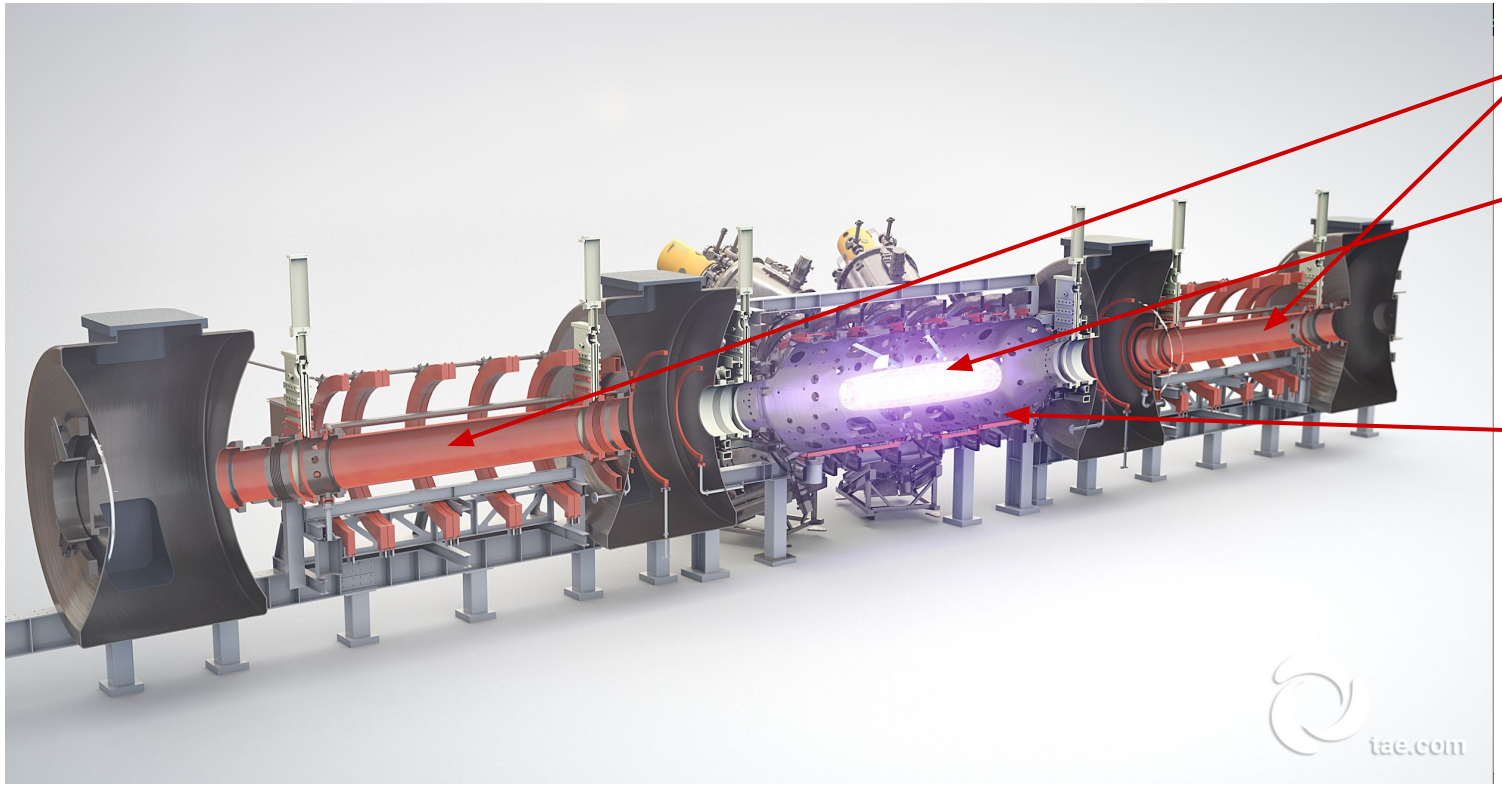
Hamiltonian Monte Carlo in Inverse Problems

Ill-conditioning and multi-modality

Ian Langmore (presenting)
Michael Dikovsky
Scott Geraedts
Peter Nordgaard
Rob von Behren

Inverse Problems

Norman: Experimental FRC Plasma Generator



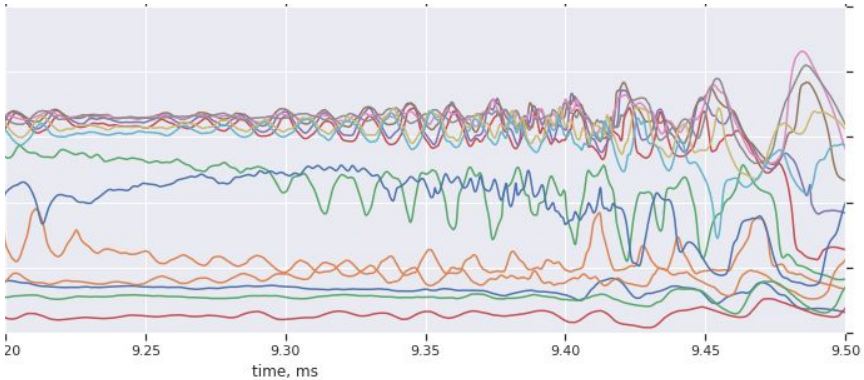
Plasma formed on each end, then fired into center vessel

Plasma confined by magnetic fields, heated/stabilized by neutral beams

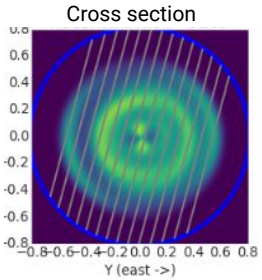
Ports provide access for measurement devices

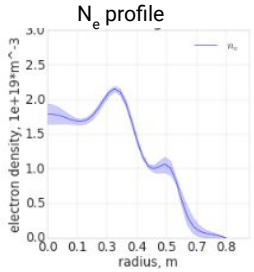
Goal: Learn to confine plasma long enough, at high enough temperatures, en route to net positive energy (in later machine)

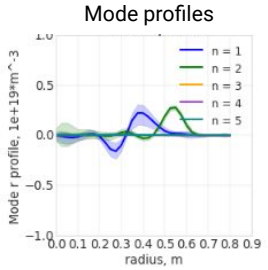
Measurements in → Reconstructed plasma out

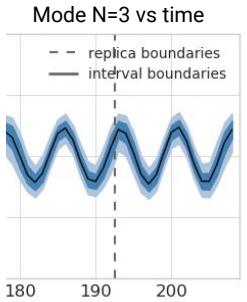


Interferometer traces









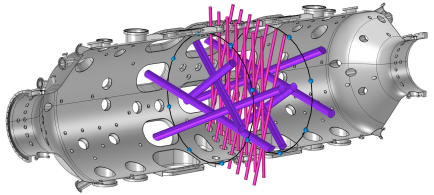
Dikovsky. 2021

Reconstruction Flow Chart

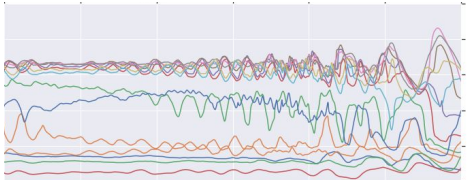
Plasma

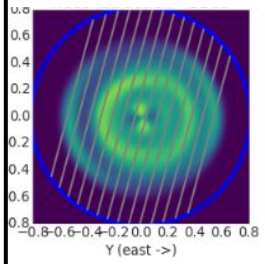
?

Interferometer



Measurements





Reconstructed Plasma

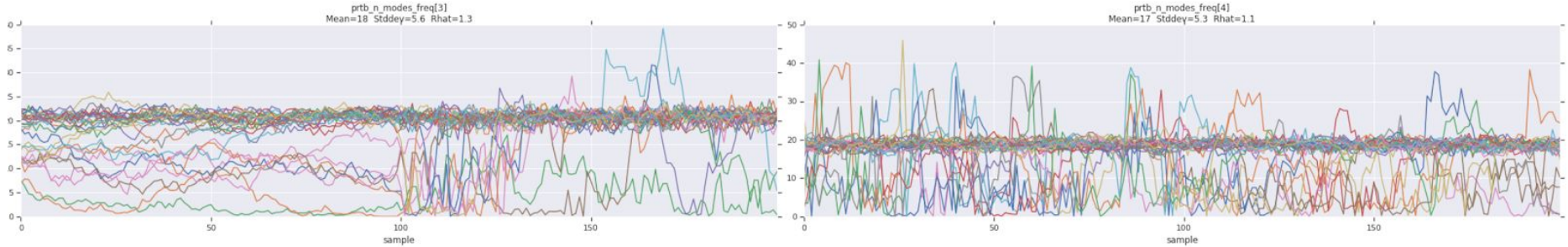
Prior Assumptions: Smoothness, not too large,...

(Interferometer)
Forward + Noise Model

$$m = AN_e + \sqrt{\sigma^2_{const} + \sigma^2_{prop} AN_e} \cdot \epsilon, \quad \text{with} \quad \epsilon \sim \mathcal{N}(0, I)$$

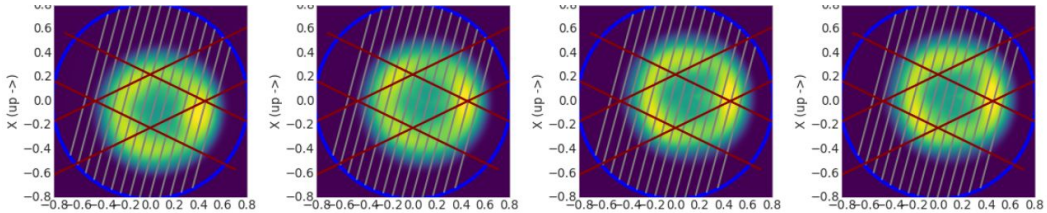
$$(AN_e)_i \approx C \int_{-1}^{1} N_e(x_i + \ell v) d\ell$$

Reconstructions are Samples of Plasmas



Samples of random variables

Map to samples of plasmas



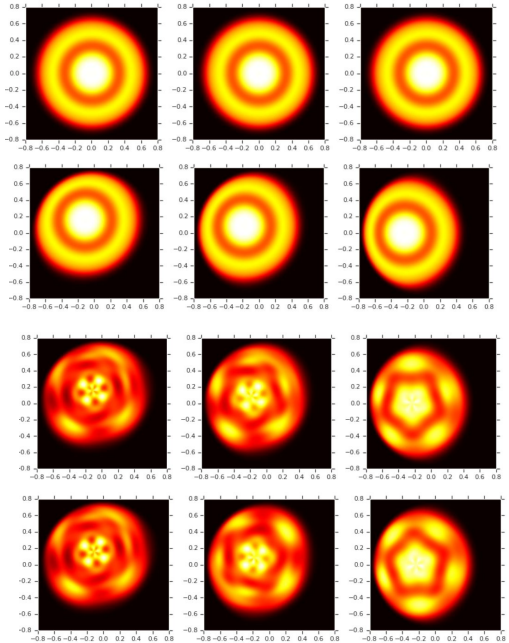
Model for Electron Density (N_e)

Our Prior is over these

$$N_e(r,\theta) = \log\left[1+\exp\left\{\sum_{k=1}^{K}\xi_k u_k(r)\right\}\right], \quad \text{where} \quad \sum_{k=1}^{\infty} u_k(r)u_k(r') \to \exp\left\{-\frac{|r-r'|^2}{2(0.15)^2}\right\}, \quad \text{and} \quad \xi_k \sim \mathcal{N}(0,(0.1)^2)$$

$$(r\cos\theta, r\sin\theta) \mapsto (r\cos\theta - \delta_x, r\sin\theta - \delta_y), \quad \text{where} \quad \delta_x, \delta_y \sim \mathcal{N}(0,(0.1)^2)$$

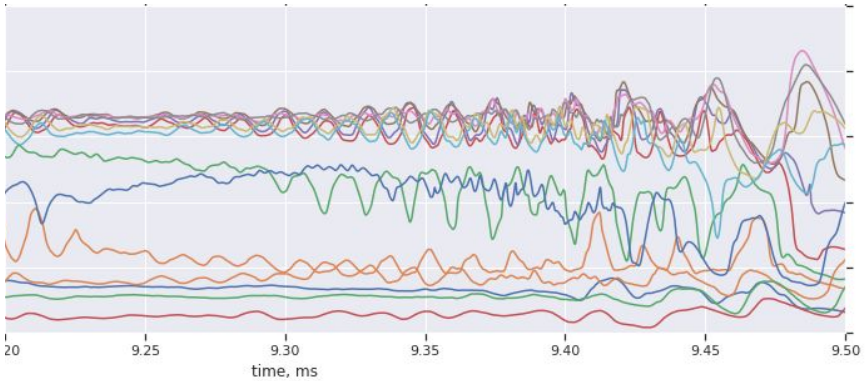
$$N_e(r,\theta) \mapsto N_e(r,\theta)\left[1+\text{Bound}_{(-1,1)}\left(\sum_{n=1}^{N}\eta_n \sin(n\theta)\right)\right], \quad \text{where} \quad \eta_n \sim \mathcal{N}(0, 1/n^2).$$



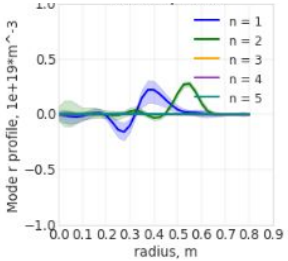
Now turn every random variable into a random process in time...

Do you need a Bayesian reconstruction?

Interferometer traces



Mode profiles



Gives uncertainty quantification.

UQ is meaningful only if…

- measurements + model ⇒ more than one acceptable solution
- your prior is reasonable

Helps determine structures too difficult to infer in your head

- e.g. if your measurements have many angles or states (temperature/density/…)
  - ...regularized optimization would do this too

Sampling Difficulties

Reparameterized Gaussian Inverse Problem

$$Y = A\varphi(X) + \epsilon$$

- $Y \in R^m$ is one measurement
- $A \in R^{m \times n}$ is a linear measurement operator
- φ is a nonlinear parameterization of the plasma
- $X \in R^n$ is the unknown
- ϵ is noise

Difficulties include

- Nonlinear parameterization + low noise ⇒ samples come from thin layer around complex surface
- Measurements are sparse and biased
- Parameterization often cannot fit the data

Gaussian Toy Problem

$$Y = AX + \epsilon$$

- $Y \in R^M$ is *one* measurement
- $X \in R^N$ are unknown parameters
- $\epsilon \sim Normal(0, \sigma^2 I_M)$

"Solve" the equation

$$X = A^{-1}(Y - \epsilon)$$

equivalently

$$X \sim Normal(A^{-1}Y, \sigma^2 (A^T A)^{-1})$$

Bayesian Inverse Problems = Solving equations with random coefficients

Not really a solution

- Doesn't work if A is singular
- Doesn't take priors into account

Nonetheless,

- Is the Bayesian solution with a "flat prior"
- Shows you need the right equation to get right answer

Gaussian Toy Problem : Bayesian Solution

$$Y = AX + \epsilon$$

- $Y \in R^M$ is *one* measurement
- $X \in R^N$ are unknown parameters
- $\epsilon \sim Normal(0, \sigma^2 I_M)$

The *forward model* implies a *likelihood*

$$p(y|x) \propto e^{-\|y-Ax\|^2/(2\sigma^2)}$$

Add prior knowledge

$$p(x) \propto e^{-x^T C_{pr}^{-1} x/2}$$

Get the (Gaussian) posterior

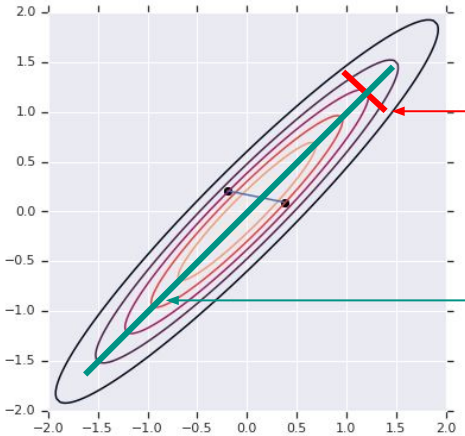
$$p(x|y) \propto p(x)p(y|x)$$

Ill-Conditioned Posterior Covariance

Posterior covariance

$$C = [C_{pr}^{-1} + \sigma^{-2} A^T A]^{-1}$$

Level sets of p(x) in 2-D



Measurements constrain some directions

Other directions are unconstrained

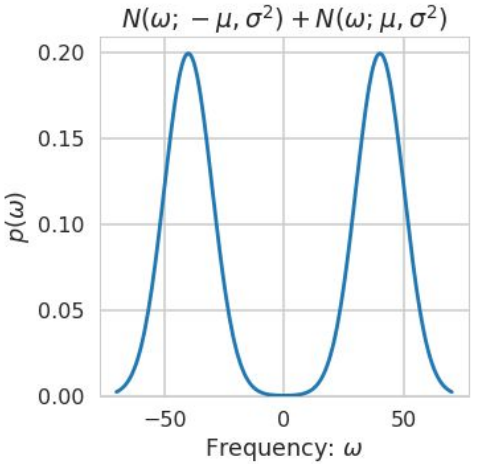
Ill-Conditioned Covariance

Sampling is slower

If $C = LL^T$, then sampling is slowed down (roughly) by the condition number of L

Multi-Modal Posteriors

Rotation direction is *not resolved* by Interferometer alone



Multi-Modality

Markov Chains get stuck

Probability of chain jumping between modes is

~ exp{-$(\mu/\sigma)^2$}

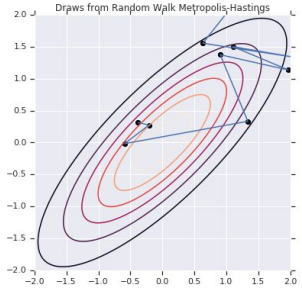
Hamiltonian Monte Carlo

Sampling: Random-Walk Metropolis Hastings

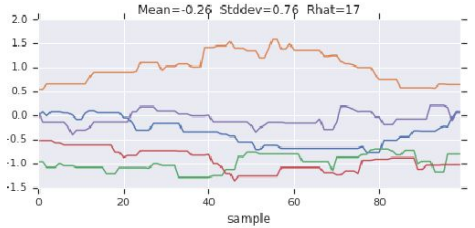
Metropolis Hastings recipe to sample from $p(z)$

1. Initialize $z = z^0$
2. Propose a move $z \to y \sim q(y|z)$
3. Accept with probability $\min\left\{1, \frac{q(z|y)p(y)}{q(y|z)p(z)}\right\}$
    1. If Accept, set $z^1 = y$
    2. If Reject, set $z^1 = z^0$
4. Iterate...

Random Walk Metropolis-Hastings if $q(y|z) \sim \mathcal{N}(y; z, \sigma^2 I)$ is Gaussian



5 chains sampling a 50-dimensional Gaussian:
Pictured is the first component

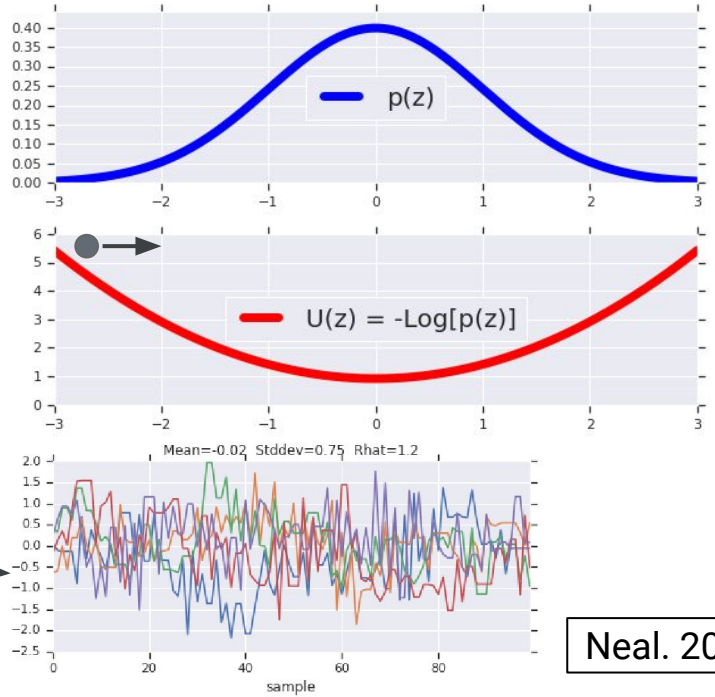


Random Walk behavior ⇒ Need (O(N)) evaluations of Log[p(z)] for each effective sample.

HMC: A proposal that scales well

HMC = MCMC with the following proposal:

1. Let *U(z)* := *-Log[p(z)]* define a surface in R^N
2. Start a ball at z^0
3. Give the ball a Gaussian “kick”
4. Let the ball roll for time T, giving you the proposal



If well tuned, the “rolling” allows the proposal to travel a long distance

⇒ Need $O(N^{1/4})$ evaluations of ∇ Log[p(z)] for each effective sample.

Neal. 2012

HMC: A few details

Increase dimension $\mathbb{R}^n \to \mathbb{R}^n \times \mathbb{R}^n$ by adding "momentum" ζ, and then...

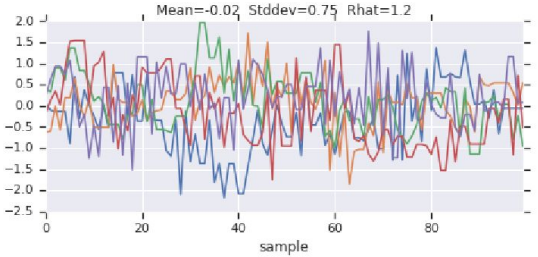
1. Initialize $(z, \zeta) = (z^0, \zeta^0)$, where $\zeta^0 \sim \mathcal{N}(0, I)$
2. Define $H(z, \zeta) := -\log p(z) + \|\zeta\|^2/2$
3. Propose $(z(T), \zeta(T))$, the time-T (numerical) solution to the initial value problem:

$$\dot{z}(t) = \frac{\partial H}{\partial \zeta}, \quad z(0) = z^0,$$

$$\dot{\zeta}(t) = -\frac{\partial H}{\partial z}, \quad \zeta(0) = \zeta^0,$$

4. Accept with probability

$$\min\left\{1, \exp\left\{H(z^0, \zeta^0) - H(z(T), \zeta(T))\right\}\right\}$$



If numerical integration were perfect, you would accept *every proposal*

This produces samples $[(z^0, \zeta^0), \ldots, (z^K, \zeta^k)]$ from $p(z, \zeta) \propto \exp\{-H(z, \zeta)\}$

The samples $(\zeta^0, \ldots, \zeta^K)$ may be discarded.

The samples $(z^0, \ldots, z^K)$ are from $p(z)$.

Neal. 2012

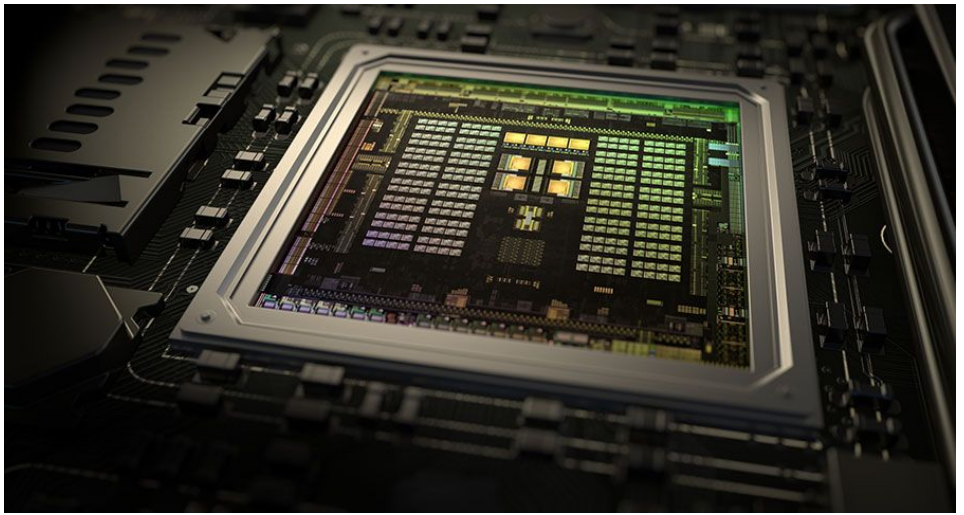
Barriers to using HMC in Inverse Problems

Need to compute gradients of the log prior/likelihood

- ➔ Auto-differentiating software “required”
  - ◆ requires writing forward model in TensorFlow/Jax/PyTorch/etc…
- ➔ GPUs highly recommended

Sampling may *still* be slow

- ➔ **Ill-Conditioning + Multi-modality** (this talk!)
  - ◆ ***All*** methods have issues with these



MCMC in an Industrial Research Setting

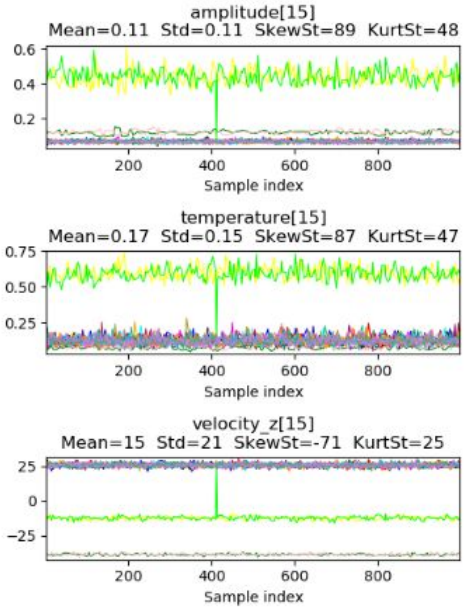
New experimental data arrives *daily*

- Reconstructions must be done/re-done for 1000's of experiments
- New experiments ⇒ new artifacts appear and old model may not work

Physicists modify the model *weekly*.

- Cannot tune code for each model : Require automatic parameter choices
- sampling code has to be fast/accurate
- sampling code should give informative answer

Figure: Stuck chains mean the correct values cannot be determined.



Ill-Conditioned Posterior Covariance

HMC Efficiency Tradeoff

Smaller numerical integration step size ⇒

- Lower integration error
- Higher Prob[Accept]

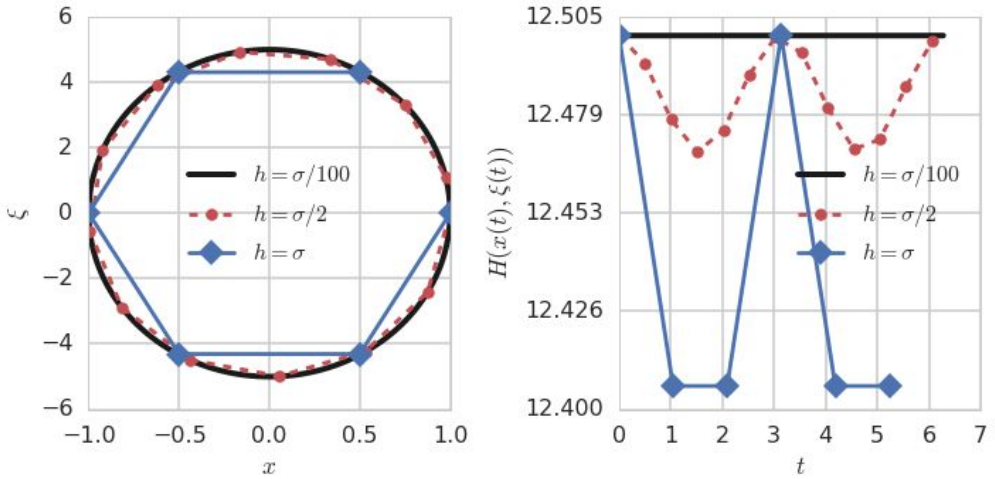
But also…

- number of steps needed

  ~ O(1 / step_size)

Rough best practice:

1. Adjust step_size until P[Accept] ≈ 0.68
2. Set num_leapfrog_steps ~ LargestScale / step_size

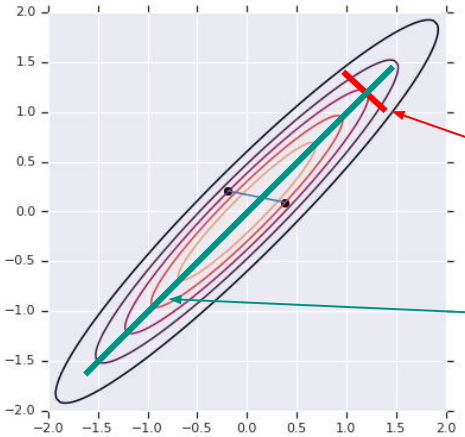


Integration error due to finite step size

Beskos. 2010

What are the optimal parameters?

Level sets of p(z) in 2-D



What are the optimal step size h^* and number of integration steps ℓ^*?

Assume...

- Gaussian target $p(z)$
- Eig(Covariance[Z]) is $\sigma_1^2 \geq \cdots \geq \sigma_N^2 > 0$

Then

- Must have stable integration along smallest scales
  $\Rightarrow h^* < 2\sigma_N$
- $T = h^* \ell^*$ must be large enough to traverse largest scales
  $\Rightarrow h^* \ell^* = c\sigma_1$

$$\Rightarrow \ell^* > \frac{c}{2} \frac{\sigma_1}{\sigma_N}$$

The correct asymptotics are

$$h^* \propto \left(\sum_{i=1}^{n} \frac{1}{\sigma_i^4} \right)^{-1/4},$$

Suggested measure of sampling effort

$$\ell^* \propto \kappa := \left(\sum_{i=1}^{n} \frac{\sigma_1^4}{\sigma_i^4} \right)^{1/4}$$

L. 2019

Preconditioning

To sample random variable X...

- Find diffeomorphism F so that $Z := F^{-1}(X) \approx \mathcal{N}(0, I)$
- Sample $Z_1, Z_2, \ldots$
- Transform back: $X_n = F(Z_n)$ to obtain $X_1, X_2, \ldots$

Generic Preconditioning

Linear Preconditioning

For example, if we estimate Covariance$(X) \approx C = LL^T$

- Set $Z = L^{-1}X$
- Covariance$(Z) \approx I$
- We hope $Z \approx \mathcal{N}(0, I)$

Preconditioning through a nonlinearity

To sample random variable X...

- Find diffeomorphism F so that $Z := F^{-1}(X) \approx \mathcal{N}(0, I)$
- Sample $Z_1, Z_2, \ldots$
- Transform back: $X_n = F(Z_n)$ to obtain $X_1, X_2, \ldots$

Generic Preconditioning

Quasi-Linear Preconditioning

Often our prior is a nonlinear function of a Gaussian. E.g.

- $X = G(W)$, where $W \sim \mathcal{N}(0, I)$

We remove this nonlinearity before computing covariance

- Estimate the factor L such that Covariance $(G^{-1}(X)) = LL^T$
- Set $Z = L^{-1}G^{-1}(X)$
- $\Rightarrow$ Covariance$(Z) \approx I$

Preconditioning with standard deviations

Let D = Diag(σ_1,...,σ_N) be the matrix of standard deviations

Proposition 4.1. *Suppose $C = LL^T$ is diagonally dominant with, for every i, $\sum_{j \neq i} |C_{ij}/C_{ii}| \leq \delta < 1$. Then,*

$$\kappa(D^{-1}L) \leq N^{1/4} \sqrt{\frac{1+\delta}{1-\delta}}.$$

Works well if C is diagonally dominant

Is close to the ideal diagonal preconditioner

Proposition 4.2. *Let D_{opt} be a preconditioner minimizing $\kappa(G^{-1}L)$ over all diagonal matrices G. Then,*

$$\kappa(D^{-1}L) \leq \sqrt{N}\, \kappa(D_{opt}^{-1}L).$$

Furthermore, if at most K entries in each row of LL^T are nonzero, then

$$\kappa(D^{-1}L) \leq \sqrt{K}\, \kappa(D_{opt}^{-1}L).$$

Precondition with *sample* standard deviations

Let D-hat be the (diagonal) matrix of ***sample*** standard deviations

Proposition 4.3. *Given* $\varepsilon, p \in (0,1)$,

$$\kappa(\hat{D}^{-1}L) \leq \kappa(D^{-1}L)\sqrt{\frac{1+\varepsilon}{1-\varepsilon}},$$

with probability p, *as soon as the number of i.i.d. samples* S *satisfies*

$$S \geq \frac{25}{\varepsilon^2}\log\left(\frac{3N}{p}\right).$$

Needs O(Log[N]) i.i.d. samples

Full Covariance Preconditioning → InvWishart

By *full covariance preconditioning*, we mean starting with the sample covariance, $\hat{C}$, factorizing as $\hat{C} = \hat{L}\hat{L}^T$, then preconditioning with $\hat{L}$.

Lemma 4.1. *Suppose* $(X^1, \ldots, X^S)$ *are i.i.d. samples of* $X \sim \mathcal{N}(0, C)$, *and we precondition sampling of* X *with the* S*-sample factor* $\hat{L}$. *Then, the preconditioned* κ *follows the law of* $\kappa(B)$, *for* $BB^T \sim InverseWishart(S, N)$.

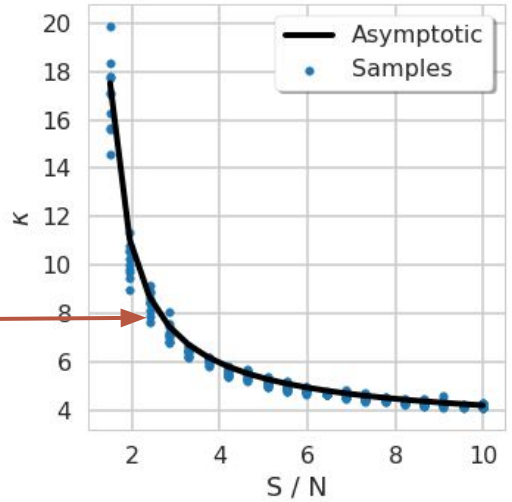
L. 2019

Asymptotic Expression for Kappa(InvWishart)

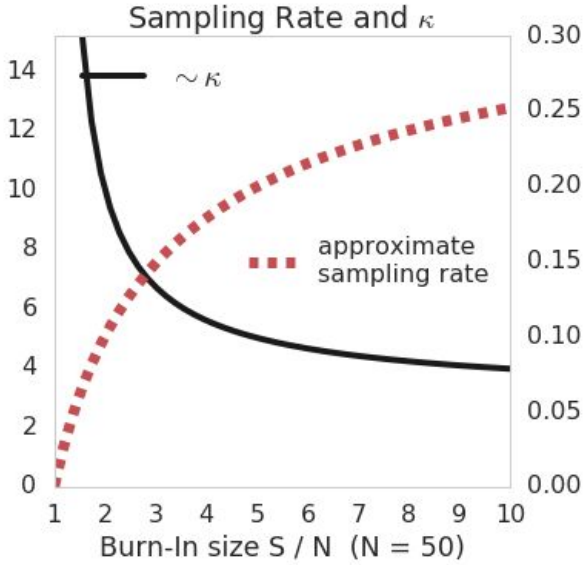
Proposition 4.4. *If* $BB^T \sim InverseWishart(N, S)$, *and* $N \to \infty$ *with* $S/N \to \omega \in (1, \infty)$, *then*

$$\frac{\kappa(B)}{N^{1/4}} \to \frac{(1+\omega^{-1})^{1/4}}{1-\omega^{-1/2}}$$

Need ~ 2.5 x N i.i.d. samples



Use Kappa to Decide on Burn-In Size



Algorithm To produce S_f *final* samples.

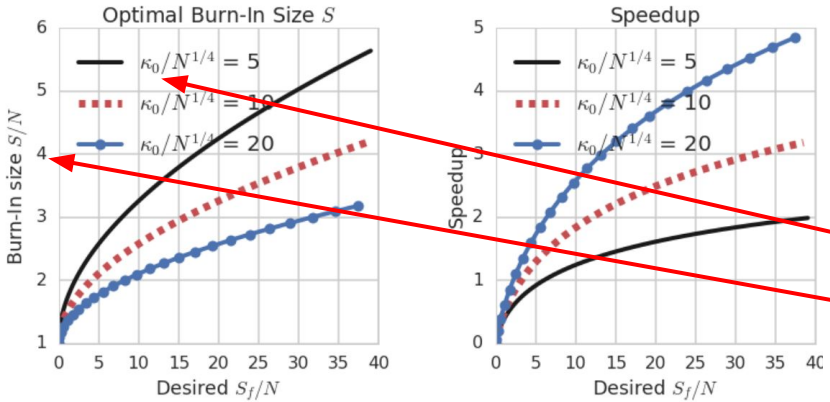
Before preconditioning, suppose $\kappa = \kappa_0$, then

1. Gather burn-in samples $(Z^1, \ldots, Z^S)$
2. Precondition with Cholesky(Covariance($(Z^1, \ldots, Z^S)$))
3. Gather S_f *final* samples

Assuming $seconds/ESS \propto \kappa(S)$, the **speedup** is

$$\frac{S_f \kappa_0}{S\kappa_0 + S_f \kappa(S)}$$

Optimal number of iid samples



Algorithm To produce S_f *final* samples.

Before preconditioning, suppose $\kappa = \kappa_0$, then

1. Gather burn-in samples $(Z^1, \ldots, Z^S)$
2. Precondition with Cholesky(Covariance($(Z^1, \ldots, Z^S)$))
3. Gather S_f *final* samples

Assuming $seconds/ESS \propto \kappa(S)$, the **speedup** is

$$\frac{S_f \kappa_0}{S\kappa_0 + S_f \kappa(S)}$$

To make this work, we must...

- Estimate κ_0
- Get i.i.d., samples

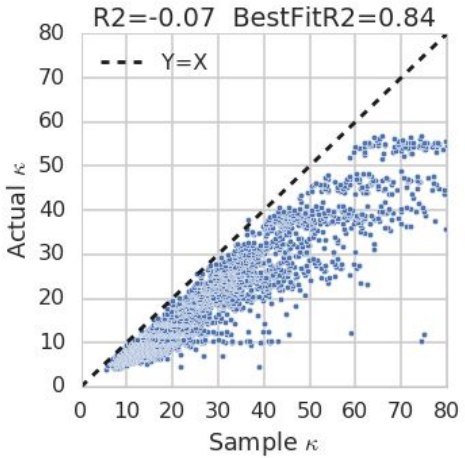
How to measure initial κ ?

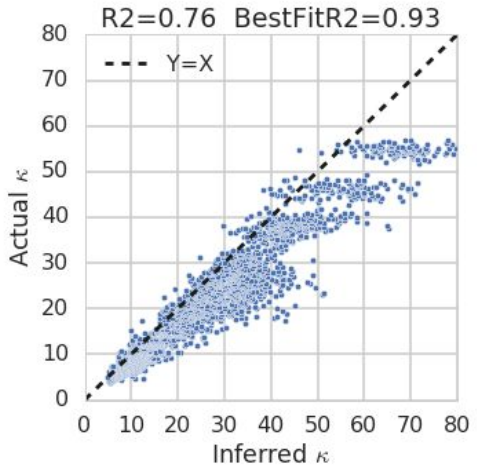
Don't use the sample covariance: It requires > N samples, and is not accurate

Instead, use a relation between Kappa, h, P[Accept]

Meaningful, even if non-Gaussian →

$$\kappa := \sigma_1 \nu \approx \frac{\sigma_1}{\bar{h}} 2^{7/4} \sqrt{\Phi^{-1}\left(1 - \frac{\bar{a}}{2}\right)}.$$



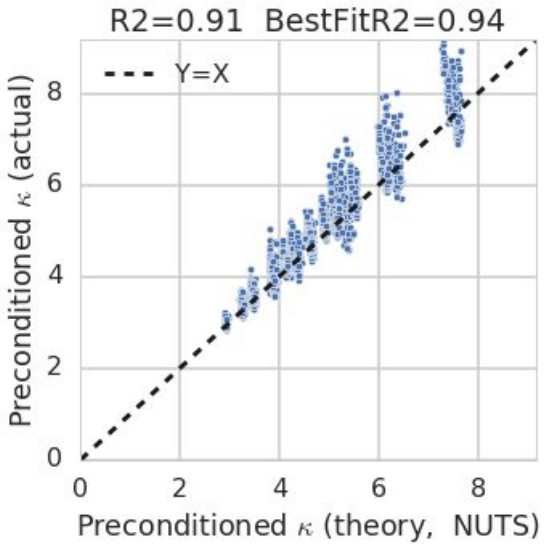


What to Use in Place of IID Samples?

What to use in place of i.i.d. samples?

⇒ Use NUTS samples, keep track of Mean[ESS]

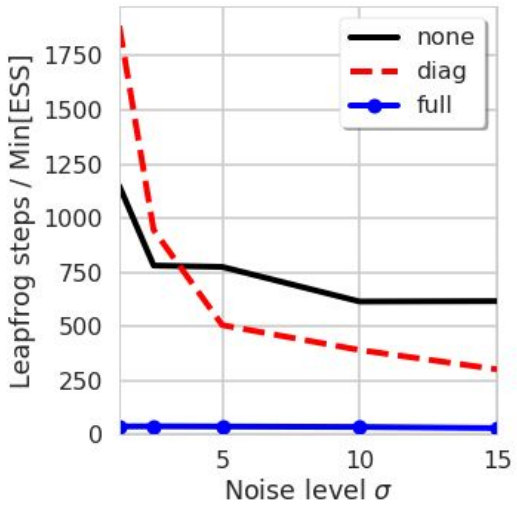
NUTS is slower – luckily, once we precondition we don't need NUTS



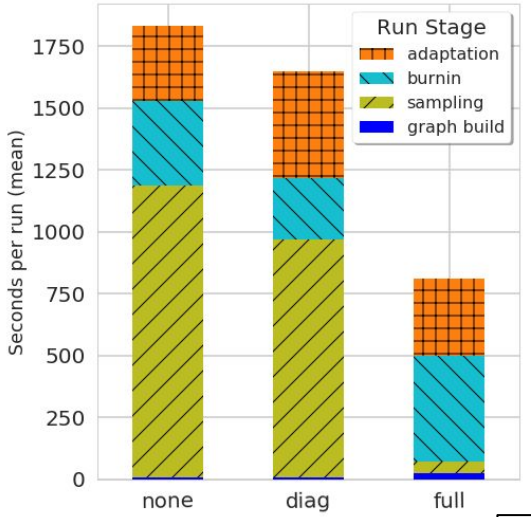
How much speedup do you get in practice?

Results on a non-Gaussian plasma reconstruction problem

Sampling speedup is 30x



Overall speedup is 2x

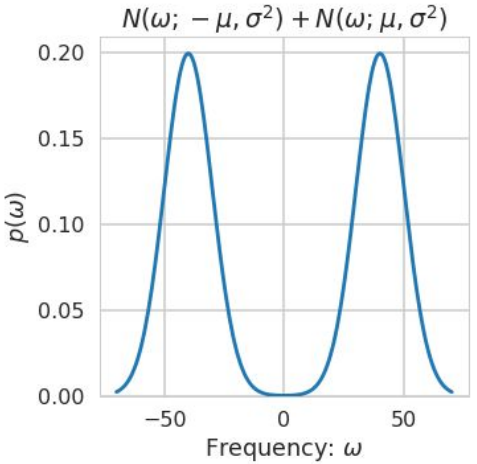


L. 2021

Multi-Modal Posteriors

Multi-Modal Posteriors

Rotation direction is *not resolved* by Interferometer alone



Multi-Modality

Markov Chains get stuck

Probability of chain jumping between modes is

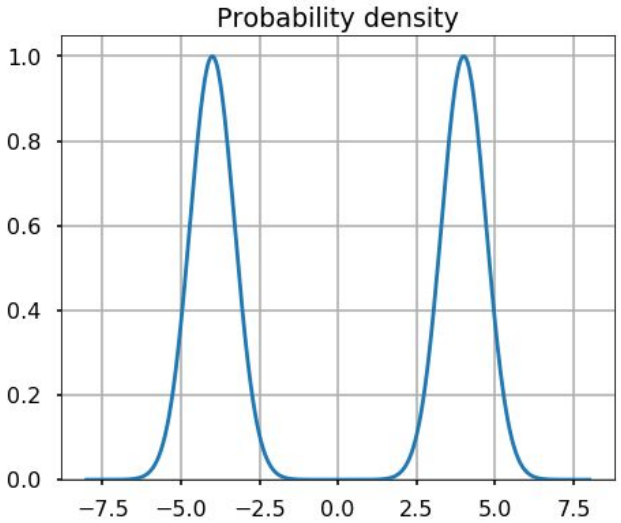
$\sim \exp\{-(\mu/\sigma)^2\}$

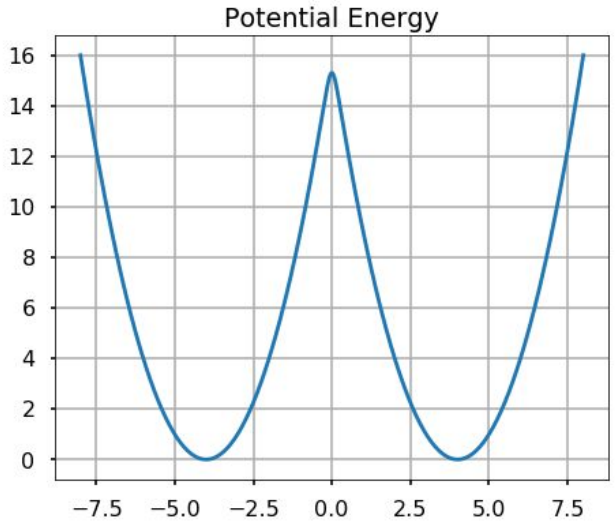
Mangoubi '18

Potential Energy Viewpoint

With p the *probability density* and U the *potential energy*:

$$p(x) = e^{-U(x)}$$

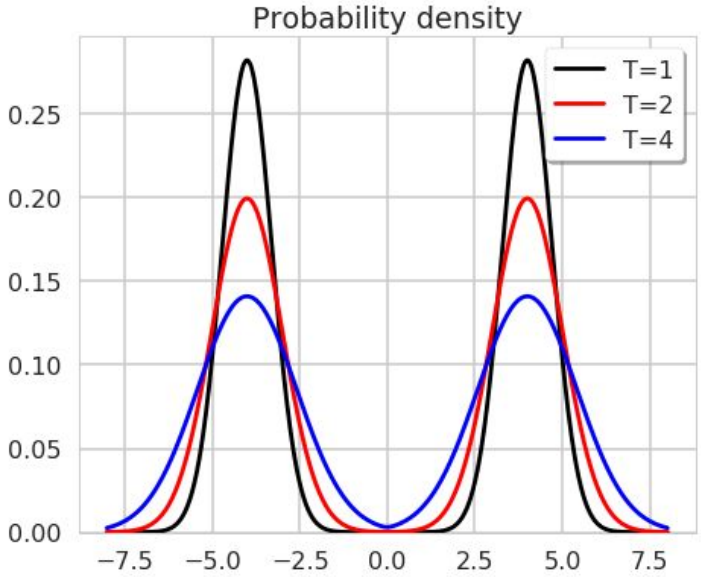


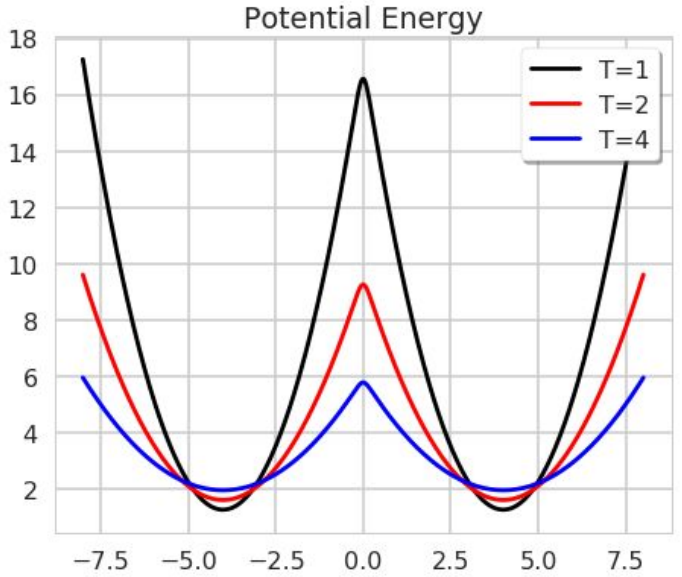


Tempered Densities in 1-D

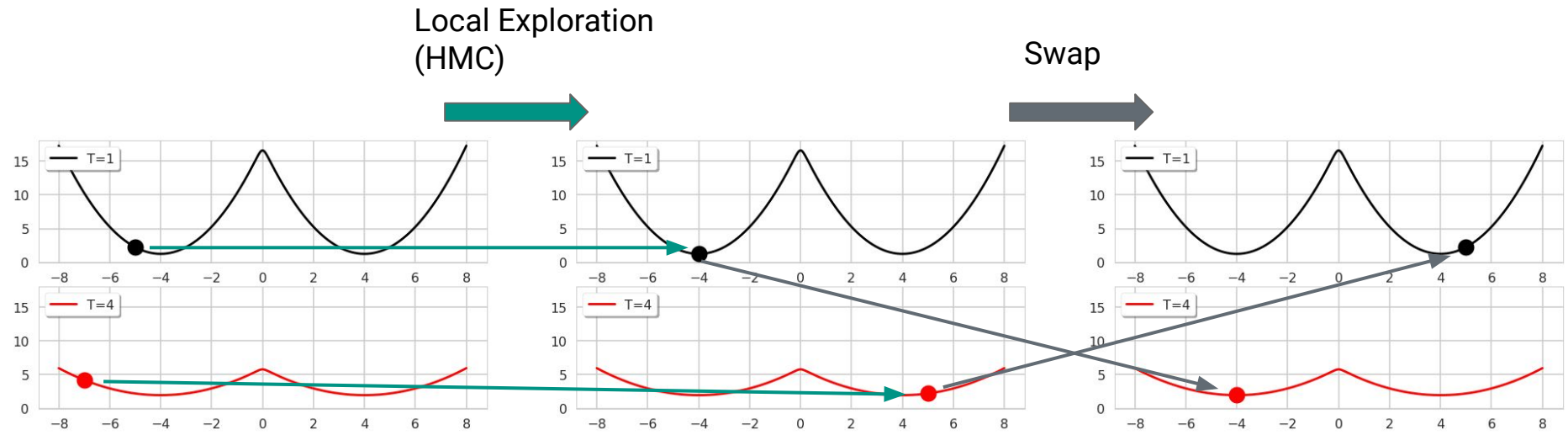
With p the *probability density*, U the *potential energy*, and T the *temperature*:

$$p_T(x) \propto e^{-U(x)/T}$$





Replica Exchange Monte Carlo (REMC)



Replica Exchange Monte Carlo (REMC)

1. Run R *replicas* in parallel
2. The r^{th} replica samples with temperature T_r
   a. $1 = T_0 < T_1 < \ldots < T_{R-1}$
3. The $T_0 = 1$ replica is the target
4. Replicas alternate *local exploration* and *swaps*

Swaps are accepted according to a **Metropolis condition**

$$\mathrm{P}[\mathrm{Swap}_{(1,2)} \mid x_1, x_2] = \min\left\{1, \alpha_{(1,2)(x_1,x_2)}\right\},$$

$$\alpha_{(1,2)} := \frac{\pi(x_2, x_1, x_3, \ldots)}{\pi(x_1, x_2, x_3, \ldots)} = \frac{\pi_1(x_2)\pi_2(x_1)}{\pi_1(x_1)\pi_2(x_2)}.$$

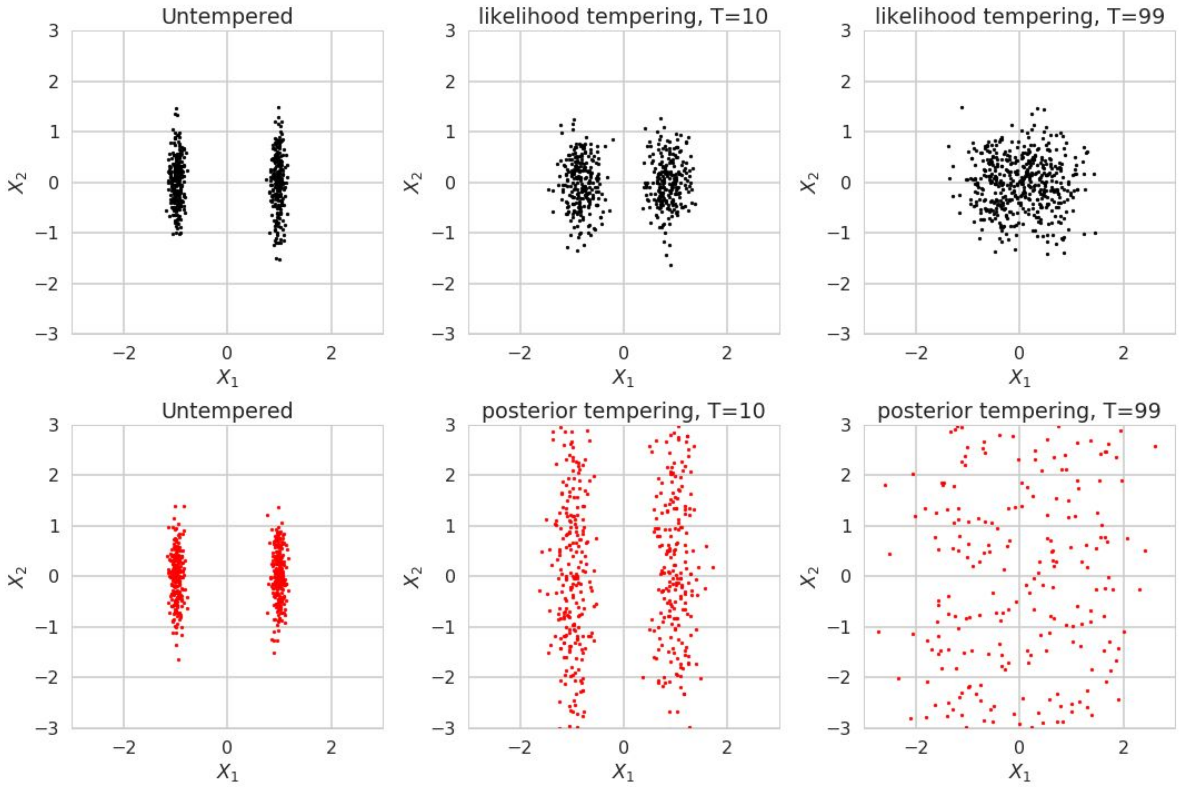
Likelihood Tempering & Posterior Tempering

Likelihood Tempering

$$p(x)p(y|x)^{1/T}$$

Posterior Tempering

$$p(x)^{1/T}p(y|x)^{1/T}$$



Likelihood Tempering: Better for Us

Posterior Tempering

$$p(x)^{1/T} p(y|x)^{1/T}$$

- Hottest replica ~ HUGE
  - explores beyond the prior
  - unstable as $T \to \infty$
- Requires $O(N^{1/2})$ replicas
  - N = dimension of unknown X
  - (for Gaussian)

Likelihood Tempering

$$p(x) p(y|x)^{1/T}$$

- Hottest replica ~ Prior so…
  - easy to sample from hottest replica
  - stable as $T \to \infty$
- Requires $O(\text{Min}(N^{1/2}, M^{1/2}))$ replicas
  - M = rank of fwd model
  - (for Gaussian)

Number of HMC Integration Steps can be Small

Since...

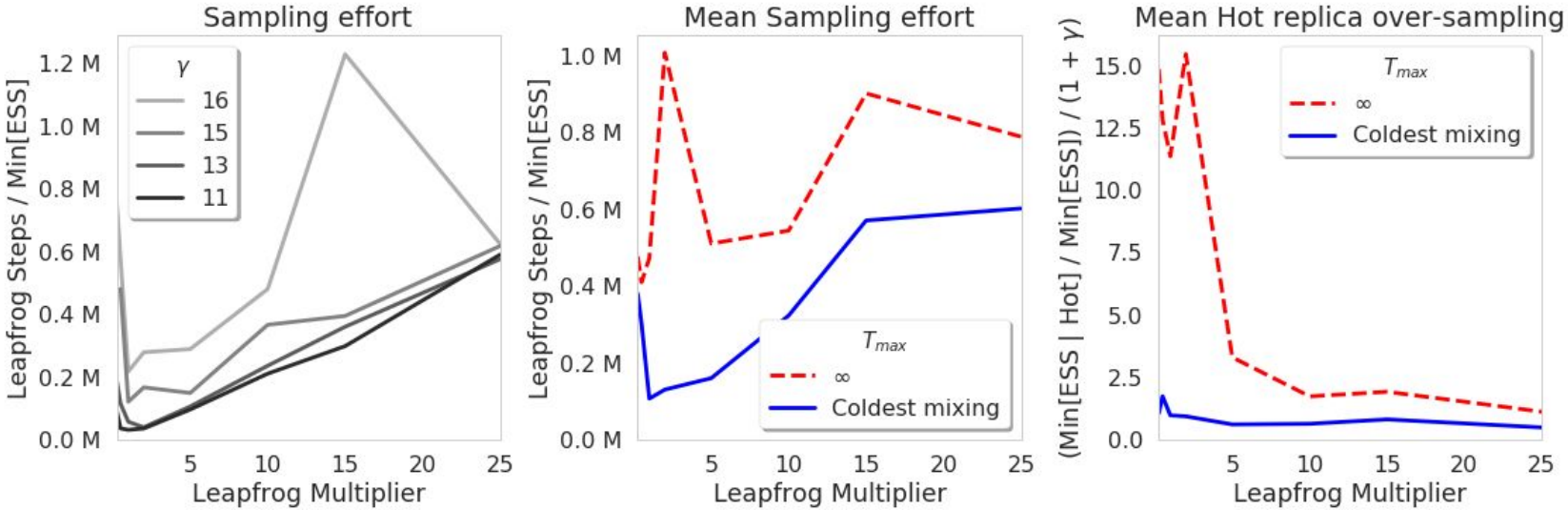
- on GPU, *every replica* should use same # integration steps
- the hottest replica needs to jump between modes – colder replicas can “mix by swapping”
- hottest replica is well conditioned, so requires fewer leapfrog steps
- about 1 / (1 + Λ) fraction of samples make it from hot to cold

We find...

- **can use $\text{NumSteps}^{\text{opt}}_{\text{hot}}$ / Sqrt(1 + Λ)** integration steps for *all* replicas
  - $\text{NumSteps}^{\text{opt}}_{\text{hot}}$ is the optimal number of steps for the hottest replica, without swapping
- between every “hot to cold” sample, hottest replica travels distance ~ h_{hot} $\text{NumSteps}^{\text{opt}}_{\text{hot}}$
  - this is the ideal distance

REMC on a Spectroscopy Problem

Most efficient when we use heuristic on last slide (leapfrog multiplier = 1)



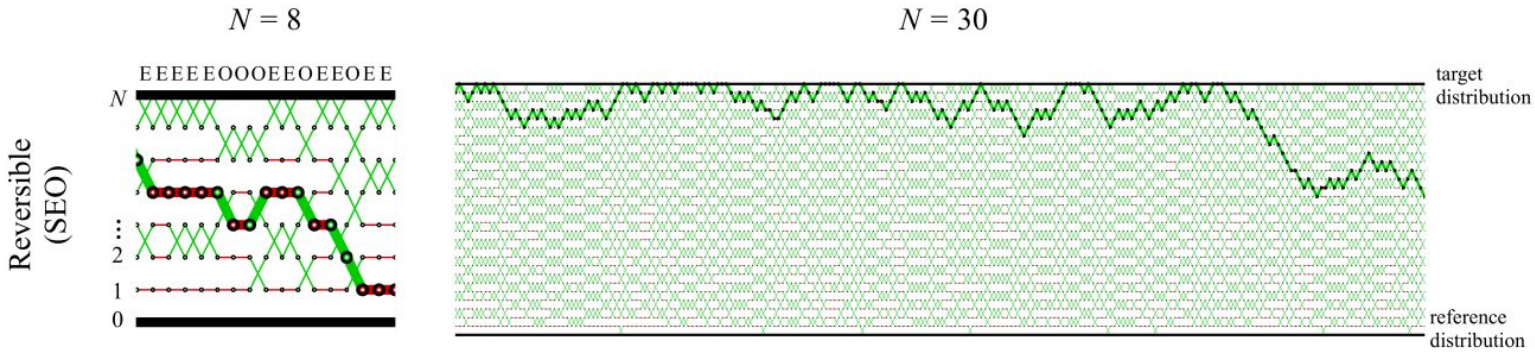
L. 2021

Random Walk Swap Proposals

Randomly propose either…

- *Even swaps*: Swap(1, 2) AND Swap(3, 4) AND …
- *Odd swaps:* Swap(2, 3) AND Swap(4, 5) AND …

As number of replicas R increases, only 1 / R proposals make it from “hot to cold”



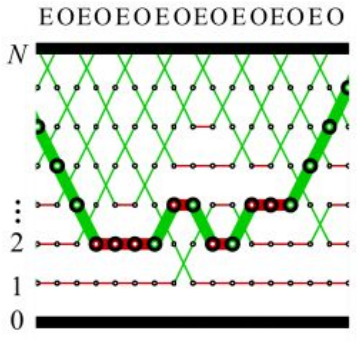
Deterministic Even-Odd

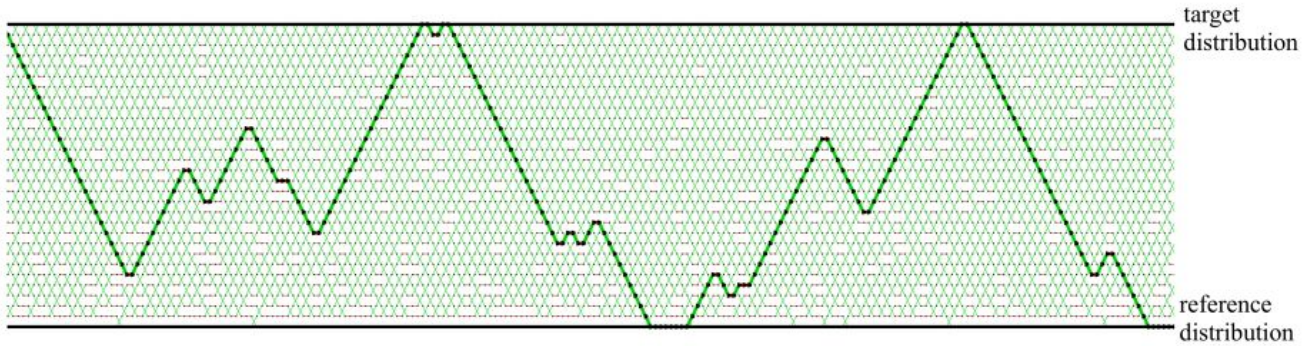
Deterministically alternate even & odd swaps

- Less likely to change direction
- As R → ∞,

As number of replicas R increases, 1 / (1 + Λ) fraction of proposals make it from hot to cold

Non-reversible (DEO)

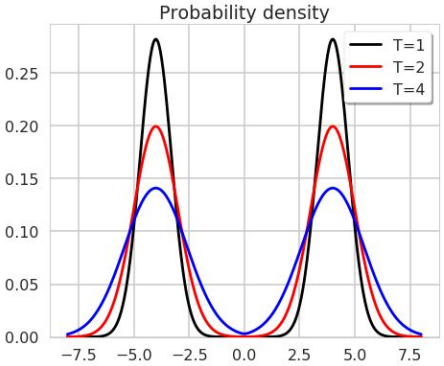


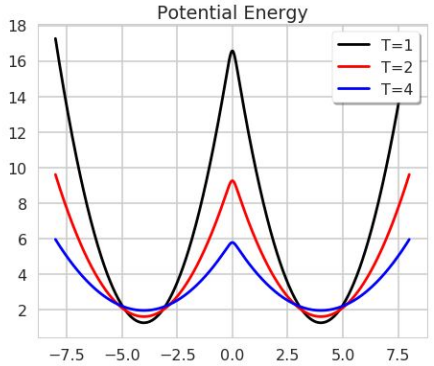


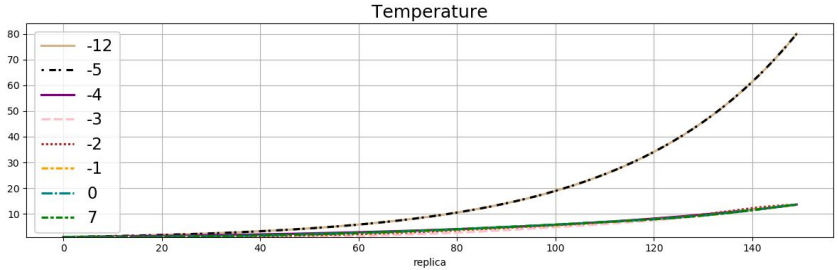
Syed 2020

Choosing the highest temperature

T_{max} is chosen to be the lowest temperature such that it, and all temperatures above it, are “mixing well”





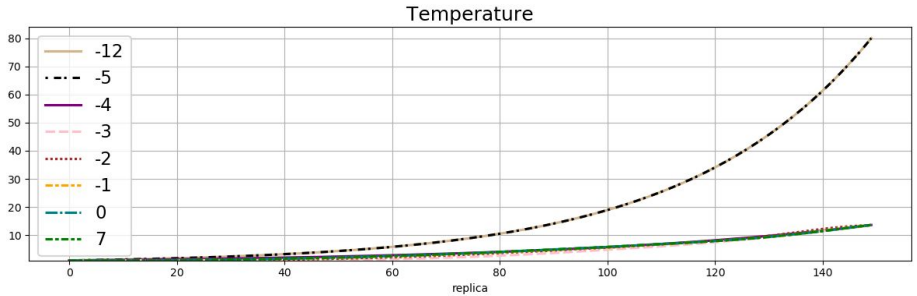


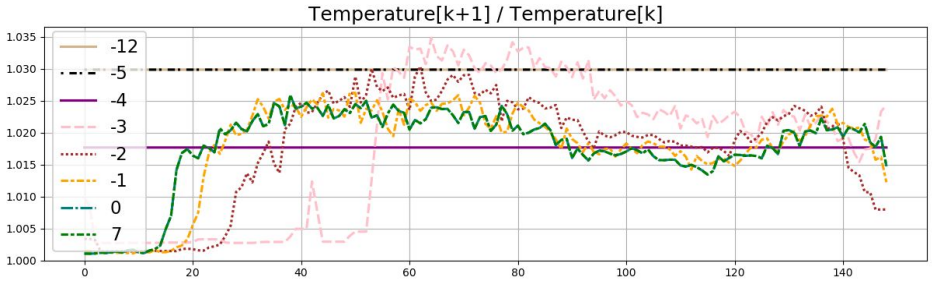
Communication barrier equilibration

After fixing $T_0 = 1$, and T_{max}, we adjust the gap between neighboring temperatures until

P[Swap replicas k → k+1]

are all equal





References

Neal. 2012: *MCMC Using Hamiltonian Dynamics.*

Beskos. 2010: *Optimal Tuning of the Hybrid Monte Carlo Algorithm.*

L. 2019: *A Condition Number for Hamiltonian Monte Carlo.*

Dikovsky. 2021: *Multi-instrument Bayesian Reconstruction of Plasma Shape Evolution...*

Syed 2020: *Non-reversible Parallel Tempering: A Scalable Highly Parallel MCMC Scheme*

L. 2021: *Hamiltonian Monte Carlo for Inverse Problems; Ill-Conditioning and Multi-Modality*