A Contour Stochastic Gradient Langevin Dynamics Algorithm for Simulations of Multi-modal Distributions

A scalable dynamic importance sampling algorithm

Wei Deng [1] Guang Lin [1] Faming Liang [1]

October 23, 2020

[1]Purdue University

Non-convex energy function leads to slow mixing

Given a non-convex energy function $U(x)$ of a density $\pi(x) \propto e^{-\frac{U(x)}{\tau}}$, the standard sampling algorithm converges quite slowly.

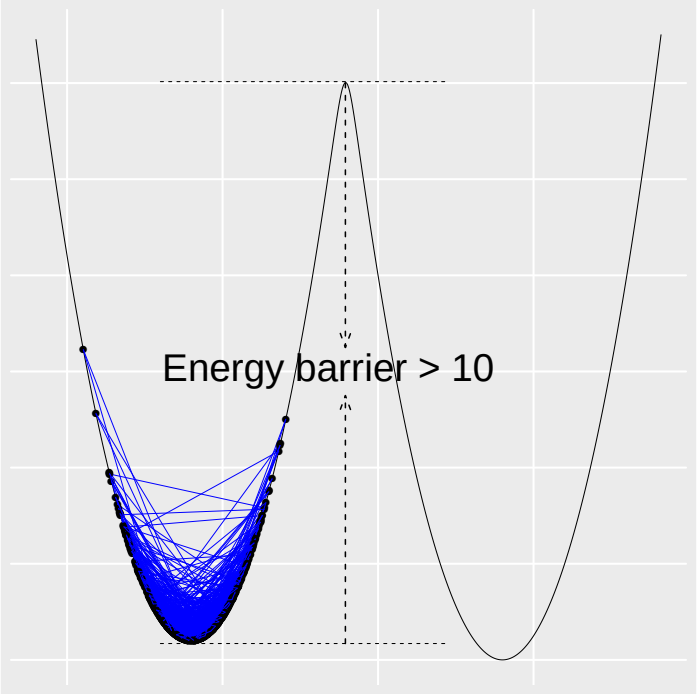


Figure 1: An example of a non-convex energy function $U(x)$.

Acceleration via importance sampling

To speed-up the simulation, we consider the importance sampling:

Acceleration via importance sampling

To speed-up the simulation, we consider the importance sampling:

$$\mathbb{E}[f(x)] = \int f(x) \underbrace{\pi(x)}_{\text{original density}} dx = \int f(x) \underbrace{\varpi_{\Psi_\theta}(x)}_{\text{a new density}} \underbrace{\frac{\pi(x)}{\varpi_{\Psi_\theta}(x)}}_{\text{importance weight}} dx,$$

where $f(\cdot)$ is a test function and $\pi(\cdot)$ is a multi-modal distribution.

Acceleration via importance sampling

To speed-up the simulation, we consider the importance sampling:

$$\mathbb{E}[f(x)] = \int f(x) \underbrace{\pi(x)}_{\text{original density}} dx = \int f(x) \underbrace{\varpi_{\Psi_\theta}(x)}_{\text{a new density}} \underbrace{\frac{\pi(x)}{\varpi_{\Psi_\theta}(x)}}_{\text{importance weight}} dx,$$

where $f(\cdot)$ is a test function and $\pi(\cdot)$ is a multi-modal distribution.

We can try to simulate from an easier distribution $\varpi_{\Psi_\theta}(x)$ indirectly.

Why not simulate from a flattened distribution

A flattened distribution $\varpi_{\Psi_\theta}(\cdot)$ reduces the energy barrier.

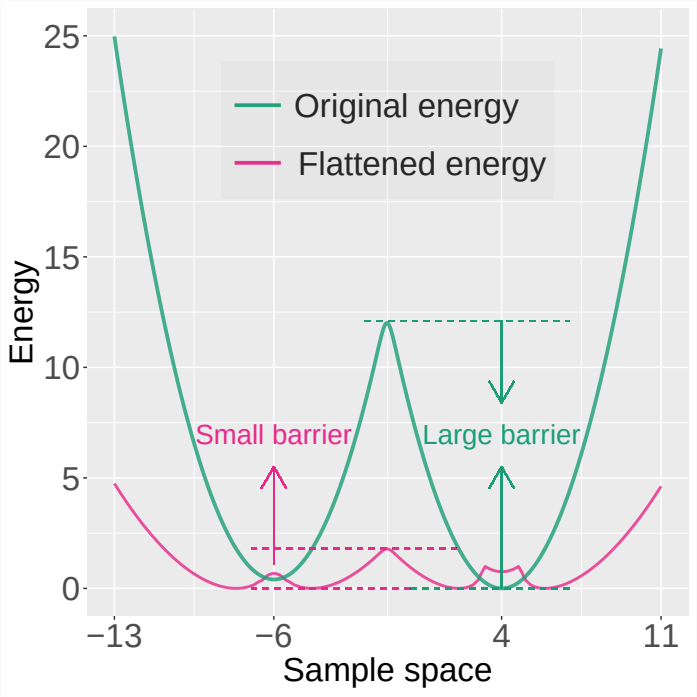


Figure 2: Energy function: original v.s. flattened.

How to construct the flattened distribution?

We partition the sample space $\mathcal{X}$ into m subregions based on $U(\boldsymbol{x})$: $\mathcal{X}_1 = \{\boldsymbol{x} : U(\boldsymbol{x}) \leq u_1\}$, $\mathcal{X}_2 = \{\boldsymbol{x} : u_1 < U(\boldsymbol{x}) \leq u_2\}$, ..., for some $\{u_i\}_{i=1}^m$.

How to construct the flattened distribution?

We partition the sample space $\mathcal{X}$ into m subregions based on $U(x)$: $\mathcal{X}_1 = \{x : U(x) \leq u_1\}$, $\mathcal{X}_2 = \{x : u_1 < U(x) \leq u_2\}$, ..., for some $\{u_i\}_{i=1}^m$.

We propose to simulate from

$$\varpi_{\Psi_\theta}(x) \propto \frac{\pi(x)}{\Psi_\theta^\zeta(U(x))},$$

where the importance weight $\Psi_\theta^\zeta(U(x))$ satisfies

How to construct the flattened distribution?

We partition the sample space $\mathcal{X}$ into m subregions based on $U(\boldsymbol{x})$: $\mathcal{X}_1 = \{\boldsymbol{x} : U(\boldsymbol{x}) \leq u_1\}$, $\mathcal{X}_2 = \{\boldsymbol{x} : u_1 < U(\boldsymbol{x}) \leq u_2\}$, ..., for some $\{u_i\}_{i=1}^m$.

We propose to simulate from

$$\varpi_{\Psi_{\boldsymbol{\theta}}}(\boldsymbol{x}) \propto \frac{\pi(\boldsymbol{x})}{\Psi_{\boldsymbol{\theta}}^{\zeta}(U(\boldsymbol{x}))},$$

where the importance weight $\Psi_{\boldsymbol{\theta}}^{\zeta}(U(\boldsymbol{x}))$ satisfies

$$\text{(i) } \Psi_{\boldsymbol{\theta}}(U(\boldsymbol{x})) = \sum_{i=1}^{m} \theta(i) 1_{u_{i-1} < U(\boldsymbol{x}) \leq u_i},$$

$$\text{(ii) } \theta(i) = \theta_\star(i), \text{where } \theta_\star(i) = \int_{\boldsymbol{\chi}_i} \pi(\boldsymbol{x}) d\boldsymbol{x} \text{ for } i \in \{1, 2, \cdots, m\}.$$

The sampler moves like a “random walk” in the space of energy.

How to learn $\theta_\star$ on the fly (I)

However, the extension of that idea to the Langevin kernel is not straightforward. The naive setup $\Psi_{\boldsymbol{\theta}}(U(\boldsymbol{x})) = \sum_{i=1}^{m} \theta(i) 1_{u_{i-1} < U(\boldsymbol{x}) \leq u_i}$ leads to $\frac{\partial \log \Psi_\theta(u)}{\partial u} = 0$ a.e and fails to simulate from a new density.

How to learn $\theta_\star$ on the fly (I)

However, the extension of that idea to the Langevin kernel is not straightforward. The naive setup $\Psi_{\boldsymbol{\theta}}(U(\boldsymbol{x})) = \sum_{i=1}^{m} \theta(i) 1_{u_{i-1} < U(\boldsymbol{x}) \leq u_i}$ leads to $\frac{\partial \log \Psi_\theta(u)}{\partial u} = 0$ a.e and fails to simulate from a new density.

To tackle this issue, we set $\Psi_{\boldsymbol{\theta}}(U(\boldsymbol{x}))$ as piecewise continuous:

$$\Psi_{\boldsymbol{\theta}}(U(\boldsymbol{x})) = \sum_{i=1}^{m} \left(\theta(i-1) e^{(\log \theta(i) - \log \theta(i-1)) \frac{U(\boldsymbol{x}) - u_{i-1}}{\Delta u}} \right) 1_{u_{i-1} < U(\boldsymbol{x}) \leq u_i},$$

How to learn $\theta_\star$ on the fly (I)

However, the extension of that idea to the Langevin kernel is not straightforward. The naive setup $\Psi_{\boldsymbol{\theta}}(U(x)) = \sum_{i=1}^{m} \theta(i) 1_{u_{i-1} < U(x) \leq u_i}$ leads to $\frac{\partial \log \Psi_\theta(u)}{\partial u} = 0$ a.e and fails to simulate from a new density.

To tackle this issue, we set $\Psi_{\boldsymbol{\theta}}(U(x))$ as piecewise continuous:

$$\Psi_{\boldsymbol{\theta}}(U(x)) = \sum_{i=1}^{m} \left(\theta(i-1) e^{(\log \theta(i) - \log \theta(i-1)) \frac{U(x) - u_{i-1}}{\Delta u}}\right) 1_{u_{i-1} < U(x) \leq u_i},$$

which leads to the desired gradient for the flattened distribution

$$\nabla_x \log \varpi_{\Psi_{\boldsymbol{\theta}}}(x) = -\left[1 + \zeta\tau \frac{\log \theta(J(x)) - \log \theta((J(x) - 1) \vee 1)}{\Delta u}\right] \frac{\nabla_x U(x)}{\tau}.$$

Contour SGLD: A scalable adaptive importance sampling

Sampling step Sample x_{k+1} using the SGLD algorithm

$$x_{k+1} = x_k - \epsilon_{k+1}\frac{N}{n}\left[1 + \zeta\tau\frac{\log\theta_k(\tilde{J}(x_k)) - \log\theta_k((\tilde{J}(x_k) - 1)\vee 1)}{\Delta u}\right]\nabla_x\widetilde{U}(x_k) + \sqrt{2\tau\epsilon_{k+1}}w_{k+1},$$

where $w_{k+1} \sim N(0, I_d)$, d is the dimension, ϵ_{k+1} is the learning rate. $\tilde{J}(x)$ corresponds to the index that satisfies $u_{\tilde{J}(x)-1} < \frac{N}{n}\widetilde{U}(x) \leq u_{\tilde{J}(x)}$.

Contour SGLD: A scalable adaptive importance sampling

Sampling step Sample x_{k+1} using the SGLD algorithm

$$x_{k+1} = x_k - \epsilon_{k+1}\frac{N}{n}\left[1 + \zeta\tau\frac{\log\theta_k(\tilde{J}(x_k)) - \log\theta_k((\tilde{J}(x_k)-1)\vee 1)}{\Delta u}\right]\nabla_x\widetilde{U}(x_k) + \sqrt{2\tau\epsilon_{k+1}}w_{k+1},$$

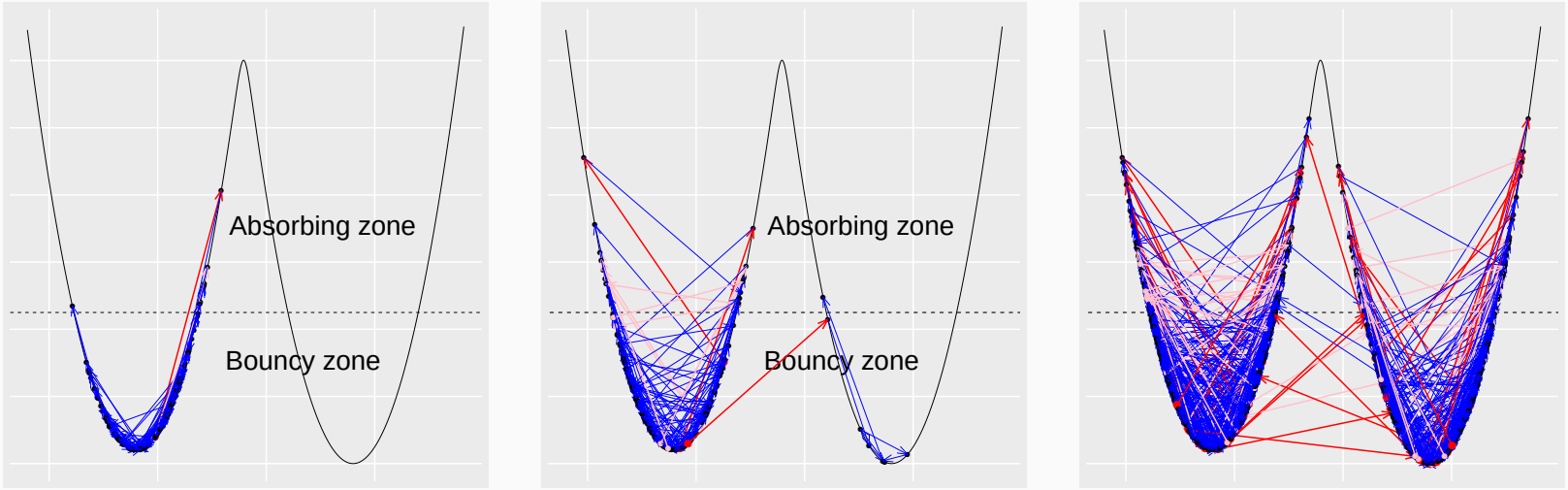
where $w_{k+1} \sim N(0, I_d)$, d is the dimension, ϵ_{k+1} is the learning rate. $\tilde{J}(x)$ corresponds to the index that satisfies $u_{\tilde{J}(x)-1} < \frac{N}{n}\widetilde{U}(x) \leq u_{\tilde{J}(x)}$.

Stochastic approximation Update the estimate of θ by setting

$$\theta_{k+1}(i) = \theta_k(i) + \omega_{k+1}\theta_k^{\zeta}(\tilde{J}(x_{k+1}))\left(1_{i=\tilde{J}(x_{k+1})} - \theta_k(i)\right),$$

where $1_{i=\tilde{J}(x_{k+1})}$ is an indicator function for $i = 1, 2, \ldots, m$.

A demo of Contour SGLD



(a) CSGLD paths (early) (b) CSGLD paths (mid) (c) CSGLD paths (late)

Figure 3: Sample trajectories of CSGLD.

Stability for the mean-field system

Lemma (Stability)

Given a small enough ϵ (learning rate), a large enough n (batch size) and m (partition number), there is a constant $\phi > 0$ such that the mean-field system $h(\boldsymbol{\theta})$ satisfies

$$\forall \boldsymbol{\theta} \in \boldsymbol{\Theta}, \langle h(\boldsymbol{\theta}), \boldsymbol{\theta} - \boldsymbol{\theta}_\star \rangle \leq -\phi \|\boldsymbol{\theta} - \boldsymbol{\theta}_\star\|^2 + \mathcal{O}\left(\epsilon + \frac{1}{m} + \delta_n(\boldsymbol{\theta})\right),$$

where $\delta_n(\cdot)$ is a bias, $\boldsymbol{\theta}_\star = (\int_{\mathcal{X}_1} \pi(\mathbf{x})d\mathbf{x}, \int_{\mathcal{X}_2} \pi(\mathbf{x})d\mathbf{x}, \ldots, \int_{\mathcal{X}_m} \pi(\mathbf{x})d\mathbf{x})$.

Convergence to a unique fixed point

Theorem (L^2 convergence rate)

Given mild assumptions, $\boldsymbol{\theta}_k$ converges to a unique fixed point $\boldsymbol{\theta}_\star$ regardless of the non-convexity of $U(\boldsymbol{x})$ due to the stability condition.

$$\mathbb{E}\left[\|\boldsymbol{\theta}_k - \boldsymbol{\theta}_\star\|^2\right] = \mathcal{O}\left(\omega_k + \sup_{i \geq k_0} \epsilon_i + \frac{1}{m} + \sup_{i \geq k_0} \delta_n(\boldsymbol{\theta}_i)\right).$$

Convergence to a unique fixed point

Theorem (L^2 convergence rate)

Given mild assumptions, $\boldsymbol{\theta}_k$ converges to a unique fixed point $\boldsymbol{\theta}_\star$ regardless of the non-convexity of $U(\boldsymbol{x})$ due to the stability condition.

$$\mathbb{E}\left[\|\boldsymbol{\theta}_k - \boldsymbol{\theta}_\star\|^2\right] = \mathcal{O}\left(\omega_k + \sup_{i \geq k_0} \epsilon_i + \frac{1}{m} + \sup_{i \geq k_0} \delta_n(\boldsymbol{\theta}_i)\right).$$

Convergence to a unique fixed point

Theorem (L^2 convergence rate)

Given mild assumptions, $\boldsymbol{\theta}_k$ converges to a unique fixed point $\boldsymbol{\theta}_\star$ regardless of the non-convexity of $U(\boldsymbol{x})$ due to the stability condition.

$$\mathbb{E}\left[\|\boldsymbol{\theta}_k - \boldsymbol{\theta}_\star\|^2\right] = \mathcal{O}\left(\omega_k + \sup_{i \geq k_0} \epsilon_i + \frac{1}{m} + \sup_{i \geq k_0} \delta_n(\boldsymbol{\theta}_i)\right).$$

Convergence to a unique fixed point

Theorem (L^2 convergence rate)

Given mild assumptions, $\boldsymbol{\theta}_k$ converges to a unique fixed point $\boldsymbol{\theta}_\star$ regardless of the non-convexity of $U(\boldsymbol{x})$ due to the stability condition.

$$\mathbb{E}\left[\|\boldsymbol{\theta}_k - \boldsymbol{\theta}_\star\|^2\right] = \mathcal{O}\left(\omega_k + \sup_{i \geq k_0} \epsilon_i + \frac{1}{m} + \sup_{i \geq k_0} \delta_n(\boldsymbol{\theta}_i)\right).$$

Convergence of the weighted averaging estimators

Theorem (Convergence of the weighted averaging estimators)

Given mild assumptions. For any bounded function f, we have

$$\left| \mathbb{E}\left[\frac{\sum_{i=1}^{k} \theta_i(\tilde{J}(x_i)) f(x_i)}{\sum_{i=1}^{k} \theta_i(\tilde{J}(x_i))} \right] - \int_{\boldsymbol{x}} f(x)\pi(dx) \right|$$

$$= \mathcal{O}\left(\frac{1}{k\epsilon} + \sqrt{\epsilon} + \sqrt{\frac{\sum_{i=1}^{k} \omega_k}{k}} + \frac{1}{\sqrt{m}} + \sup_{i \geq k_0} \sqrt{\delta_n(\boldsymbol{\theta}_i)} \right).$$

Remark: The numerical error is slightly worse than $\mathcal{O}\left(\frac{1}{k\epsilon} + \epsilon\right)$ in the standard SGLD algorithm. This is necessary as simulating from the flattened distribution $\varpi_{\Psi_{\boldsymbol{\theta}_\star}}$ may lead to exponential accelerations.

Convergence of the weighted averaging estimators

Theorem (Convergence of the weighted averaging estimators)

Given mild assumptions. For any bounded function f, we have

$$\left| \mathbb{E}\left[\frac{\sum_{i=1}^{k} \theta_i(\tilde{J}(x_i)) f(x_i)}{\sum_{i=1}^{k} \theta_i(\tilde{J}(x_i))} \right] - \int_{\boldsymbol{x}} f(x)\pi(dx) \right|$$

$$= \mathcal{O}\left(\frac{1}{k\epsilon} + \sqrt{\epsilon} + \sqrt{\frac{\sum_{i=1}^{k} \omega_k}{k}} + \frac{1}{\sqrt{m}} + \sup_{i \geq k_0} \sqrt{\delta_n(\boldsymbol{\theta}_i)} \right).$$

Remark: The numerical error is slightly worse than $\mathcal{O}\left(\frac{1}{k\epsilon} + \epsilon\right)$ in the standard SGLD algorithm. This is necessary as simulating from the flattened distribution $\varpi_{\Psi_{\boldsymbol{\theta}_\star}}$ may lead to exponential accelerations.