Big[\Big| \hat{Q}_n(X_n, A(X_n)) + g(X_N^{n,A}) - Q_n(X_n, a^{opt}(X_n)) \Big| \Big] \Big\}. \end{aligned} \tag{29}$$

Convergence Analysis

Theorem. Assume there exists an optimal feedback control $(a^{opt}(X_k))_{k=1}^n$ for the control problem with the optimal state-action value Q_k for $k = 1, 2, \cdots, n$. Then, as $M \to \infty$,

$$
\begin{aligned}
&\inf_{A \in \mathscr{A}_M} \mathbb{E}_M \Big[\Big| \hat{Q}_n(X_n, A(X_n)) + g(X_N^{n,A}) - Q_n(X_n, a^{opt}(X_n)) \Big| \Big] \\
&\quad = \mathscr{O}_{\mathbb{P}} \Bigg(\Big(\gamma_M^4 \frac{K_M log(M)}{M} \Big)^{\frac{1}{2n}} + \Big(\frac{\gamma_M \sqrt{\log(M)}}{\sqrt{M}} \Big(\eta_M \gamma_M + [g]_L \frac{\rho_M - \rho_M^{N-n+1}}{1-\rho_M} \Big) \Big)^{\frac{1}{2n}} \\
&\quad + \sup_{1 \le k \le n} \inf_{A \in \mathscr{A}_M} \inf_{\Phi \in \mathscr{Q}_M} \Big(\mathbb{E}_M \big[\big| \Phi(X_k, A(X_k)) + g(X_N^{n,A}) - Q_k(X_k, a^{opt}(X_k)) \big| \big] \Big)^{\frac{1}{2n}} \\
&\quad + \sup_{0 \le k \le n-1} \inf_{A \in \mathscr{A}_M} \Big(\mathbb{E}_M \big[\big| A(X_k) - a^{opt}(X_k) \big| \big] \Big)^{\frac{1}{2n}} + \Big(\big| \hat{Q}_0(X_0, \hat{a}_0(X_0)) - Q_0(X_0, a^{opt}(X_0)) \big| \Big)^{\frac{1}{2n}} \Bigg),
\end{aligned}
\tag{30}
$$

where $\mathbb{E}_M$ denotes the expectation conditioned by the training set used to estimate the optimal policies $(\hat{a}_k)_{k=1}^n$. The notation $z_M = \mathscr{O}_{\mathbb{P}}(y_M)$ as $M \to \infty$ stands for that there exists $c > 0$ such that $\mathbb{P}(|z_M| > c|y_M|) \to 0$ as M goes to infinity.

Experiments

Consider the Maier–Stein system in $\mathbb{R}^2$ that

$$\begin{cases} dX_t = b(X_t)dt + \varepsilon dB_t\,, \\ X_0 = (-1,0), X_T = (1,0) \in \mathbb{R}^2\,, \end{cases} \tag{31}$$

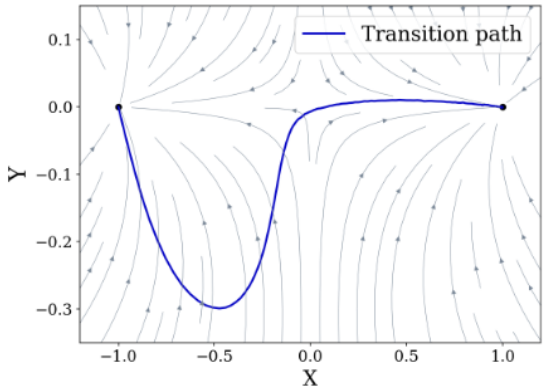
where

$$b(X_t) = \begin{pmatrix} x - x^3 - \beta x y^2 \\ -(1+x^2)y \end{pmatrix},$$

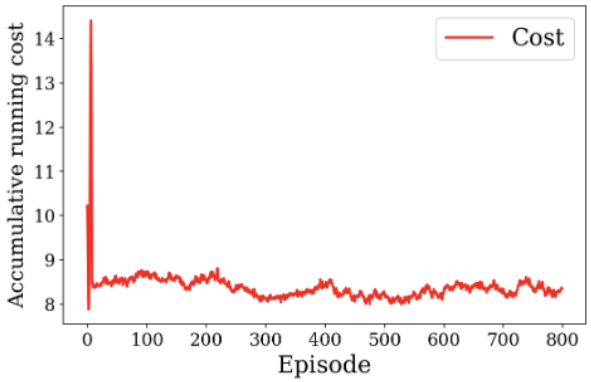
and the ε is a positive constant representing the noise intensity. Onsager–Machlup action functional

$$S^{OM}(X,\dot{X}) = \frac{1}{2}\int_0^T [|u_t|^2 + \nabla \cdot b(X_t)]dt\,.$$

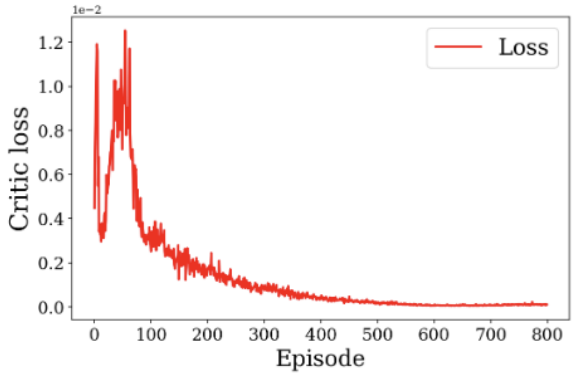
Experiments



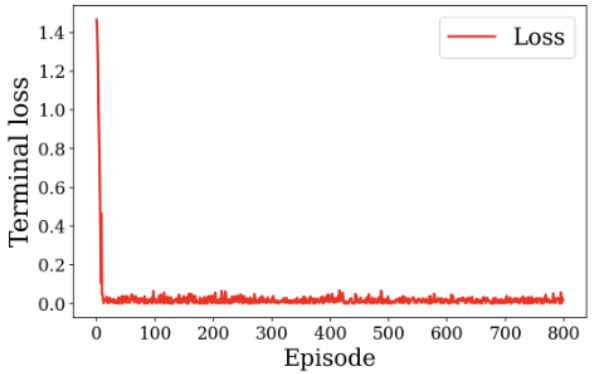
(a) The most probable transition pathway



(b) Accumulative running cost



(c) Critic loss



(d) Terminal loss

Figure 4: Maier-Stein system with $\beta = 10$, noise intensity $\varepsilon = 0.2$, and $T = 10$.

Experiements

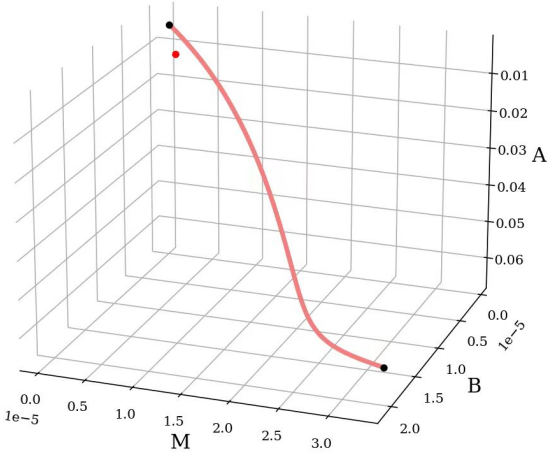
Example (Lactose Operon Model (Yildirim and Mackey, 2004))

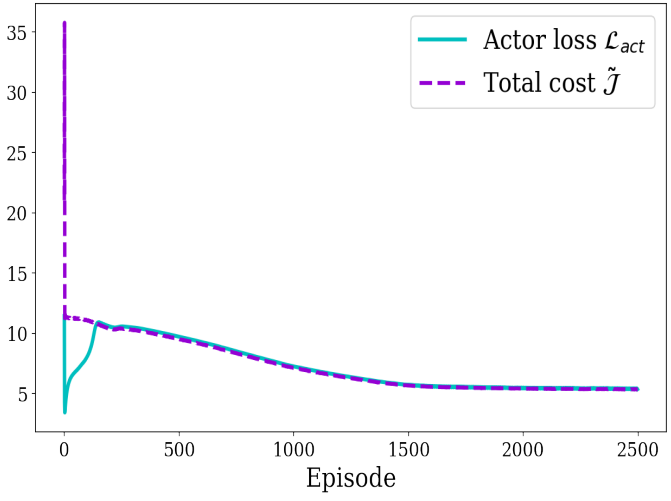
$$\begin{cases} \dfrac{dM}{dt} = \alpha_M \dfrac{1 + K_1(e^{-\mu\tau_M} A_{\tau_M})^n}{K + K_1(e^{-\mu\tau_M} A_{\tau_M})^n} - \tilde{\gamma}_M M + \varepsilon dB_t\,, \\ \dfrac{dB}{dt} = \alpha_B e^{-\mu\tau_B} M_{\tau_B} - \tilde{\gamma}_B B + \varepsilon dB_t\,, \\ \dfrac{dA}{dt} = \alpha_A B \dfrac{L}{K_L + L} - \beta_A B \dfrac{A}{K_A + A} - \tilde{\gamma}_A A + \varepsilon dB_t\,, \end{cases} \tag{32}$$

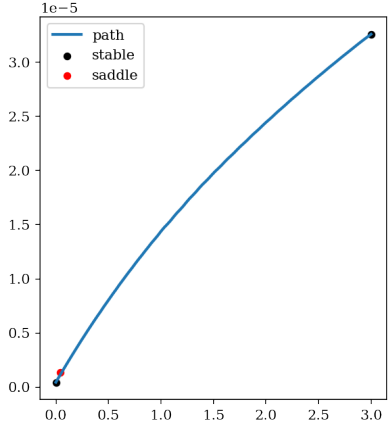
Denote M as mRNA concentration, B as the galactosidase concentration, and A represents the concentration of allolactose. Onsager-Machlup action functional

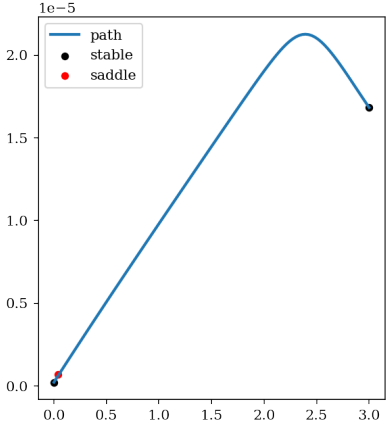
$$S^{OM}(X, \dot{X}) = \frac{1}{2} \int_0^T [|u_t|^2 + \nabla \cdot b(X_t)] dt$$

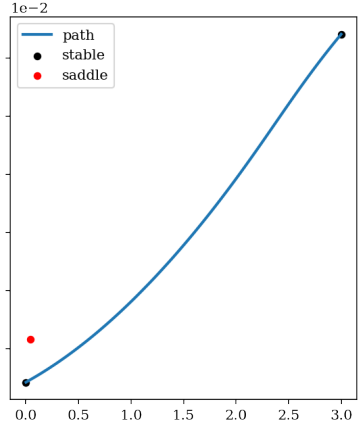
Experiement results











Take home message

1. Transition pathway through optimal control: Non-Gaussian Lévy noise, multiplicative noise

2. Deep learning methods: supervised learning ($\sim$ 2 hours); Maximum principle ($\sim$ 30 mins); Reinforcement Learning ($\sim$ 5 mins)