AMAGOLD:

Amortized Metropolis Adjustment for Efficient Stochastic Gradient MCMC

Ruqi Zhang, **A. Feder Cooper**, Christopher De Sa

June 2020

Cornell University

Setting the scene

Markov chain Monte Carlo (MCMC)

Stochastic gradient MCMC (SG-MCMC)

Metropolis-Hastings (M-H) correction

Setting the scene

Markov chain Monte Carlo (MCMC)

- Are a family of sampling methods popular in Bayesian inference
- Approximate computationally intractable posterior
- Depend on size of inference task's dataset

Stochastic gradient MCMC (SG-MCMC)

Metropolis-Hastings (M-H) correction

Setting the scene

Markov chain Monte Carlo (MCMC)

- Are a family of sampling methods popular in Bayesian inference
- Approximate computationally intractable posterior
- Depend on size of inference task's dataset

Stochastic gradient MCMC (SG-MCMC)

- Use subsampling to decouple from dataset size
- Provide speed-ups, but at the cost of introducing bias

Metropolis-Hastings (M-H) correction

Setting the scene

Markov chain Monte Carlo (MCMC)

- Are a family of sampling methods popular in Bayesian inference
- Approximate computationally intractable posterior
- Depend on size of inference task's dataset

Stochastic gradient MCMC (SG-MCMC)

- Use subsampling to decouple from dataset size
- Provide speed-ups, but at the cost of introducing bias

Metropolis-Hastings (M-H) correction

- Removes bias by rejecting fraction of Markov chain's transitions
- Reintroduces dependency on dataset size

Our contribution in a nutshell

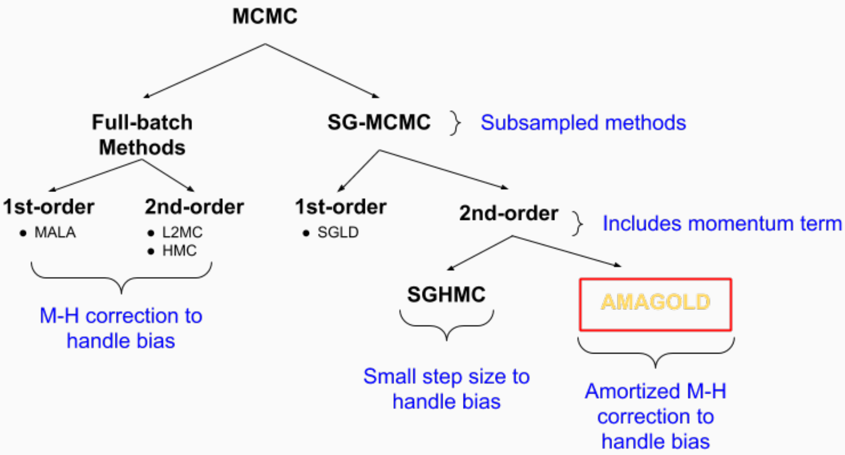
***Question**: Is there a way to construct an unbiased (i.e. exact) SG-MCMC algorithm that retains the efficiency we get from using stochastic gradients?*

***Answer**: Yes, which we demonstrate by introducing **AMAGOLD**:*

Exact without being prohibitively expensive

Uses M-H correction, amortizing its cost by applying it every T algorithm steps

Situating AMAGOLD within MCMC



Situating AMAGOLD within 2nd-order MCMC

Algorithm	Exact?	Stochastic Gradient?
AMAGOLD	Yes	Yes
L2MC	Yes	No
HMC	Yes	No
SGHMC	No	Yes

Bayesian inference

Given: Some dataset $\mathcal{D}$, domain Θ

Sample: From posterior distribution $\pi(\theta) \propto \exp(-U(\theta))$ where

$$U(\theta) = -\sum_{x \in \mathcal{D}} \log p(x|\theta) - \log p(\theta).$$

- $U(\theta)$: Energy function
- θ: Ranges over Θ
- $\pi \propto \mu$: π is the unique distribution with PDF proportional to μ

Second-order MCMC

A second-order chain (e.g. HMC, SGHMC, L2MC) *augments* state space with momentum r

Joint distribution:

$$\pi(\theta, r) \propto \exp(-H(\theta, r)) = \exp\left(-U(\theta) - \frac{1}{2\sigma^2}\|r\|^2\right),$$

H (Hamiltonian): measures total energy of system.

Stochastic gradient, second-order MCMC

Full-batch energy function (e.g. HMC, L2MC)

$$U(\theta) = -\sum_{x \in \mathcal{D}} \log p(x|\theta) - \log p(\theta).$$

Need M-H correction step to prevent bias due to discretization

Stochastic gradient energy function (e.g. SGHMC, AMAGOLD)

$$\tilde{U}(\theta) \approx -\frac{|\mathcal{D}|}{|\tilde{\mathcal{D}}|} \sum_{x \in \tilde{\mathcal{D}}} \log p(x|\theta) - \log p(\theta).$$

Naively using stochastic gradient estimate can lead to convergence to wrong stationary distribution

Stochastic gradient Hamiltonian Monte Carlo (SGHMC)

Algorithm 1 SGHMC

given: Energy U, initial state $\theta \in \Theta$, step size ϵ, momentum variance σ^2, friction β
loop
  optionally, resample momentum:
  $r \sim \mathcal{N}(0, \sigma^2)$
  initialize position and momentum:
  $r_{\frac{1}{2}} \leftarrow r$, $\theta_0 \leftarrow \theta$
  for $t = 1$ **to** T **do**
    position update: $\theta_t \leftarrow \theta_{t-1} + \epsilon \sigma^{-2} r_{t-\frac{1}{2}}$
    sample noise $\eta_t \sim \mathcal{N}(0, 4\epsilon\beta\sigma^2)$
    sample random energy component $\tilde{U}_t$
    update momentum:

$$r_{t+\frac{1}{2}} \leftarrow r_{t-\frac{1}{2}} - \epsilon \nabla \tilde{U}_t(\theta_t) - 2\epsilon\beta r_{t-\frac{1}{2}} + \eta_t$$

  end for
  new values: $(\theta, r) \leftarrow (\theta_T, r_{T+\frac{1}{2}})$
  ▷ no M-H step
end loop

Brief recap

Second-order MCMC

- Compute posterior distribution using sampling
- Include momentum term
- Full-batch variants (L2MC, HMC) and minibatch
- Minibatch variants (SGHMC, AMAGOLD)

Exact methods

- Guarantee convergence to correct stationary distribution (L2MC, HMC, AMAGOLD)
- Use M-H correction to remove bias

Inexact methods

- Do not have same convergence guarantees (SGHMC)

Exactness and reversibility

Detailed Balance Condition A Markov chain with transition probability operator G is reversible if for any pair of states x and y

$$\pi(x)G(x,y) = \pi(y)G(y,x).$$

Computing the M-H acceptance probability

$$\tau = \min\left(1, \frac{\pi(y)P(y,x)}{\pi(x)P(x,y)}\right).$$

Exactness and skew-reversibility

Given some measure-preserving involution over the state space denoted $x \mapsto x^\perp$, a chain G is *skew-reversible* if $\pi(x) = \pi(x^\perp)$ and

$$\pi(x)G(x, y) = \pi(y^\perp)G(y^\perp, x^\perp).$$

For Hamiltonian dynamics we use the involution that negates the momentum, i.e. $(\theta, r)^\perp = (\theta, -r)$.

AMAGOLD

AMAGOLD: **A**mortized **M**etropolis-**A**djusted stochastic **G**radient second-**O**rder **L**angevin **D**ynamics

Convergence rate intuition:

- Essentially equivalent to full-batch L2MC, up to a constant factor
- Approaches L2MC's rate as batch size increases or step size decreases

AMAGOLD's relationship to prior methods

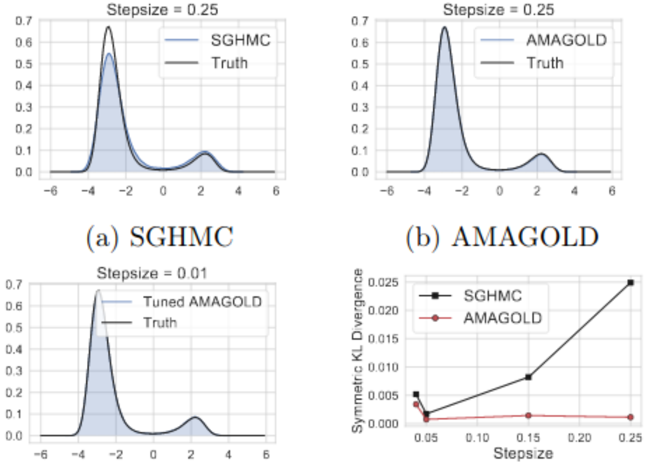
Algorithm	Exact?	Stochastic Gradient?
AMAGOLD	Yes	Yes
L2MC	Yes	No
HMC	Yes	No
SGHMC	No	Yes

Full-batch
$\rightarrow$ L2MC with AMA

Full-batch, $\beta = 0$, resample
$\rightarrow$ HMC

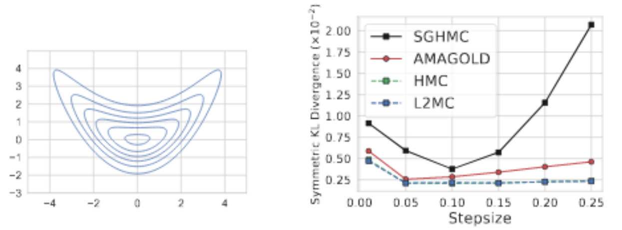
Disable M-H, adjust hyperparameters
$\rightarrow$ SGHMC

AMAGOLD in practice

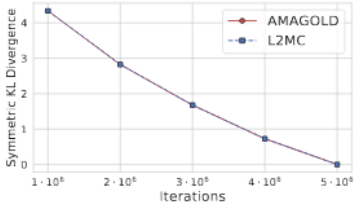


(a) SGHMC (b) AMAGOLD (c) Tuned AMAGOLD (d) KL Divergence

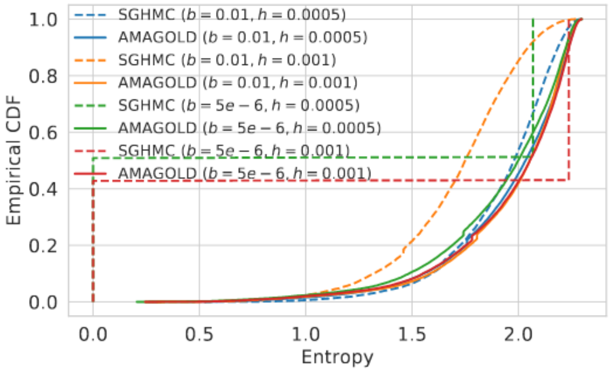
AMAGOLD on synthetic data



(a) Dist1 (b) KL comparison on Dist1



AMAGOLD on large-scale BNNs



Summary and future work