Hessian-Free High-Resolution Nesterov Acceleration For Sampling

Ruilin Li[1], Hongyuan Zha[2], Molei Tao[1*]

1: Georgia Institute of Technology
2:The Chinese University of Hong Kong, Shenzhen

ICML 2022

The Sampling Problem

Task: sampling from a statistical distribution:

- given: a probability density $Z^{-1}\rho(x)$, $x \in \mathbb{R}^d$

The Sampling Problem

Task: sampling from a statistical distribution:

- given: a probability density $Z^{-1}\rho(x)$, $x \in \mathbb{R}^d$
- goal: samples of r.v. with this density

The Sampling Problem

Task: sampling from a statistical distribution:

- given: a probability density $Z^{-1}\rho(x)$, $x \in \mathbb{R}^d$
- goal: samples of r.v. with this density

Relevance:

- stochastic optimization

The Sampling Problem

Task: sampling from a statistical distribution:

- given: a probability density $Z^{-1}\rho(x)$, $x \in \mathbb{R}^d$
- goal: samples of r.v. with this density

Relevance:

- stochastic optimization
- generative modeling

The Sampling Problem

Task: sampling from a statistical distribution:

- given: a probability density $Z^{-1}\rho(x)$, $x \in \mathbb{R}^d$
- goal: samples of r.v. with this density

Relevance:

- stochastic optimization
- generative modeling
- Bayesian inference

The Sampling Problem

Task: sampling from a statistical distribution:

- given: a probability density $Z^{-1}\rho(x)$, $x \in \mathbb{R}^d$
- goal: samples of r.v. with this density

Relevance:

- stochastic optimization
- generative modeling
- Bayesian inference
- optimal transport

The Sampling Problem

Task: sampling from a statistical distribution:

- given: a probability density $Z^{-1}\rho(x)$, $x \in \mathbb{R}^d$
- goal: samples of r.v. with this density

Relevance:

- stochastic optimization
- generative modeling
- Bayesian inference
- optimal transport
- ...

A First Algorithm: Langevin Monte Carlo

LMC (a.k.a. Unadjusted Langevin Algorithm)

Let $f = -\log \rho$. Use iteration

$$x_k = x_{k-1} - h\nabla f(x_{k-1}) + \sqrt{2h}\xi_k$$

with i.i.d. standard normal ξ_k. Output x_k for $k \gg 1$.

A First Algorithm: Langevin Monte Carlo

LMC (a.k.a. Unadjusted Langevin Algorithm)

Let $f = -\log \rho$. Use iteration

$$x_k = x_{k-1} - h\nabla f(x_{k-1}) + \sqrt{2h}\xi_k$$

with i.i.d. standard normal ξ_k. Output x_k for $k \gg 1$.

Relation with Optimization

A First Algorithm: Langevin Monte Carlo

LMC (a.k.a. Unadjusted Langevin Algorithm)

Let $f = -\log \rho$. Use iteration

$$x_k = x_{k-1} - h\nabla f(x_{k-1}) + \sqrt{2h}\xi_k$$

with i.i.d. standard normal ξ_k. Output x_k for $k \gg 1$.

Relation with Optimization

$x_k \quad = x_{k-1} - h\nabla f(x_{k-1}) + \sqrt{2h}\xi_k$

LMC = Gradient Descent + Appropriately Added Noise

Momentum Acceleration?

- **optimization**: momentum can accelerate the convergence.

Momentum Acceleration?

- **optimization**: momentum can accelerate the convergence.
- what about **sampling**?

Momentum Acceleration?

- **optimization**: momentum can accelerate the convergence.
- what about **sampling**?

continuous dynamics as a bridge for designing accelerated sampler

Momentum Acceleration?

- **optimization**: momentum can accelerate the convergence.
- what about **sampling**?

continuous dynamics as a bridge for designing accelerated sampler

infinitesimal learning rate limit

GD with momentum converges to, as $h \to 0$, an ODE

$$\begin{cases} \dot{q} &= p \\ \dot{p} &= -\gamma p - \nabla f(q) \end{cases}, \qquad \lim_{t \to \infty} q(t) = \text{local min of } f$$

Momentum Acceleration?

- **optimization**: momentum can accelerate the convergence.
- what about **sampling**?

continuous dynamics as a bridge for designing accelerated sampler

infinitesimal learning rate limit

GD with momentum converges to, as $h \to 0$, an ODE

$$\begin{cases} \dot{q} &= p \\ \dot{p} &= -\gamma p - \nabla f(q) \end{cases}, \qquad \lim_{t\to\infty} q(t) = \text{local min of } f$$

add noise appropriately:

$$\begin{cases} dq &= p dt \\ dp &= (-\gamma p - \nabla f(q)) dt + \sqrt{2\gamma} dW_t \end{cases}$$

Momentum Acceleration?

- **optimization**: momentum can accelerate the convergence.
- what about **sampling**?

continuous dynamics as a bridge for designing accelerated sampler

infinitesimal learning rate limit

GD with momentum converges to, as $h \to 0$, an ODE

$$\begin{cases} \dot{q} &= p \\ \dot{p} &= -\gamma p - \nabla f(q) \end{cases}, \qquad \lim_{t\to\infty} q(t) = \text{local min of } f$$

add noise appropriately:

$$\begin{cases} dq &= p dt \\ dp &= (-\gamma p - \nabla f(q)) dt + \sqrt{2\gamma} dW_t \end{cases}$$

$$q(t) \to Z^{-1}\exp(-f(q))dq \quad \text{under reasonable conditions}$$

Momentum Acceleration?

- **optimization**: momentum can accelerate the convergence.
- what about **sampling**?

continuous dynamics as a bridge for designing accelerated sampler

infinitesimal learning rate limit

GD with momentum converges to, as $h \to 0$, an ODE

$$\begin{cases} \dot{q} &= p \\ \dot{p} &= -\gamma p - \nabla f(q) \end{cases}, \qquad \lim_{t\to\infty} q(t) = \text{local min of } f$$

add noise appropriately:

$$\begin{cases} dq &= p dt \\ dp &= (-\gamma p - \nabla f(q))dt + \sqrt{2\gamma} dW_t \end{cases}$$

$$q(t) \to Z^{-1} \exp(-f(q))dq \quad \text{under reasonable conditions}$$

discretize: KLMC [Dalalyan & Riou-Durand 20], RMA [Shen & Lee 19], $\cdots$

Finite h However Has Positive Effect

Reality: h is not infinitesimal.

Finite h However Has Positive Effect

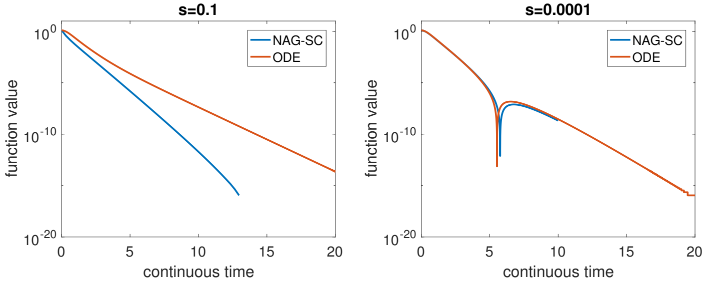
Reality: h is not infinitesimal.

Optimization: [Shi, Du, Jordan & Su 21]: GD with momentum (e.g., NAG-SC) is *faster* than its ODE limit when learning rate is $\nrightarrow 0$.

Finite h However Has Positive Effect

Reality: h is not infinitesimal.

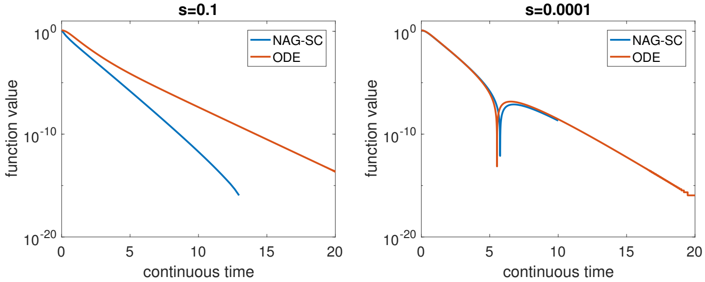
Optimization: [Shi, Du, Jordan & Su 21]: GD with momentum (e.g., NAG-SC) is *faster* than its ODE limit when learning rate is $\nrightarrow 0$.



Finite h However Has Positive Effect

Reality: h is not infinitesimal.

Optimization: [Shi, Du, Jordan & Su 21]: GD with momentum (e.g., NAG-SC) is *faster* than its ODE limit when learning rate is $\nrightarrow 0$.



<u>Goal</u>: exploit this finite h effect to further accelerate **Sampling**.

The Strategy

Turn NAG-SC **optimizer** with $\gg 0$ LR into a **sampler**:

The Strategy

Turn NAG-SC **optimizer** with $\gg 0$ LR into a **sampler**:

- View NAG-SC as a discretization of a high-resolution ODE.

The Strategy

Turn NAG-SC **optimizer** with $\gg 0$ LR into a **sampler**:

- View NAG-SC as a discretization of a high-resolution ODE.

NAG-SC:
$$\begin{cases} x_{k+1} & = y_k - s\nabla f(y_k) \\ y_{k+1} & = x_{k+1} + c(x_{k+1} - x_k) \end{cases}$$

The Strategy

Turn NAG-SC **optimizer** with $\gg 0$ LR into a **sampler**:

- View NAG-SC as a discretization of a high-resolution ODE.

NAG-SC: $$\begin{cases} x_{k+1} & = y_k - s\nabla f(y_k) \\ y_{k+1} & = x_{k+1} + c(x_{k+1} - x_k) \end{cases}$$

$$q_k = y_k, p_k = (y_k - x_k)/h, h = \sqrt{cs}, \gamma = \frac{1-c}{h}, \alpha = \frac{s}{h} \Longrightarrow$$

equivalent: $$\begin{cases} p_{k+1} = p_k - h\gamma p_k - h\nabla f(q_k) \\ q_{k+1} = q_k + hp_{k+1} - h\alpha\nabla f(q_k) \end{cases}$$

The Strategy

Turn NAG-SC **optimizer** with $\gg 0$ LR into a **sampler**:

- View NAG-SC as a discretization of a high-resolution ODE.

NAG-SC: $$\begin{cases} x_{k+1} &= y_k - s\nabla f(y_k) \\ y_{k+1} &= x_{k+1} + c(x_{k+1} - x_k) \end{cases}$$

$$q_k = y_k, p_k = (y_k - x_k)/h, h = \sqrt{cs}, \gamma = \tfrac{1-c}{h}, \alpha = \tfrac{s}{h} \Longrightarrow$$

equivalent: $$\begin{cases} p_{k+1} = p_k - h\gamma p_k - h\nabla f(q_k) \\ q_{k+1} = q_k + hp_{k+1} - h\alpha\nabla f(q_k) \end{cases}$$

It is a discretization of $$\begin{cases} \dot{q} = p - \alpha\nabla f(q) \\ \dot{p} = -\gamma p - \nabla f(q) \end{cases}$$

The Strategy

Turn NAG-SC **optimizer** with $\gg 0$ LR into a **sampler**:

- View NAG-SC as a discretization of a high-resolution ODE.

NAG-SC: $\begin{cases} x_{k+1} &= y_k - s\nabla f(y_k) \\ y_{k+1} &= x_{k+1} + c(x_{k+1} - x_k) \end{cases}$

$q_k = y_k, p_k = (y_k - x_k)/h, h = \sqrt{cs}, \gamma = \frac{1-c}{h}, \alpha = \frac{s}{h} \Longrightarrow$

equivalent: $\begin{cases} p_{k+1} = p_k - h\gamma p_k - h\nabla f(q_k) \\ q_{k+1} = q_k + hp_{k+1} - h\alpha\nabla f(q_k) \end{cases}$

It is a discretization of $\begin{cases} \dot{q} = p - \alpha\nabla f(q) \\ \dot{p} = -\gamma p - \nabla f(q) \end{cases}$

Note: unlike [Shi et al. 21], no Hess f needed.

The Strategy

Turn NAG-SC **optimizer** with $\gg 0$ LR into a **sampler**:

- View NAG-SC as a discretization of a high-resolution ODE

$$\begin{cases} \dot{q} = p - \alpha \nabla f(q) \\ \dot{p} = -\gamma p - \nabla f(q) \end{cases}$$

The Strategy

Turn NAG-SC **optimizer** with $\gg 0$ LR into a **sampler**:

- View NAG-SC as a discretization of a high-resolution ODE

$$\begin{cases} \dot{q} = p - \alpha \nabla f(q) \\ \dot{p} = -\gamma p - \nabla f(q) \end{cases}$$

- View α as a hyperparameter, no longer dependent on LR.

The Strategy

Turn NAG-SC **optimizer** with $\gg 0$ LR into a **sampler**:

- View NAG-SC as a discretization of a high-resolution ODE

$$\begin{cases} \dot{q} = p - \alpha \nabla f(q) \\ \dot{p} = -\gamma p - \nabla f(q) \end{cases}$$

- View α as a hyperparameter, no longer dependent on LR.
- Add noise appropriately:

$$\begin{cases} dq & = (p - \alpha \nabla f(q))dt + \sqrt{2\alpha} dW_t \\ dp & = (-\gamma p - \nabla f(q))dt + \sqrt{2\gamma} dB_t \end{cases}$$

s.t. $q(\infty) \sim Z^{-1} \exp(-f(q)) dq$.

The Strategy

Turn NAG-SC **optimizer** with $\gg 0$ LR into a **sampler**:

- View NAG-SC as a discretization of a high-resolution ODE

$$\begin{cases} \dot{q} = p - \alpha \nabla f(q) \\ \dot{p} = -\gamma p - \nabla f(q) \end{cases}$$

- View α as a hyperparameter, no longer dependent on LR.
- Add noise appropriately:

$$\begin{cases} dq & = (p - \alpha \nabla f(q))dt + \sqrt{2\alpha} dW_t \\ dp & = (-\gamma p - \nabla f(q))dt + \sqrt{2\gamma} dB_t \end{cases}$$

  s.t. $q(\infty) \sim Z^{-1} \exp(-f(q))dq$.
- Discretize time. 1st-order and RMA versions in the paper.

Results

- Non-asymptotic error bound $\Rightarrow$ theoretically guaranteed advantage

Results

- Non-asymptotic error bound $\Rightarrow$ theoretically guaranteed advantage
- Choice of α: theory and experiments

Results

- Non-asymptotic error bound $\Rightarrow$ theoretically guaranteed advantage
- Choice of α: theory and experiments
- Experiments: strongly convex, convex, and nonconvex f's.

Results

- Non-asymptotic error bound ⇒ theoretically guaranteed advantage
- Choice of α: theory and experiments
- Experiments: strongly convex, convex, and nonconvex f's.

Bayesian Neural Network Example