Optimization of Advanced Marko Chain Monte Carlo Methods with Applications to Biological Systems

Richard D. Paul

Forschungszentrum Jülich

April 7, 2021

Roadmap

- Experimentally assess optimal acceptance rates for model problems. Do theoretical results hold?
- How do non-identifiable dimensions affect the optimal acceptance rate and convergence?
- Combine Adaptive Metropolis with polytope samplers
- Is Adaptive Metropolis able to approximate optimal acceptances rates?
- Does Adaptive Metropolis work with non-identifiable dimensions?
- What other approaches exist there? Active subspaces [2]

Roadmap

- Experimentally assess optimal acceptance rates for model problems. Do theoretical results hold?
- How do non-identifiable dimensions affect the optimal acceptance rate and convergence?
- Combine Adaptive Metropolis with polytope samplers
- Is Adaptive Metropolis able to approximate optimal acceptances rates?
- Does Adaptive Metropolis work with non-identifiable dimensions?
- What other approaches exist there? Active subspaces [2]

Optimal acceptance rates

- Performance measures:
  - Effective Sample Size (ESS):

$$N_{eff} = \frac{N}{1+\tau}, \quad \tau \dots \text{ autocorrelation time}$$

  - Expected Squared Jump Distance (ESJD):

$$\text{ESJD} = \frac{1}{N-1} \sum_{i=1}^{N-1} \|\theta_n - \theta_{n+1}\|_2^2$$

  - Upper bound on Expected Squared Jump Distance per second (ESJD/s):

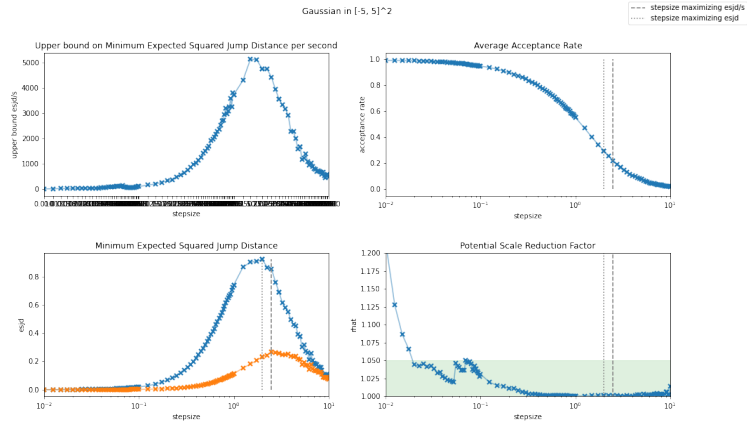
$$\text{ESJD/s} = \frac{\text{ESJD}}{t_\alpha r_\alpha + t_d}, \quad t_\alpha \dots \text{ time per simulation}, \quad r_\alpha \dots \text{ acceptance rate},$$
$$t_d \dots \text{ time per draw}$$

Optimal acceptance rates

- 4 chains where run until they hit convergence ($\widehat{R} < 1.05$) or maximal number of samples was exceeded
  - $\widehat{R}$ can be computed incrementally

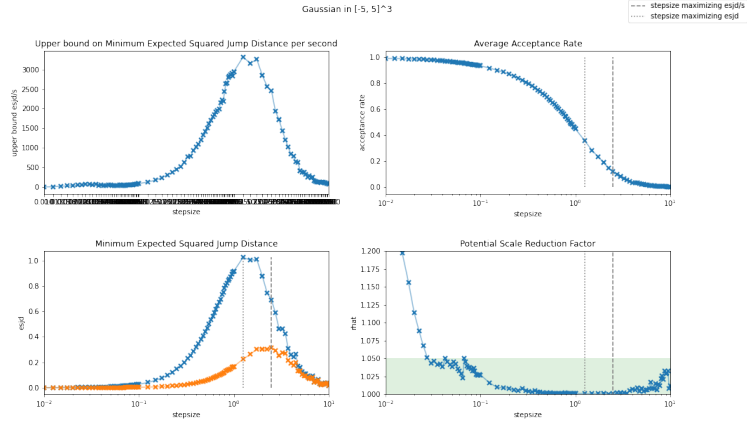
Optimal acceptance rates

- Orange upper bound on the ESJD/s



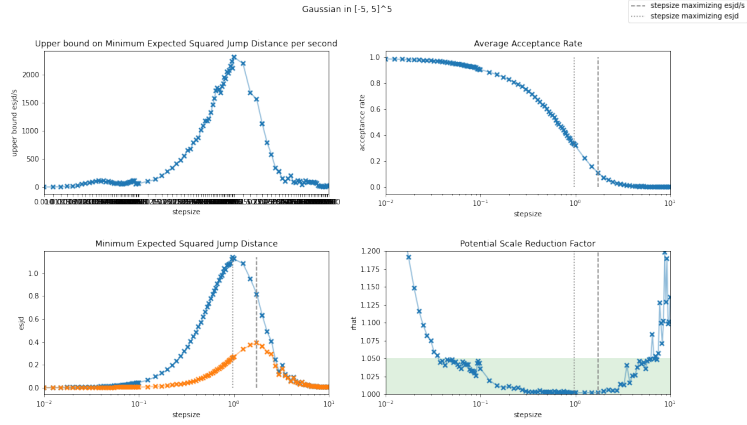
Optimal acceptance rates

- Orange upper bound on the ESJD/s



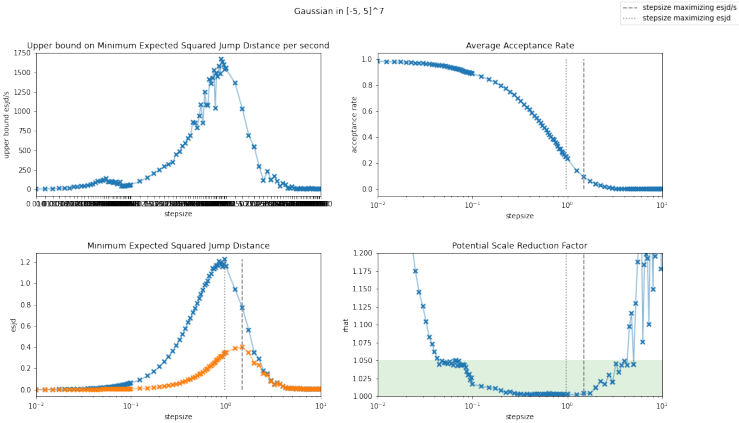
Optimal acceptance rates

- Orange upper bound on the ESJD/s



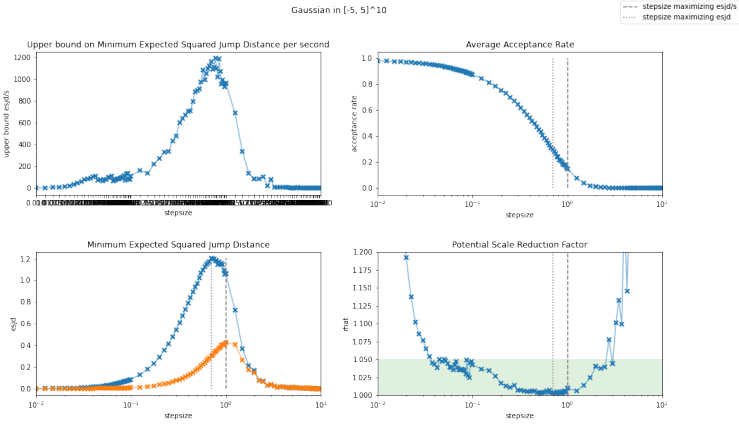
Optimal acceptance rates

- Orange upper bound on the ESJD/s



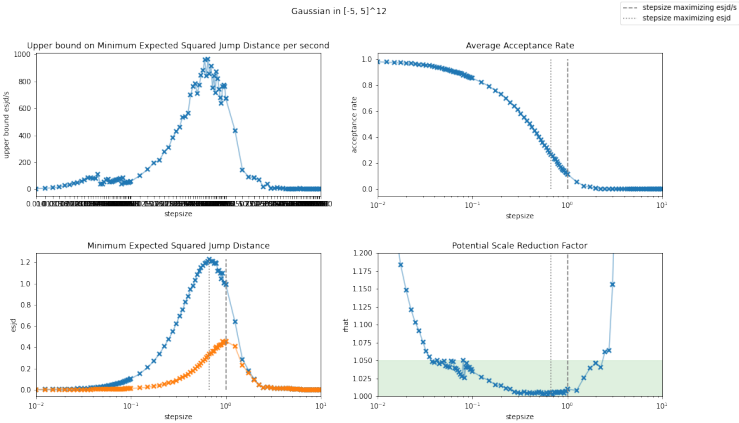
Optimal acceptance rates

- Orange upper bound on the ESJD/s



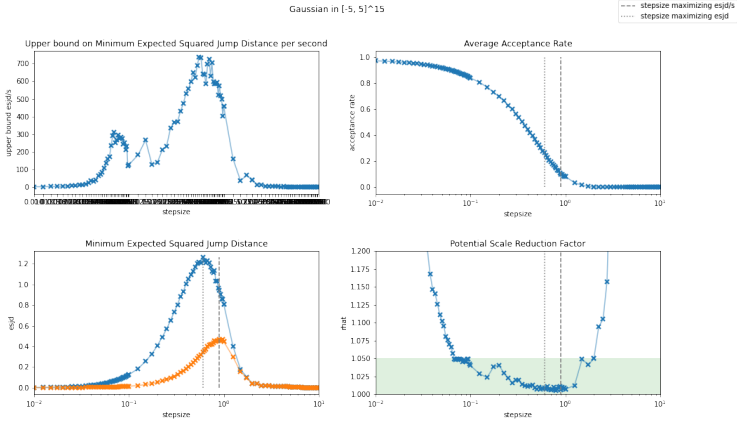
Optimal acceptance rates

- Orange upper bound on the ESJD/s



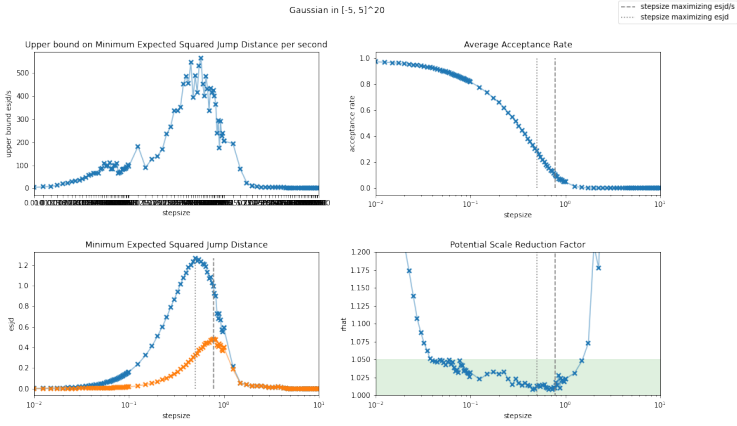
Optimal acceptance rates

- Orange upper bound on the ESJD/s



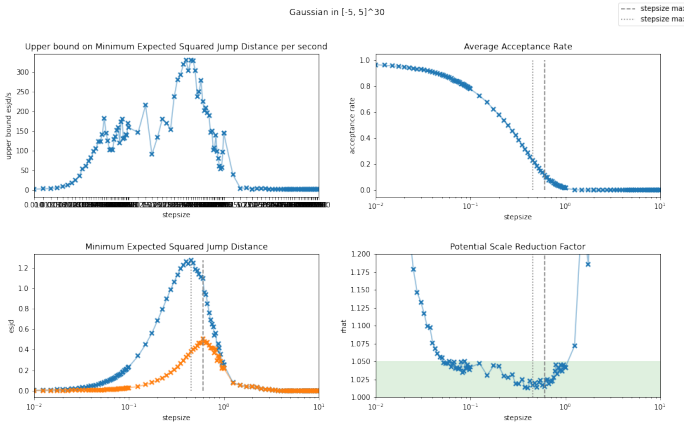
Optimal acceptance rates

- Orange upper bound on the ESJD/s



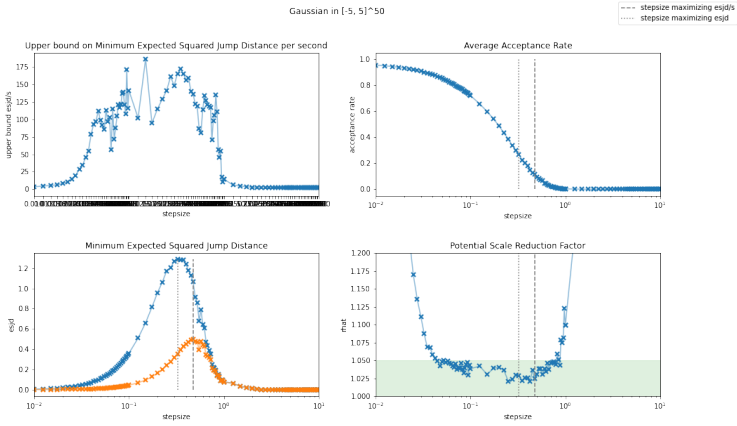
Optimal acceptance rates

- Orange upper bound on the ESJD/s



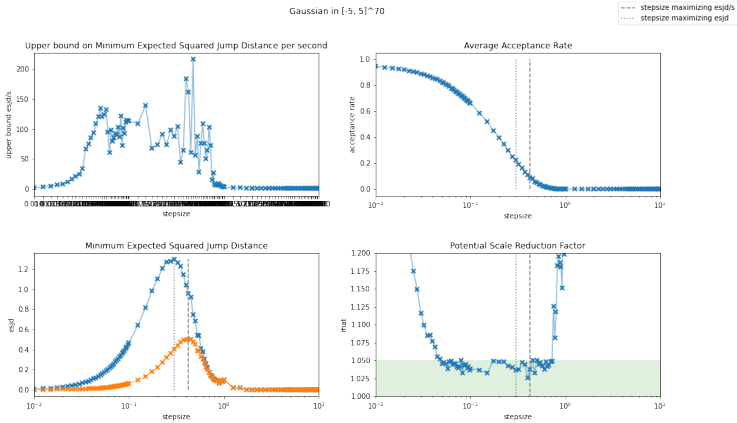
Optimal acceptance rates

- Orange upper bound on the ESJD/s



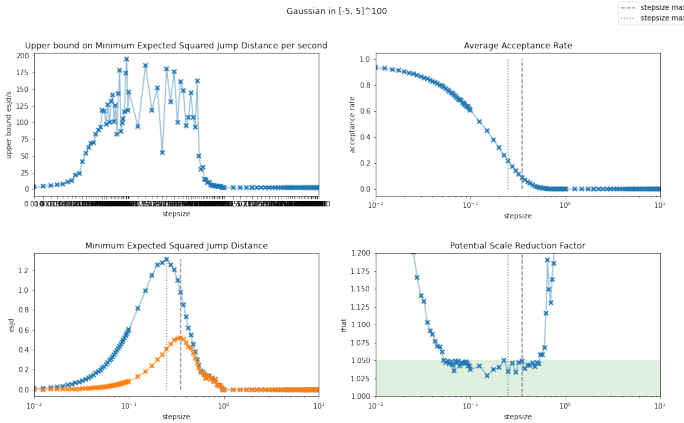
Optimal acceptance rates

- Orange upper bound on the ESJD/s



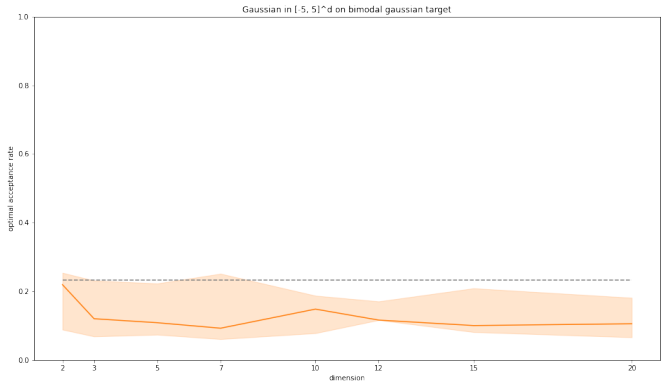
Optimal acceptance rates

- Orange upper bound on the ESJD/s



Optimal acceptance rates

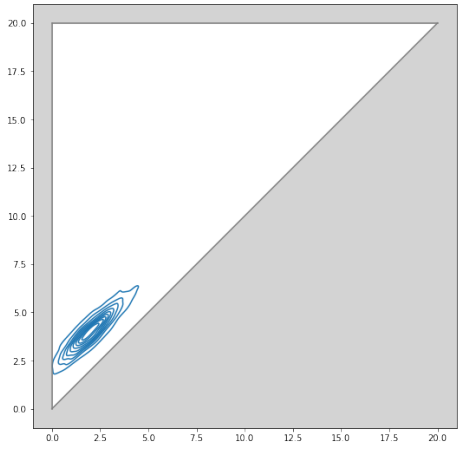
- Optimal acceptance rate per dimension



Optimal acceptance rates

- Justifies two things:
  - Use ESJD instead of ESS, as it is more robust, cheaper and seems like a good estimator
  - Tuning to theoretically optimal 0.234 does not hold, if we consider time costs & costly simulations
- Motivates: Offline tuning of the stepsize using ESJD

Offline tuning using ESJD



- First Approach:
  - Objective function draws N samples using stepsize s and compute ESJD.
  - Plug objective function in general-purpose optimizer
- Issue: Objective function is time-inhomogeneous and/or noisy
  - Low probability starting point errorneously favors large step sizes

$\rightarrow$ Possibly calls for *Bayesian optimization* approaches?

$\rightarrow$ Log-uniform sampling of stepsize parameters?

Starting point selection

- We want to optimize step size for the "interesting bits", so we first need to get there

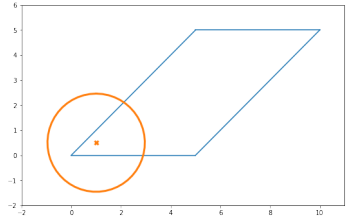
→ Typical warm start procedure is *burn in*

- Charles Geyer:
  "Any point you don't mind having in a sample is a good starting point."[1]

→ Use *maximum a posteriori/maximum likelihood* estimator by first optimizing the target distribution

[1] http://users.stat.umn.edu/ geyer/mcmc/burn.html

Proposal Preconditioning: Polytope Rounding

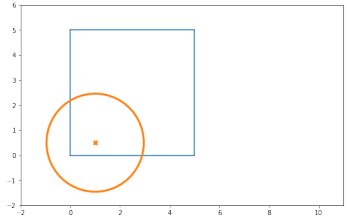


Polytope:

$$\mathcal{P} = \{x : Ax \leq b\}$$

Proposal:

$$x'_n = x_n + z, \quad z \sim \mathcal{N}(0, I)$$

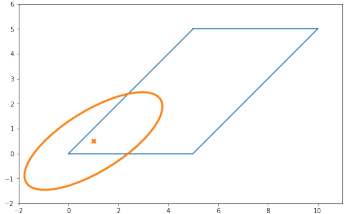


Polytope:

$$\mathcal{P} = \{x : ATx \leq b\}$$

Proposal:

$$x'_n = x_n + z, \quad z \sim \mathcal{N}(0, I)$$



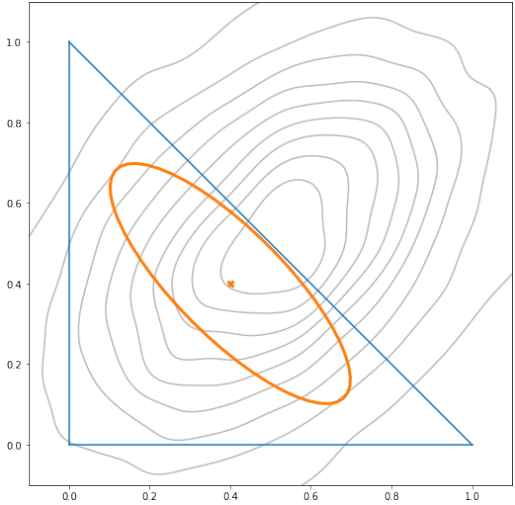
Polytope:

$$\mathcal{P} = \{y : Ay \leq b\}, \quad y = Tx$$

Proposal:

$$y'_n = y_n + Tz, \quad z \sim \mathcal{N}(0, I)$$

Proposal Preconditioning: Dikin Ellipsoid and/or Fisher Matrix



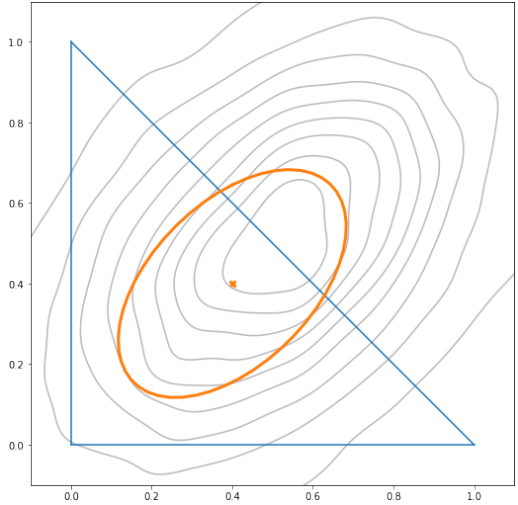
Dikin Walk [3]:

$$x'_n = x_n + \frac{r}{\sqrt{d}} D^{-1/2} z, \quad z \sim \mathcal{N}(0, I)$$

with

$$D = \sum_i \frac{a_i a_i^\top}{(b_i - a_i^\top x)^2}$$

Proposal Preconditioning: Dikin Ellipsoid and/or Fisher Matrix



CSmMALA (s=1) without drift [5]:

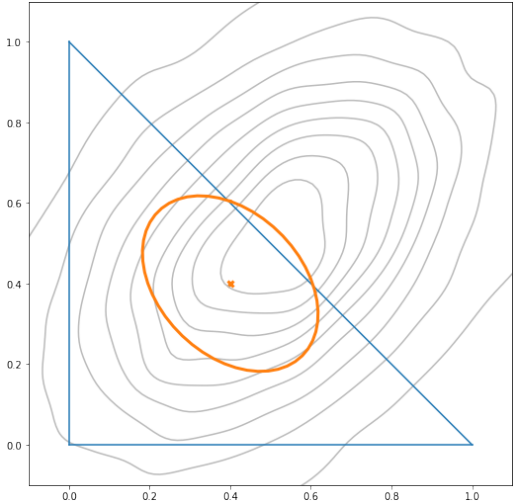
$$x'_n = x_n + rF^{-1/2}z, \quad z \sim \mathcal{N}(0, I)$$

with

$$F = -\mathbb{E}\Big[\frac{\partial^2}{\partial x^2}\log \mathcal{L}(d|x)\Big|x\Big],$$

with $\mathcal{L}$ being the likelihood function

Proposal Preconditioning: Dikin Ellipsoid and/or Fisher Matrix



CSmMALA (s=0.5) without drift [5]:

$$x'_n = x_n + rG^{-1/2}z, \quad z \sim \mathcal{N}(0, I)$$

with

$$G = sF + (1-s)D$$

with $\mathcal{L}$ being the likelihood function

Proposal Preconditioning: Dikin Ellipsoid and/or Fisher Matrix

- CSmMALA(s=0.5) 40 times higher ESS as the second best algorithm for a Gaussian target [5]
- CSmMALA takes two orders of magnitude more time for real ^{13}C data

$\rightarrow$ **Gradients of ^{13}C too expensive**

Local Approximations and Gradient Esimation

- Idea: Use local approximations of the target function, that are easy to evaluate
- *How do we construct such approximations?*

→ Conrad et al. have the answer! [1]

Local Approximations and Gradient Esimation

- Maintain set $\mathcal{S}$ of samples and their values, independently from chain samples
- Choose $N_L = \sqrt{d}(d+1)$ or $N_Q = \sqrt{d}(d+1)(d+2)/2$ points for linear or quadratic model respectively which are nearest to current state
- Fit model and cross-validate it on the N samples
- If error threshold is exceeded, add new point to $\mathcal{S}$

- Conrad et al. [1] replace exact target function with approximate one, thus sample the "wrong" distribution
- We only want to estimate the local gradient by using the fitted model's gradient
- Try updating the quadratic model with less points than N_Q by using least Frobenius norm update [4]

What to do with the Gradient Esimate?

- Accelerate CSmMALA
- Precondition Hit & Run algorithm with Fisher information estimate

What to do with the Gradient Esimate?

- Accelerate CSmMALA
- **Precondition Hit & Run algorithm with Fisher information estimate**

Fisher-preconditioned Hit & Run?

- CSmMALA takes convex combination of Dikin & Fisher information to remain inside polytope & adapt to target function
- Hit & Run is guaranteed to remain inside polytope!

→ Precondition it with Fisher information matrix to adapt to target function

- Hit & Run works best in rounded polytopes

→ **Pitfall:** Take care to *round* target function, i.e. evaluate it at the correct position

Roadmap

- Experimentally assess optimal acceptance rates for model problems. Do theoretical results hold?
- Offline stepsize tuning starting from MAP estimate
- Local approximation and gradient estimates for cheap preconditioning

Postponed:

- How do non-identifiable dimensions affect the optimal acceptance rate and convergence?
- Combine Adaptive Metropolis with polytope samplers
- Is Adaptive Metropolis able to approximate optimal acceptances rates?
- Does Adaptive Metropolis work with non-identifiable dimensions?
- What other approaches exist there? Active subspaces [2]

Thanks!

Conrad, P. R., Marzouk, Y. M., Pillai, N. S., and Smith, A.

Accelerating asymptotically exact mcmc for computationally intensive models via local approximations.
Journal of the American Statistical Association 111, 516 (2016), 1591–1607.

Constantine, P. G., Kent, C., and Bui-Thanh, T.

Accelerating markov chain monte carlo with active subspaces.
SIAM Journal on Scientific Computing 38, 5 (jan 2016), A2779–A2805.

Kannan, R., and Narayanan, H.

Random walks on polytopes and an affine interior point method for linear programming.
Mathematics of Operations Research 37, 1 (2012), 1–20.

Powell, M. J. D.

Least frobenius norm updating of quadratic models that satisfy interpolation conditions.
Mathematical Programming 100, 1 (May 2004), 183–215.

Theorell, A.

Bayesian methods for data-driven characterization of cells.
PhD thesis, Rheinisch-Westfälische Technische Hochschule Aachen, Aachen, 2019.
Veröffentlicht auf dem Publikationsserver der RWTH Aachen University 2020; Dissertation, Rheinisch-Westfälische Technische Hochschule Aachen, 2019.