Quantum Parallel Markov Chain Monte Carlo(s)

Andrew Holbrook

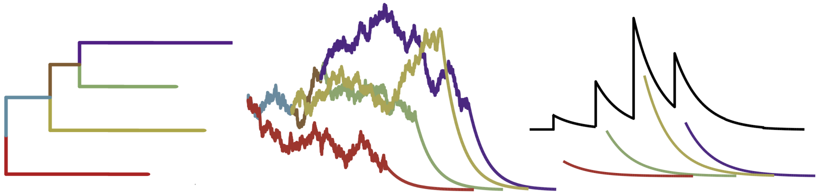
UCLA Biostatistics

April 19, 2023

My Personal Path to QPMCMC

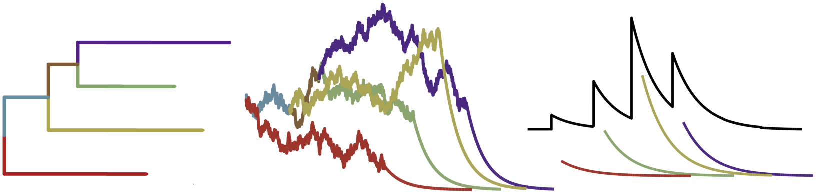
A Unified Model for Viral Spread

- Holbrook, Ji and Suchard (2022). *From viral evolution to spatial contagion: a biologically modulated Hawkes model*, Bioinformatics.



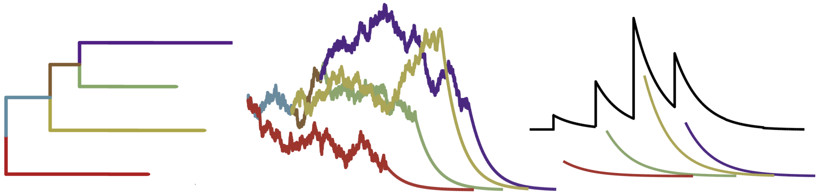
A Unified Model for Viral Spread

- Holbrook, Ji and Suchard (2022). *From viral evolution to spatial contagion: a biologically modulated Hawkes model*, Bioinformatics.
- Virus-specific latent variables connect a spatiotemporal Hawkes process model with a phylogenetic diffusion prior.



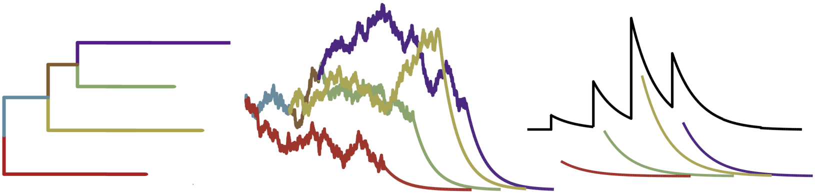
A Unified Model for Viral Spread

- Holbrook, Ji and Suchard (2022). *From viral evolution to spatial contagion: a biologically modulated Hawkes model*, Bioinformatics.
- Virus-specific latent variables connect a spatiotemporal Hawkes process model with a phylogenetic diffusion prior.
- The number of latent variables is $\mathcal{O}(N)$, for N the number of observed viruses.



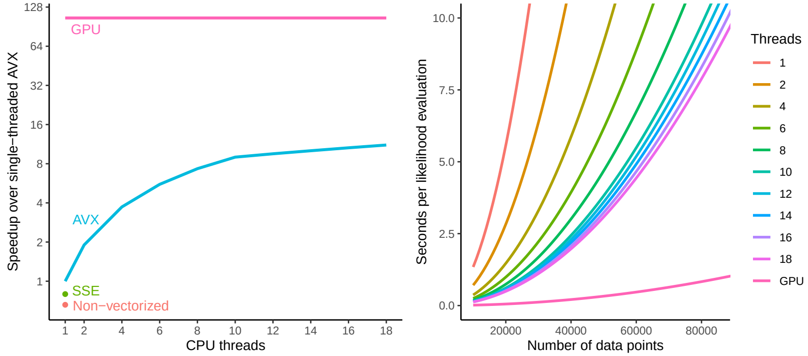
A Unified Model for Viral Spread

- Holbrook, Ji and Suchard (2022). *From viral evolution to spatial contagion: a biologically modulated Hawkes model*, Bioinformatics.
- Virus-specific latent variables connect a spatiotemporal Hawkes process model with a phylogenetic diffusion prior.
- The number of latent variables is $\mathcal{O}(N)$, for N the number of observed viruses.
- Hawkes likelihood computations require $\mathcal{O}(N^2)$ floating-point operations.



A Unified Model of Viral Spread

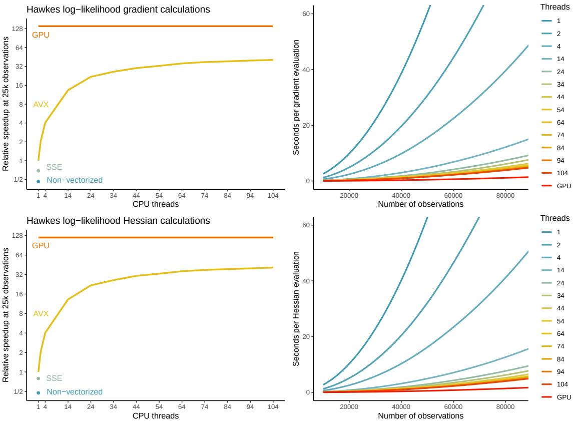
Proposal #1: use parallel computing to accelerate MH bottleneck, i.e., likelihood computations.



What about high dimensionality?

A Unified Model of Viral Spread

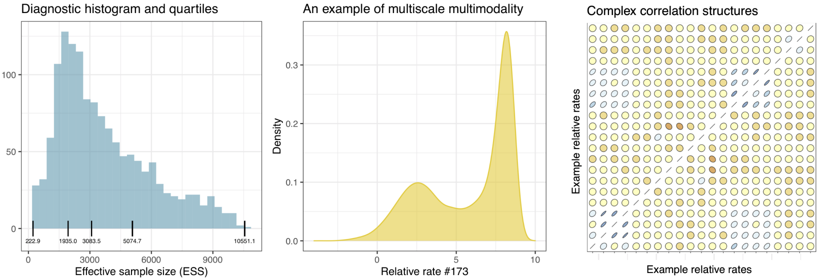
Proposal #2: also use parallel computing to accelerate adaptive HMC bottlenecks, i.e., log-likelihood gradient/Hessian.



What about bad geometry (non-linearity, multimodality)?

A Unified Model of Viral Spread

Proposal #3: to analyze over 23k Ebola cases (2014-2016 West Africa), run the chain for 30 days using Nvidia GV100 GPU.



Some Questions

- Can we parallelize the general structure of MCMC to overcome inferential challenges?

Some Questions

- Can we parallelize the general structure of MCMC to overcome inferential challenges?
- What other computational tools might accelerate Bayesian inference?

Some Questions

- Can we parallelize the general structure of MCMC to overcome inferential challenges?
- What other computational tools might accelerate Bayesian inference?
- Quantum computing achieves remarkable speedups for a limited set of problems, but can it help Bayesians?

Some Questions

- Can we parallelize the general structure of MCMC to overcome inferential challenges?
- What other computational tools might accelerate Bayesian inference?
- Quantum computing achieves remarkable speedups for a limited set of problems, but can it help Bayesians?
- What can quantum computing do for biomedicine?

Efficient Multiproposal Structures

Multiproposal MCMC

A multiproposal MCMC algorithm builds a transition kernel $P(\boldsymbol{\theta}_0, \mathrm{d}\boldsymbol{\theta})$ by:

1. generating P proposals $\boldsymbol{\Theta}_{-0} = (\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_P)$ from a joint distribution $Q(\boldsymbol{\theta}_0, \mathrm{d}\boldsymbol{\Theta}_{-0}) =: q(\boldsymbol{\theta}_0, \boldsymbol{\Theta}_{-0})\mathrm{d}\boldsymbol{\Theta}_{-0}$; and

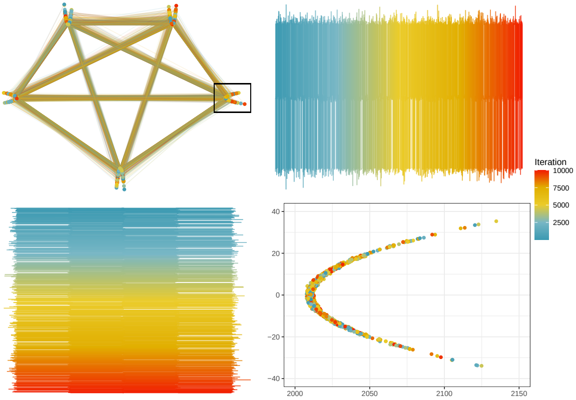
2. selecting the next state with probabilities

$$\pi_p = \frac{\pi(\boldsymbol{\theta}_p) q(\boldsymbol{\theta}_p, \boldsymbol{\Theta}_{-p})}{\sum_{p'=0}^{P} \pi(\boldsymbol{\theta}_{p'}) q(\boldsymbol{\theta}_{p'}, \boldsymbol{\Theta}_{-p'})}, \quad p \in \{0, 1, \ldots, P\}.$$

This kernel maintains detailed balance and leaves $\pi(\mathrm{d}\boldsymbol{\theta})$ invariant.

Multiproposal MCMC

PRO: using large numbers of proposals P helps overcome multimodality and non-linearity.



CON: requires $\mathcal{O}(P)$ target evaluations $\pi(\boldsymbol{\theta}_p)$ and proposal evaluations $q(\boldsymbol{\theta}_p, \boldsymbol{\Theta}_{-p})$, each of the latter being $\mathcal{O}(P)$.

Simplified Acceptance Probabilities

Can we somehow enforce $q(\boldsymbol{\theta}_p, \boldsymbol{\Theta}_{-p}) = q(\boldsymbol{\theta}_{p'}, \boldsymbol{\Theta}_{-p'})$, $\forall p, p' \in \{0, 1, \ldots, P\}$, to obtain simplified acceptance probabilities

$$\pi_p = \frac{\pi(\boldsymbol{\theta}_p)}{\sum_{p'=0}^{P} \pi(\boldsymbol{\theta}_{p'})}, \quad p \in \{0, 1, \ldots, P\}.$$

Such structured multiproposals would result in $\mathcal{O}(P^2)$ time savings and simpler implementation. I consider two such approaches in

- Holbrook (2023a). *Generating MCMC proposals by randomly rotating the regular simplex*, Journal of Multivariate Analysis.

Tjelmeland Correction (a free lunch)

Tjelmeland (2004) suggests the two-step multiproposal

1. $\bar{\boldsymbol{\theta}} \sim N_D(\boldsymbol{\theta}^{(s)}, \boldsymbol{\Sigma})$;
2. $\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_P \overset{iid}{\sim} N_D(\bar{\boldsymbol{\theta}}, \boldsymbol{\Sigma})$.

Why? No satisfactory explanation. But it turns out that this structure leads to the desired equality (Holbrook 2023a):

$$q(\boldsymbol{\theta}_p, \boldsymbol{\Theta}_{-p}) = q(\boldsymbol{\theta}_{p'}, \boldsymbol{\Theta}_{-p'}), \forall p, p' \in \{0, 1, \dots, P\} .$$

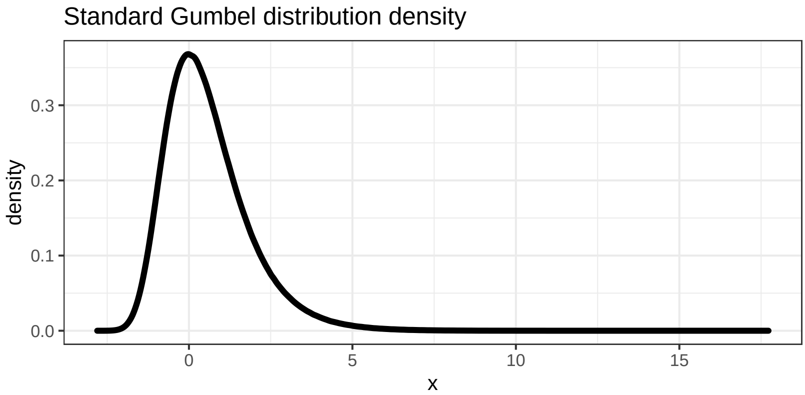
As promised, the resulting acceptance probabilities are:

$$\pi_p = \frac{\pi(\boldsymbol{\theta}_p)}{\sum_{p'=0}^{P} \pi(\boldsymbol{\theta}_{p'})} , \quad p \in \{0, 1, \dots, P\} .$$

Only the $\mathcal{O}(P)$ target evaluations remain in our way.

A Quantum Parallel MCMC

The Gumbel Distribution



If $z \sim \mathit{Gumbel}(0, 1)$, then it has density and distribution functions

$$g(z) = \exp\big(-z - \exp(-z)\big) \quad \text{and} \quad G(z) = \exp\big(-\exp(-z)\big)\,.$$

Gumbel-Max Trick

We wish to sample from the discrete distribution $\hat{p} \sim \mathit{Discrete}(\boldsymbol{\pi})$ for $\hat{p} \in \{0, 1, \ldots, P\}$ and we only know $\boldsymbol{\pi}^* = c\boldsymbol{\pi}$ for some $c > 0$.

Define $\boldsymbol{\lambda}^* = \log \boldsymbol{\pi}^* = \log \boldsymbol{\pi} + \log c$ and suppose $z_0, z_1, \ldots, z_P \overset{iid}{\sim} \mathit{Gumbel}(0, 1)$.

Finally, define $\alpha_p^* := \lambda_p^* + z_p$ and $\hat{p} = \arg\max_{p=0,\ldots,P} \alpha_p^*$.

Then the following holds (Papandreou and Yuille, 2011):

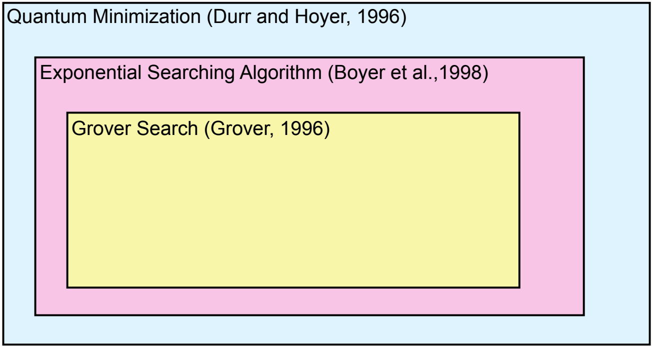
$$\Pr(\hat{p} = p) = \pi_p \,, \quad p = 0, 1, \ldots, P.$$

Data: Initial Markov chain state $\boldsymbol{\theta}^{(0)}$; total length of Markov chain S; total number of proposals per iteration P.
Result: A Markov chain $\boldsymbol{\theta}^{(1)}, \dots, \boldsymbol{\theta}^{(S)}$.
for $s \in \{1, \dots, S\}$ **do**
    $\boldsymbol{\theta}_0 \leftarrow \boldsymbol{\theta}^{(s-1)}$;
    $\bar{\boldsymbol{\theta}} \leftarrow \mathit{Normal}_D(\boldsymbol{\theta}_0, \boldsymbol{\Sigma})$;
    $z_0 \leftarrow \mathit{Gumbel}(0, 1)$;
    for $p \in \{1, \dots, P\}$ **do**
        $\boldsymbol{\theta}_p \leftarrow \mathit{Normal}_D(\bar{\boldsymbol{\theta}}, \boldsymbol{\Sigma})$;
        $z_p \leftarrow \mathit{Gumbel}(0, 1)$;
    end
    $\hat{p} \leftarrow \arg\min_{p=0,\dots,P} \Big(f(p) := -\big(z_p + \log \pi(\boldsymbol{\theta}_p)\big)\Big)$;
    $\boldsymbol{\theta}^{(s)} \leftarrow \boldsymbol{\theta}_{\hat{p}}$;
end
return $\boldsymbol{\theta}^{(1)}, \dots, \boldsymbol{\theta}^{(S)}$.

Quantum Parallel MCMC

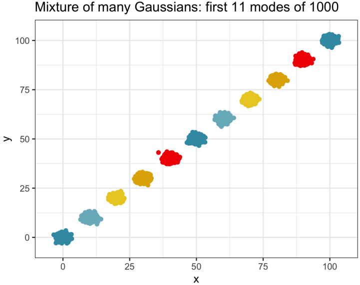
Use a quantum circuit with $O(\sqrt{P})$ depth to obtain

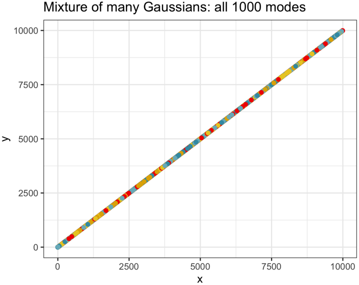
$$\hat{p} = \arg\min_{p=0,\dots,P} \left(f(p) := -\big(z_p + \log \pi(\boldsymbol{\theta}_p)\big) \right) .$$



- Holbrook (2023b). *A quantum parallel Markov chain Monte Carlo*, JCGS.

QPMCMC: Racing to an ESS of 100



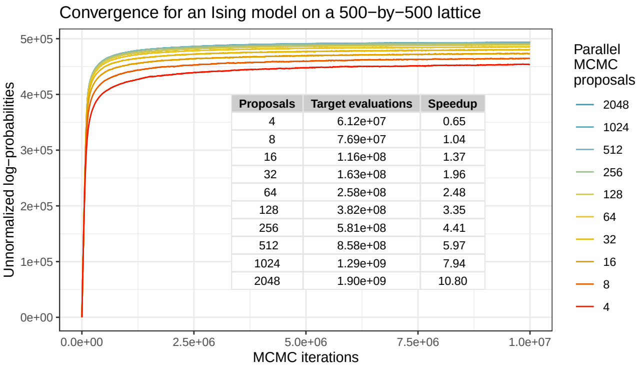


Proposals	MCMC iterations	Target evaluations	Speedup	Efficiency gain
1,000	249,398 (200,998, 311,998)	24,988,963 (20,149,132, 31,265,011)	9.98 (9.98, 9.98)	1
5,000	14,398 (12,998, 16,998)	3,314,560 (2,989,418, 3,916,281)	21.72 (21.70, 21.74)	7.58 (6.25, 9.71)
10,000	5,998 (4,998, 6,998)	1,993,484 (1,662,592, 2,330,842)	30 (29.96, 30.26)	12.87 (8.64, 18.80)

Ising Model Target

Consider the Ising-type lattice model over configurations $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_D)$ consisting of D individual spins $\theta_d \in \{-1, 1\}$

$$\pi(\boldsymbol{\theta}|\rho) \propto \exp\left(\rho \sum_{(d,d')\in\mathcal{E}} \theta_d \theta_{d'}\right) .$$



Proposals	Target evaluations	Speedup
4	6.12e+07	0.65
8	7.69e+07	1.04
16	1.16e+08	1.37
32	1.63e+08	1.96
64	2.58e+08	2.48
128	3.82e+08	3.35
256	5.81e+08	4.41
512	8.58e+08	5.97
1024	1.29e+09	7.94
2048	1.90e+09	10.80

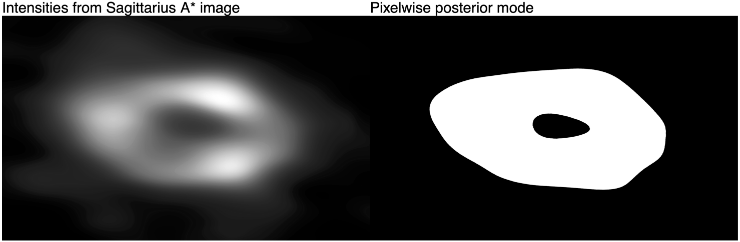
Bayesian Image Segmentation

Following Hurn (1997), y_d are intensity values associated with individual pixels.

$$
\begin{aligned}
y_d | (\mu_\ell, \sigma^2, \theta_d) &\overset{ind}{\sim} \text{Normal}(\mu_\ell, \sigma^2)\,, \quad y_d \in [0, 255]\,,\\
\theta_d &= \ell\,, \quad d \in \{1, \ldots, D\}\,,\\
\mu_\ell &\overset{iid}{\sim} \text{Uniform}(0, 255)\,, \quad \ell \in \{-1, 1\}\,,\\
\frac{1}{\sigma^2} &\sim \text{Gamma}\left(\frac{1}{2}, \frac{1}{2}\right)\\
\boldsymbol{\theta} &\sim \text{Ising}(\rho)\,, \quad \rho = 1.2\,.
\end{aligned}
$$

Bayesian Image Segmentation

Segmenting a 4,076-by-4,076 intensity map. Using 1,024 proposals, QPMCMC requires less than 10% the evaluations required by a conventional computer.



Four Problems

We achieve a quadratic speedup over conventional computers, but:

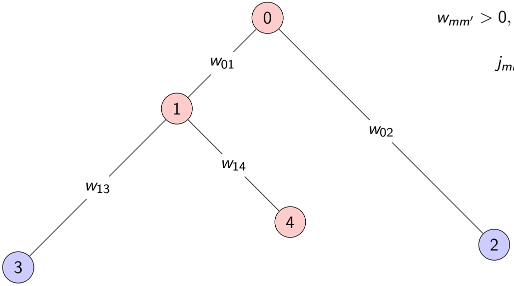
1. no explicit circuit (depth of Grover's oracle call unknown);
2. algorithm not exact (quantum minimization may not reach mininum);
3. nobody cares about quadratic speedups (GPUs can do that);
4. what's this got to do with biomedicine?

QPMCMC2

Faster QPMCMC

- Collaboration between NTU, Foxconn, UCLA and KU Leuven.
- Lin C, Chen K, Lemey P, Suchard M, Holbrook A, Hsieh M (2023). *Quantum speedups for multiproposal MCMC*.
- QPMCMC2 achieves eponential speedups for a large class of discrete graphical models: $O(P)$ to $O(1)$ operations with only $O(\log P)$ qubits
- QPMCMC2 fully explicit and exact

Ancestral Trait Reconstruction



$$w_{mm'} > 0,\ m \neq m' \in \{0, \ldots 2M_o - 2\}$$

$$j_{mm'} = f_\gamma\left(\frac{1}{w_{mm'}}\right) \geq 0$$

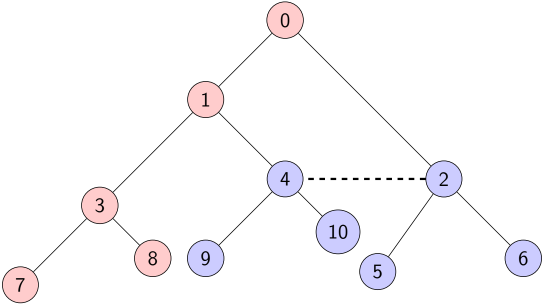
$+$: $\sigma_m = 1$

$-$: $\sigma_m = -1$

- We can always infer ancestral traits with the help of a phylogenetic Ising model:

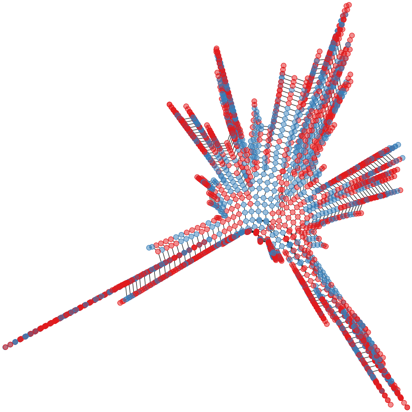
$$Pr(\boldsymbol{\sigma}_a | \boldsymbol{\sigma}_o, \beta, \gamma, \mathcal{G}) \propto \exp\left(\beta \sum_{m,m'} j_{m,m'} \sigma_m \sigma_{m'}\right).$$

Bacterial Reticulate Evolution



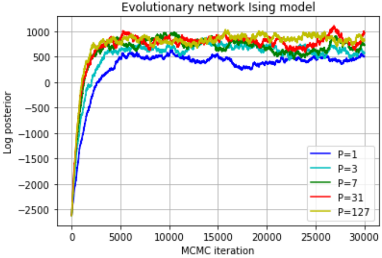
- Reticulate evolution ruins all Marc's fun. No more linear-time likelihoods/gradients via dynamic programming.

Ampicillin Resistance for 248 Salmonella Isolates

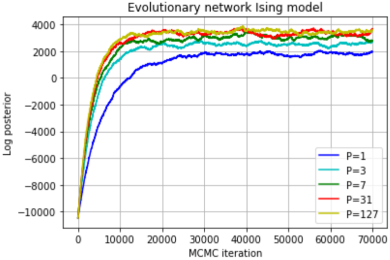


- We define a phylogenetic Ising model on a Neighbor-Net graph with 3,313 vertices and 5,945 edges.
- Figure shows vertex-wise posterior modes from 150k QPMCMC2 iterations with $P = 127$.

Convergence as a Function of MCMC Iterations

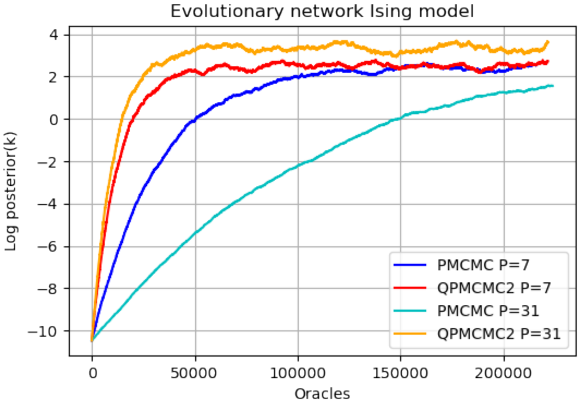


(a) Single-trait model.

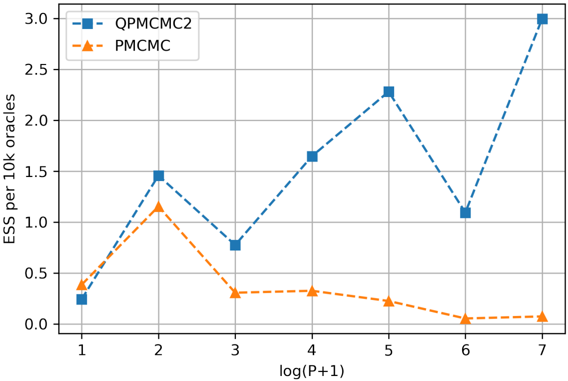


(b) Four-trait model.

Convergence as a Function of Oracle Calls



No Performance Penalty for Proposals



Intuition for QPMCMC2

- Qubits (quantum bits) $|\theta\rangle$ are complex vectors with magnitude 1, i.e., $\langle\theta|\theta\rangle = 1$.

- If orthogonal $|\theta\rangle_0, \dots, |\theta_P\rangle$ have magnitude 1, then so does

$$\frac{1}{\sqrt{P+1}} \sum_{p=0}^{P} |\theta_p\rangle .$$

  This is called a uniform superposition of the individual vectors. All elements share the same probability amplitude $1/\sqrt{P+1}$.

- If $\sum_{p=0}^{P} \pi_p = 1$, then

$$\sum_{p=0}^{P} \sqrt{\pi_p} \, |\theta_p\rangle$$

  also has magnitude 1.

Intuition for QPMCMC2

- We cannot access all elements of a superposition, but we can randomly sample one using quantum measurement.
- Measurement collapses the quantum state and returns one element of the superposition with probability given by its squared amplitude.
- Big Idea: cheaply manipulate quantum system to obtain

$$\sum_{p=0}^{P} \sqrt{\pi(\theta_p)}\,|\theta_p\rangle$$

and perform measurement to get next MCMC sample θ_p with probability $\pi(\theta_p)$.

Implementing QPMCMC2

We begin by initializing 6 quantum registers to 0

$$|0\rangle_{\mathcal{P}}\,|0\rangle_{\mathcal{H}_0}\,|0\rangle_{\mathcal{H}_1}\,|0\rangle_{\mathcal{H}_2}\,|0\rangle_{\Pi}\,|0\rangle_{S}$$

and load the current state $\boldsymbol{\theta}^{(s-1)} = \boldsymbol{\theta}_0$ onto the second register at cost $O(\log|\Theta|)$:

$$|0\rangle_{\mathcal{P}}\,|\boldsymbol{\theta}_0\rangle_{\mathcal{H}_0}\,|0\rangle_{\mathcal{H}_1}\,|0\rangle_{\mathcal{H}_2}\,|0\rangle_{\Pi}\,|0\rangle_{S}\ .$$

Implementing QPMCMC2

Next, we apply an operator $O_{\bar{q}}$ that performs the Tjelmeland correction by sampling from $\bar{q}(\boldsymbol{\theta}_0, \cdot)$ and placing result in the third register:

$$\begin{aligned} & |0\rangle_{\mathcal{P}} |\boldsymbol{\theta}_0\rangle_{\mathcal{H}_0} |0\rangle_{\mathcal{H}_1} |0\rangle_{\mathcal{H}_2} |0\rangle_{\Pi} |0\rangle_{S} \\ \longmapsto \quad & |0\rangle_{\mathcal{P}} |\boldsymbol{\theta}_0\rangle_{\mathcal{H}_0} \left|\bar{\boldsymbol{\theta}}\right\rangle_{\mathcal{H}_1} |0\rangle_{\mathcal{H}_2} |0\rangle_{\Pi} |0\rangle_{S} \ . \end{aligned}$$

Implementing QPMCMC2

We then create a superposition in the first register by placing its $\log P$ qubits in superposition and, again, apply the operator $O_{\bar{q}}$ that samples from $\bar{q}(\bar{\boldsymbol{\theta}}, \cdot)$ and puts the result in the fourth register:

$$
\begin{aligned}
& |0\rangle_{\mathcal{P}} |\boldsymbol{\theta}_0\rangle_{\mathcal{H}_0} \left|\bar{\boldsymbol{\theta}}\right\rangle_{\mathcal{H}_1} |0\rangle_{\mathcal{H}_2} |0\rangle_{\Pi} |0\rangle_{S} \\
\longmapsto \quad & \frac{1}{\sqrt{P+1}} \sum_{p=0}^{P} |p\rangle_{\mathcal{P}} |\boldsymbol{\theta}_0\rangle_{\mathcal{H}_0} \left|\bar{\boldsymbol{\theta}}\right\rangle_{\mathcal{H}_1} |0\rangle_{\mathcal{H}_2} |0\rangle_{\Pi} |0\rangle_{S} \\
\longmapsto \quad & \frac{1}{\sqrt{P+1}} \sum_{p=0}^{P} |p\rangle_{\mathcal{P}} |\boldsymbol{\theta}_0\rangle_{\mathcal{H}_0} \left|\bar{\boldsymbol{\theta}}\right\rangle_{\mathcal{H}_1} |\boldsymbol{\theta}_p\rangle_{\mathcal{H}_2} |0\rangle_{\Pi} |0\rangle_{S} \ .
\end{aligned}
$$

Implementing QPMCMC2

Next, we apply the oracle gate O_π that takes input from the 4th register and outputs to the 5th:

$$\frac{1}{\sqrt{P+1}}\sum_{p=0}^{P}|p\rangle_{\mathcal{P}}\,|\boldsymbol{\theta}_0\rangle_{\mathcal{H}_0}\,\big|\bar{\boldsymbol{\theta}}\big\rangle_{\mathcal{H}_1}\,|\boldsymbol{\theta}_p\rangle_{\mathcal{H}_2}\,|0\rangle_{\Pi}\,|0\rangle_S$$

$$\longmapsto \quad \frac{1}{\sqrt{P+1}}\sum_{p=0}^{P}|p\rangle_{\mathcal{P}}\,|\boldsymbol{\theta}_0\rangle_{\mathcal{H}_0}\,\big|\bar{\boldsymbol{\theta}}\big\rangle_{\mathcal{H}_1}\,|\boldsymbol{\theta}_p\rangle_{\mathcal{H}_2}\,|\pi^*(\boldsymbol{\theta}_p)\rangle_{\Pi}\,|0\rangle_S\ .$$

We then apply a controlled rotation to the final register, getting

$$\frac{1}{\sqrt{P+1}}\sum_{p=0}^{P}|p\rangle_{\mathcal{P}}\,|\boldsymbol{\theta}_0\rangle_{\mathcal{H}_0}\,\big|\bar{\boldsymbol{\theta}}\big\rangle_{\mathcal{H}_1}\,|\boldsymbol{\theta}_p\rangle_{\mathcal{H}_2}\,|\pi^*(\boldsymbol{\theta}_p)\rangle_{\Pi}$$
$$\left(\sqrt{1-\pi^*(\boldsymbol{\theta}_p)}\,|0\rangle_S+\sqrt{\pi^*(\boldsymbol{\theta}_p)}\,|1\rangle_S\right).$$

Implementing QPMCMC2

For the penultimate step, we perform quantum measurement on the final register

$$\frac{1}{\sqrt{P+1}}\sum_{p=0}^{P}|p\rangle_{\mathcal{P}}|\boldsymbol{\theta}_0\rangle_{\mathcal{H}_0}\left|\bar{\boldsymbol{\theta}}\right\rangle_{\mathcal{H}_1}|\boldsymbol{\theta}_p\rangle_{\mathcal{H}_2}|\pi^*(\boldsymbol{\theta}_p)\rangle_{\Pi}$$

$$\left(\sqrt{1-\pi^*(\boldsymbol{\theta}_p)}\,|0\rangle_S+\sqrt{\pi^*(\boldsymbol{\theta}_p)}\,|1\rangle_S\right).$$

If this register's qubit collapses to 1, our overall state is

$$\sum_{p'=0}^{P}\sqrt{\frac{\pi(\boldsymbol{\theta}_p)}{\sum_{p'=0}^{P}\pi(\boldsymbol{\theta}_{p'})}}\,|p\rangle_{\mathcal{P}}|\boldsymbol{\theta}_0\rangle_{\mathcal{H}_0}\left|\bar{\boldsymbol{\theta}}\right\rangle_{\mathcal{H}_1}|\boldsymbol{\theta}_p\rangle_{\mathcal{H}_2}|\pi^*(\boldsymbol{\theta}_p)\rangle_{\Pi}|1\rangle_S\ ,$$

and measurement of the 4th register effectively samples from the multiproposal kernel.

Implementing QPMCMC2

Theorem

The quantum multiproposal MCMC algorithm described above that satisfies $\pi^*(\theta_p) < 1$ for all $p \in \{0, 1, \dots, P\}$ has a running time:

$$\frac{2T(O_{\bar{q}}) + T(O_{\pi^*}) + \mathcal{O}(1)}{\min_{p \in \{0,\cdots,P\}} \pi^*(\theta_p)}.$$

Here, $O_{\bar{q}}$ and O_{π} represent the quantum operations characterized by $\bar{q}(\boldsymbol{\theta}, \boldsymbol{\theta}')$ and $\pi^*(\cdot)$ respectively, and their circuit depths $T(O_{\bar{q}})$ and $T(O_{\pi^*})$ do not depend on number of proposals P.

- Note: for many examples, such as the Ising model with bit-flip Tjelmeland-corrected proposals, the denominator does not decrease with larger P.

Future Quantum MCMC Research

- within MH, locally-balanced proposals (Zanella, 2019) choose among points in a neighborhood of the current position with probabilities, e.g., $\sqrt{\pi^*(\theta)}$;
- nonreversible MH (Turitsyn et al., 2008; Vucelja, 2014) preserves momentum between proposals by, e.g., only considering flipping $\pm$ to $\mp$. U-turns occur with probability

$$p(\boldsymbol{\theta}_\pm, \boldsymbol{\theta}_\mp) \Big/ \left(1 - \sum_{z_\pm \neq \boldsymbol{\theta}_\pm} p(\boldsymbol{\theta}_\pm, z_\pm)\right)$$

for $p(\cdot,\cdot)$ a globally defined transition probability matrix. Multiple choices for $p(\pm,\mp)$, but one is

$$p(\boldsymbol{\theta}_+, \boldsymbol{\theta}_-) = \max\left(0\,, \sum_z p(\boldsymbol{\theta}_-, z_-) - p(\boldsymbol{\theta}_+, z_+)\right)\,.$$

Future Quantum MCMC Research

- We are also thinking about quantum extensions to HMC. For discrete models, quantum enhanced MH (Layden, 2023) simulates quantum dynamics starting at binary state $|\theta_0\rangle$

$$|\theta_t\rangle = U\,|\theta_0\rangle = e^{-itH}\,|\theta_0\rangle$$

  and collapses superposition $|\theta_t\rangle$ to get proposal θ^*.

- We can extend this to continuous distributions following the quantum Hamiltonian descent algorithm (Leng, 2023), but Heisenberg's uncertainty principle causes issues.

Acknowledgments

This work is supported by:

- NSF grants DMS 2236854 and DMS 2152774;
- NIH grant K25 AI153816;
- and a gift from the Karen Toffler Charitable Trust.