Controlling a Stochastic Harmonic Oscillator

Joonsoo Lee

Advised by: Georg Stadler and Shanyin Tong

July 2023

Key Takeaways

- We are able to approximate the probability of a rare event even in high dimensions.
- Using JAX's auto differentiation functionality, it is possible to get the gradient of the probability estimation, enabling us to optimize with extreme events.

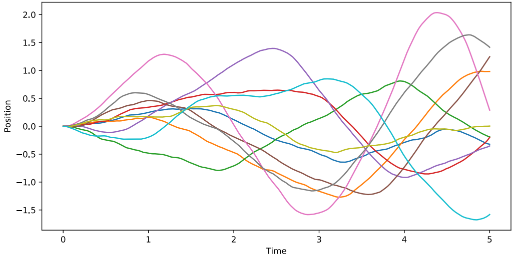
Model

Consider a nonlinear stochastically forced damped harmonic oscillator:

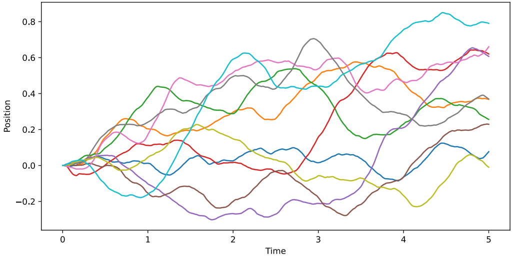
$$m\phi'' + \gamma\phi' + G(\phi, u, t) = \sigma\eta(t).$$

- G is a nonlinear deterministic forcing term, which depends on parameters u.
- We use $G = u_3\phi + u_4\phi^3$, and $\phi(0) = u_0, \phi'(0) = u_1$.
- $\sigma > 0$, η is white noise.

Random realizations (unlikely to have large displacement at final time)



No Damping ($\gamma = 0$)



Damped ($\gamma = 2.5$)

Probability Estimation

We compute low chance probability estimations using the following formula. [Schorlepp et al, '23]

$$\mathbb{P}[F(\phi) \geq z] \xrightarrow[\sigma \to 0]{} (2\pi)^{-1/2} C_F(z) \exp\left(-I_F(z)\right),$$

$$I_F(z) = \frac{1}{2} \|\eta_z(t)\|_{L^2}^2,$$

$$C_F(z) = [2 I_F(z) \det\left(1_{N\times N} - \lambda_z \mathrm{pr}_{\eta_z^\perp} \nabla^2 F(\eta_z) \mathrm{pr}_{\eta_z^\perp}\right)]^{-1/2},$$

$$\mathrm{pr}_{\eta_z^\perp} = 1_{N\times N} - \frac{\eta_z \otimes \eta_z}{||\eta_z||^2}.$$

Instanton

To calculate $I_F(z) = \frac{1}{2}\|\eta_z(t)\|_{L^2}^2$, we must find $\eta_z(t)$ (Instanton), which is the solution to the following optimization problem:

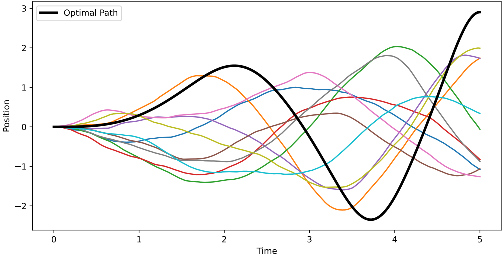
$$\min_{\eta:[0,T]\to\mathbb{R}} \frac{1}{2}\|\eta(t)\|_{L^2}^2,$$

$$\text{s.t. } m\phi'' + \gamma\phi' + G(\phi, u, t) = \sigma\eta(t),$$

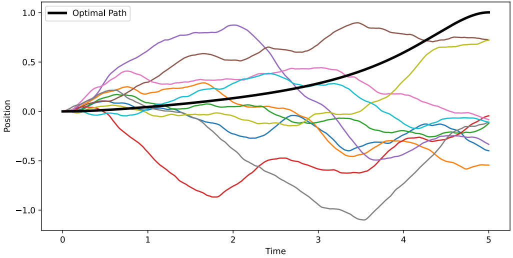
$$F(\phi) \geq z.$$

- This is a PDE constrained optimization problem.
- In terms of large deviation theory, this means that the 'most likely' forcing leading to $F(\phi) \geq z$ tells us a lot about $\mathbb{P}[F(\phi) \geq z]$.

Instanton Paths



No Damping ($\gamma = 0$)



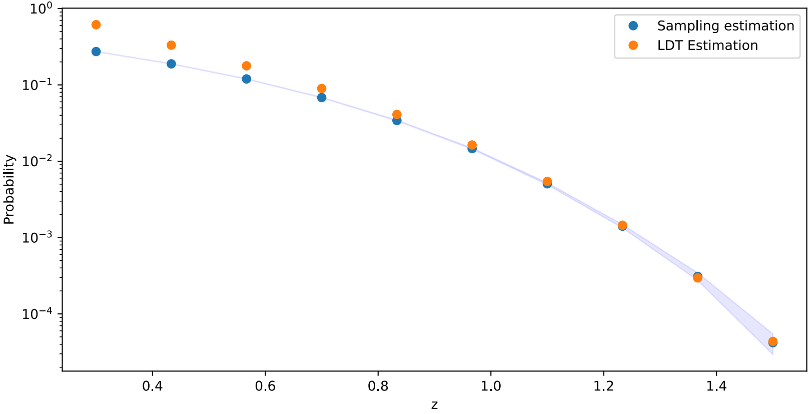
Damped ($\gamma = 2.5$)

Prefactor

Eigenvalue computations are done to find the determinant of the operator in the prefactor term,

$$C_F(z) = [2 I_F(z) \det(1_{N\times N} - \lambda_z \mathrm{pr}_{\eta_z^\perp} \nabla^2 F(\eta_z) \mathrm{pr}_{\eta_z^\perp})]^{-1/2}.$$

- The eigenvalues of the operator go to 1.
- There are several algorithms to do this, but we use randomized SVD.



LDT approximation values with $\gamma = 1.5$, $m = 1$, $\sigma = 1$. 10^6 samples calculated.

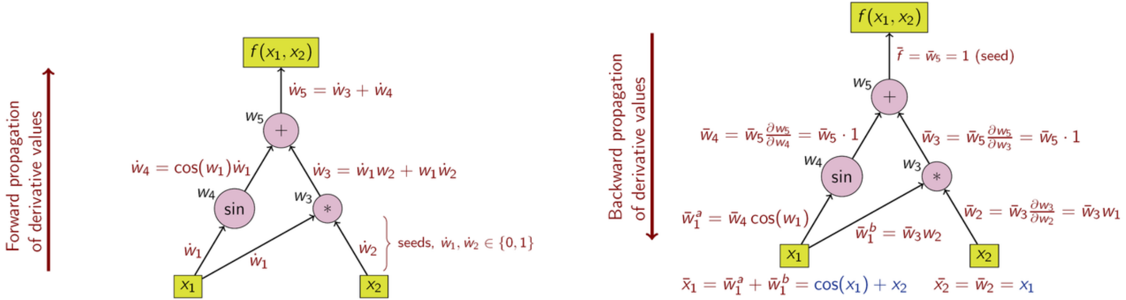
JAX Introduction

Now that we have probability estimation, we use JAX Auto-diff to obtain the gradient of the probability estimation.

- Google JAX is a python framework for transforming numerical functions.
- Often used in machine learning applications, designed to replace TensorFlow.

Algorithmic Differentiation (AD)

JAX uses algorithmic differentiation to compute gradients of functions. The basic idea is to calculate iterations of the chain rule.



(a) Forward Differentiation

(b) Backward Differentiation

Diagrams of AD for $F(x_1, x_2) = x_1 x_2 + \sin(x_1)$.

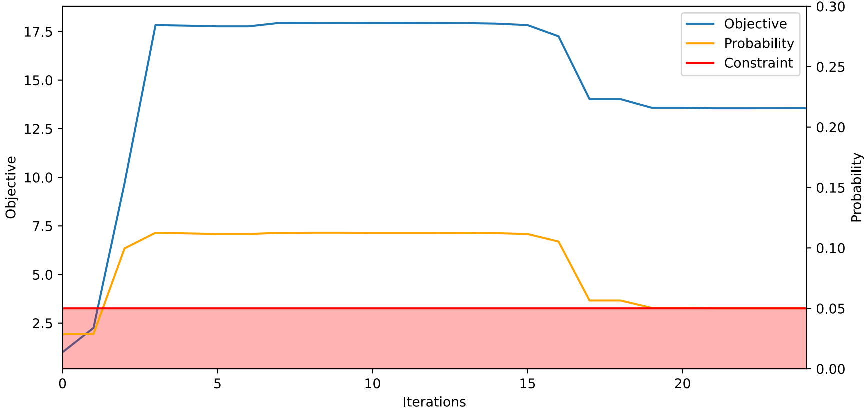
Result

We can now control the behavior of the oscillator. One important application is in solving such optimization problems:

$$\max_{u \in \mathcal{U}} J(u),$$
$$\text{s.t. } \mathbb{P}[F(u,\eta) \geq z] \leq \alpha.$$

For example, we want to find the largest initial state such that the probability of $\phi(T) \geq 0.5$ is less than 5%.

$$\max_{u \in \mathcal{U}} u_0^2 + u_1^2,$$
$$\text{s.t. } \mathbb{P}[\phi(T) \geq 0.5] \leq 0.05,$$
$$\phi(0) = u_0, \phi'(0) = u_1,$$
$$0 \leq u_0 \leq 3, 0 \leq u_1 \leq 3.$$



Objective
Probability
Constraint
17.5
15.0
12.5
10.0
7.5
5.0
2.5
0.30
0.25
0.20
0.15
0.10
0.05
0.00
0
5
10
15
20
Iterations

Key Takeaways

- We are able to approximate the probability of a rare event even in high dimensions.
- Using JAX's auto differentiation functionality, it is possible to get the gradient of the probability estimation, enabling us to optimize with extreme events.