

Implementing Hamiltonian Monte Carlo for Efficient Bayesian Evolutionary Analysis

Arman Bilge

`abil933@aucklanduni.ac.nz`

Computational Evolution Group
The University of Auckland

21 October 2015

Motivation

- Bayesian statistics has **transformed evolutionary biology**
 - Divergence dating and clock rate estimation
 - Inference of demographics and population history parameters
 - Phylodynamic and epidemiological parameters

Motivation

- Bayesian statistics has **transformed evolutionary biology**
 - Divergence dating and clock rate estimation
 - Inference of demographics and population history parameters
 - Phylodynamic and epidemiological parameters
- Want to **sample phylogenetic tree and model parameters** according to their probability given the data $P(\mathcal{T}, \theta | D)$

Motivation

- Bayesian statistics has **transformed evolutionary biology**
 - Divergence dating and clock rate estimation
 - Inference of demographics and population history parameters
 - Phylodynamic and epidemiological parameters
- Want to **sample phylogenetic tree and model parameters** according to their probability given the data $P(\mathcal{T}, \theta | D)$
- Need a way to **efficiently explore** different models

Current State-of-the-Art

- Metropolis–Hastings Markov chain Monte Carlo (**MCMC**)
- Imagine a robot taking a **random walk**

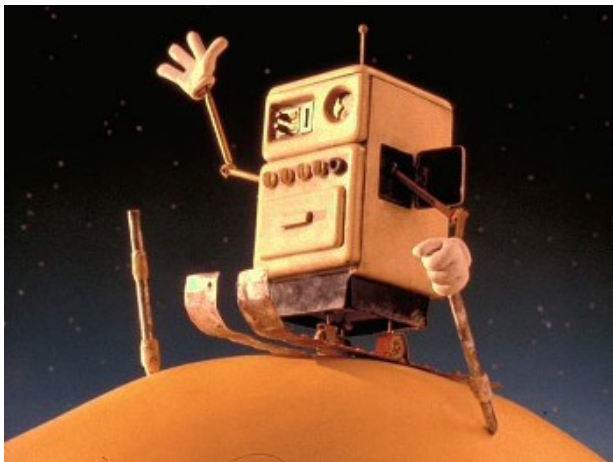
Current State-of-the-Art

- Metropolis–Hastings Markov chain Monte Carlo (**MCMC**)
- Imagine a robot taking a **random walk**
- That's just unbearably slow

Current State-of-the-Art

- Metropolis–Hastings Markov chain Monte Carlo (**MCMC**)
- Imagine a robot taking a **random walk**
- That's just unbearably slow
- **But suppose we gave our robot a pair of skis...**

Cooker, an Aspiring Skier



Physics Primer (Hamiltonian Dynamics)

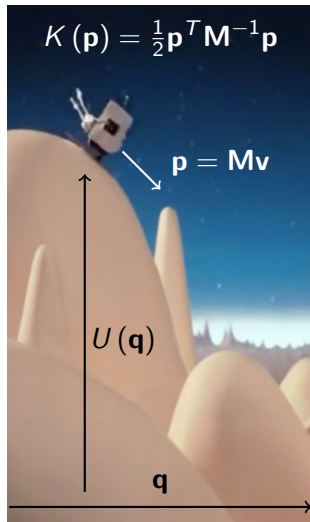
Definition

Let

- \mathbf{q} be the robot's position
- \mathbf{v} be its velocity
- \mathbf{M} be its mass
- $U(\mathbf{q})$ be its potential energy
- $K(\mathbf{p})$ be its kinetic energy

Then the Hamiltonian \mathcal{H} is

$$\mathcal{H}(\mathbf{q}, \mathbf{p}) = U(\mathbf{q}) + K(\mathbf{p})$$



Physics Primer (Hamiltonian Dynamics)

Definition

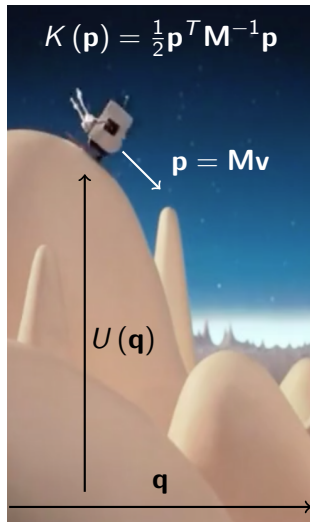
Let

- \mathbf{q} be the robot's position
- \mathbf{v} be its velocity
- \mathbf{M} be its mass
- $U(\mathbf{q})$ be its potential energy
- $K(\mathbf{p})$ be its kinetic energy

Then the Hamiltonian \mathcal{H} is

$$\mathcal{H}(\mathbf{q}, \mathbf{p}) = U(\mathbf{q}) + K(\mathbf{p})$$

- \mathcal{H} is conserved
- Simulates robot's motion



Hamiltonian Moves

Flip direction

Hamiltonian Moves

Ski!

Hamiltonian Moves

Push in a random direction

Hamiltonian Monte Carlo

How do we make a stats problem into a physics problem?

Hamiltonian Monte Carlo

How do we make a stats problem into a physics problem?

- Every location \mathbf{q} maps to some model parameters

Hamiltonian Monte Carlo

How do we make a stats problem into a physics problem?

- Every location \mathbf{q} maps to some model parameters
- Elevation at \mathbf{q} is $-\log P(\mathcal{T}, \theta | D)$

Hamiltonian Monte Carlo

How do we make a stats problem into a physics problem?

- Every location \mathbf{q} maps to some model parameters
- Elevation at \mathbf{q} is $-\log P(\mathcal{T}, \theta | D)$
- Locations with **high elevation** have **low probability**
- Locations with **low elevation** have **high probability**

Hamiltonian Monte Carlo

How do we make a stats problem into a physics problem?

- Every location \mathbf{q} maps to some model parameters
- Elevation at \mathbf{q} is $-\log P(\mathcal{T}, \theta | D)$
- Locations with **high elevation** have **low probability**
- Locations with **low elevation** have **high probability**
- Done!

Hamiltonian Monte Carlo

**Running a physics simulator seems like a lot of work!
Why should we bother?**

Hamiltonian Monte Carlo

Running a physics simulator seems like a lot of work!
Why should we bother?

Theorem (Creutz 1988)

Consider a model with n variables.

Then (under simplifying assumptions) the computation time is

- $\mathcal{O}(n^2)$ for MCMC; and
- $\mathcal{O}(n^{\frac{5}{4}})$ for HMC.

Hamiltonian Monte Carlo

Running a physics simulator seems like a lot of work!
Why should we bother?

Theorem (Creutz 1988)

Consider a model with n variables.

Then (under simplifying assumptions) the computation time is

- $\mathcal{O}(n^2)$ for MCMC; and
- $\mathcal{O}(n^{\frac{5}{4}})$ for HMC.

In practice this means **doubling the model complexity** increases computation time by

- **4x** for MCMC
- **<2.5x** for HMC!

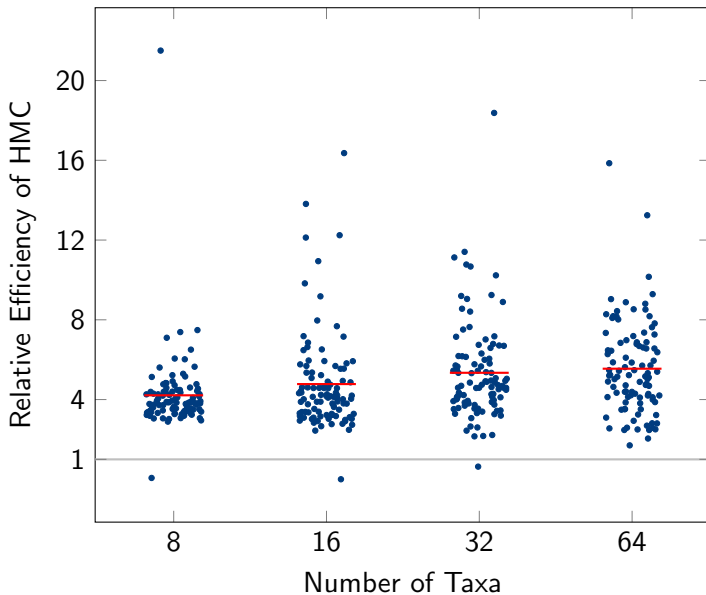
The Phylogenetic Ski Slope

Performance of HMC vs. MCMC

For {8, 16, 32, 64} taxa:

- ① Simulated 100 datasets under Yule and HKY models
- ② Estimated node heights with optimally-tuned HMC/MCMC
- ③ Measured efficiency as ESS of tree length per unit time (Effective Sample Size is # of independent samples)
- ④ Compared efficiency of HMC versus MCMC

Performance of HMC vs. MCMC



Concluding Remarks

Conclusions

- HMC **consistently out-performed** MCMC
- On average, HMC was **5x more efficient** than MCMC
- Open source implementation at
<http://github.com/armanbilge/B3/tree/hamilton>

Concluding Remarks

Conclusions

- HMC **consistently out-performed** MCMC
- On average, HMC was **5x more efficient** than MCMC
- Open source implementation at
<http://github.com/armanbilge/B3/tree/hamilton>

Future Work

- Inferring parameters for other evolutionary models
- Moving between tree topologies
- Automatic tuning of HMC for optimal performance
- Implementing and testing more sophisticated flavors of HMC

Acknowledgements

Thanks to

- Tim Vaughan and Alexei Drummond
- Members of the Computational Evolution Group
- Allan Wilson Centre Summer Scholarship
- New Zealand eScience Infrastructure

References

- M Creutz. *Physical Review D* 38.4 (1988). doi:10.1103/PhysRevD.38.1228
- RM Neal. *Handbook of Markov Chain Monte Carlo* (2011).