# Exact Bayesian Inference for Markov Switching Diffusions

Timothée Stumpf Fétizon [tim.stumpf-fetizon@warwick.ac.uk]

June 14, 2021

Joint work with K. Łatuszyński, J. Palczewski, G. Roberts

# Outline

# Model

# Many time series exhibits discrete regime shifts.

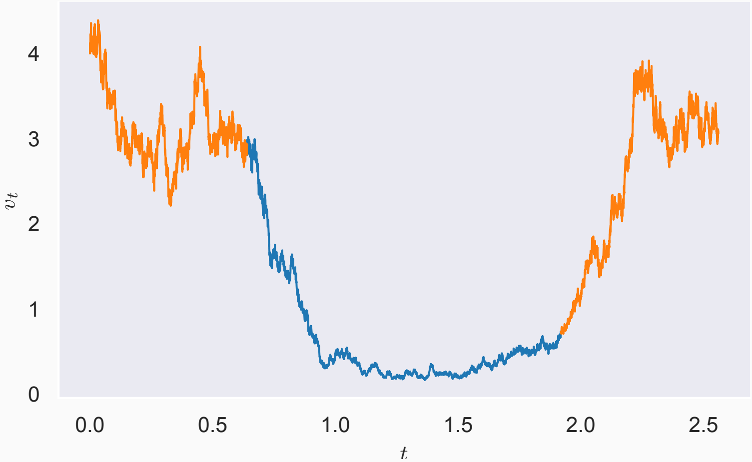



**Figure 1:** Pseudo-interest rate time series.

# We model the regime as a Markov jump process.

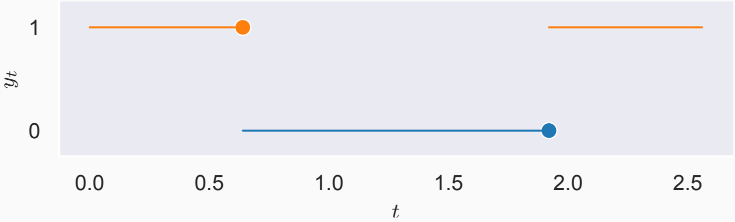



**Figure 2:** The corresponding trajectory of the regime.

$$Y(\tau_0) \xrightarrow{\mathrm{Exp}(\lambda_{Y(\tau_0)})} Y(\tau_1) \xrightarrow{\mathrm{Exp}(\lambda_{Y(\tau_1)})} Y(\tau_2) \xrightarrow{\mathrm{Exp}(\lambda_{Y(\tau_2)})} \ldots$$

**Figure 3:** A 2-state Markov jump process.

Since the exponential distribution is memoryless, $Y$ is Markovian. More generally, $\lambda_{ij}$ gives the transition rate from state $i$ to $j$.

# The diffusion process arises as a limit in discrete time.

Consider the process that evolves according to

$$\underbrace{V_{t+\epsilon} - V_t}_{\text{process increment}} = \underbrace{\mu(V_t, Y_t)}_{\text{instant drift}} \times \epsilon + \underbrace{\sigma(V_t, Y_t)}_{\text{instant volatility}} \times \underbrace{(W_{t+\epsilon} - W_t)}_{\text{Brownian increment}} \tag{1}$$

Under [conditions], there is a limiting process as $\epsilon \to 0$. We write

$$\mathrm{d}V_t = \mu(V_t, Y_t)\,\mathrm{d}t + \sigma(V_t, Y_t)\,\mathrm{d}W_t \tag{2}$$

We parameterize the *instantaneous drift* $\mu_\theta$ and *volatility* $\sigma_\theta$ in terms of a vector $\theta$.

**Intractable likelihood problem!**

$\pi(v_{t+\epsilon}|v_t, y_t, \theta)$ typically not available!

# We discretely observe the diffusion process.

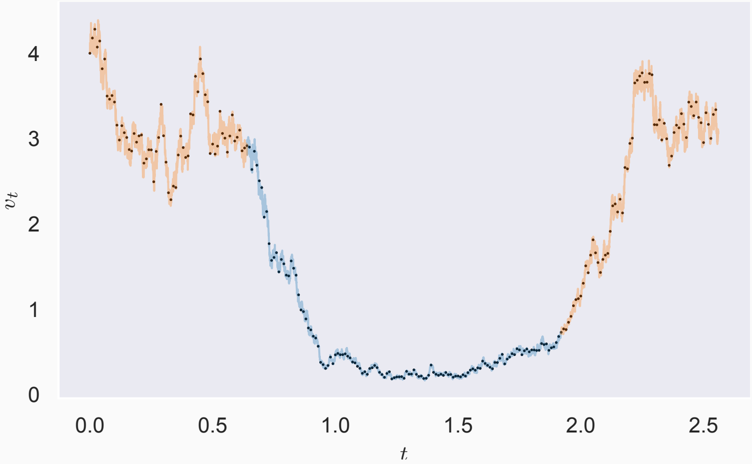



**Figure 4:** What we see: Discrete observations $v_s$ on the path of $V$. *Goal*: Sample from posterior $\pi(\theta, y, \lambda | v_s)$ for some prior $\pi(\theta, y, \lambda)$!

# Inference Strategy

# Desiderata.

We want to design an MCMC algorithm targeting the posterior on $(\theta, y, \lambda)$, s.t.:

- it targets the exact posterior, and the *Markov chain central limit theorem* applies - estimates are unbiased, standard error decays according to $\mathcal{O}(\text{computational budget}^{-1/2})$.
- it is model agnostic in principle - plug in $\mu_\theta$ and $\sigma_\theta$ and you're good to go.
- it is an "algorithm for the people" - no supercomputers required!

# Strategy.

1. transform $V$ to a process $X$ with a tractable *dominating measure*.
2. augment with the *missing data* - the bridges between observations $v_s$ and $Y$ - such that conditional updates are "easy"!
3. devise an infinite-dimensional *Gibbs sampler* with updates (parameters|missing) and (missing|parameters).
4. carry out the updates based on finite information, using *Barker's algorithm* in conjunction with *Bernoulli factories* and the *Exact algorithms*.

# Simplified setting and notation.

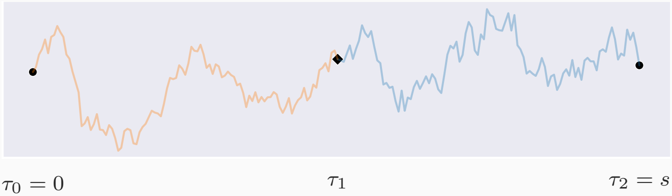



**Figure 5:** Event times; observations $\tau_0 = 0$ and $\tau_1 = s$ and jumps $\tau_1$

- for simplicity: assume that $\sigma_\theta$ is constant in $Y$ and we observe $V$ at times 0 and $s$.
- event times $\tau$ consist of observation times $\{0, s\}$ and intervening jump times. Denote consecutive jump times $\dot{\tau} \sim \ddot{\tau}$.
- ancillary quantities are denoted by $a$.
- random variables in upper case, realizations thereof in lower case.
- I skip inference for $\lambda$, which is conditionally conjugate.

# Event time augmentation.

Suppose we observe $V$ at times $\tau$. Then by the Markov property

$$\pi(v_{\tau\setminus 0}|v_0, y_{[0,s]}, \theta) = \prod_{(\dot{\tau}\sim\ddot{\tau})\in\tau} \underbrace{\pi(v_{\ddot{\tau}}|v_{\dot{\tau}}, y_{\dot{\tau}}, \theta)}_{\text{law of ordinary diffusion!}} \tag{3}$$

i.e. we can apply tools from ordinary diffusion inference to address the terms $\pi(v_{\ddot{\tau}}|v_{\dot{\tau}}, y_{\dot{\tau}}, \theta)$.

# Dominating measure.

Define the *Lamperti transform*

$$\eta_\theta(v_t) = \int^{v_t} \frac{\mathrm{d}a}{\sigma_\theta(a)} \tag{4}$$

and the *reduced process* $X_t = \eta_\theta(V_t)$ with induced measure $\mathbb{X}_{x_0,y,\theta}$ and SDE

$$\mathrm{d}X_t = \delta_\theta(X_t, Y_t)\,\mathrm{d}t + \mathrm{d}W_t \tag{5}$$

Then, by the *Girsanov theorem* and under [conditions],

$$\frac{\mathrm{d}\mathbb{X}_{x_{\dot{\tau}},y_{\dot{\tau}},\theta}}{\underbrace{\mathrm{d}\mathbb{W}_{x_{\dot{\tau}}}}_{\text{Wiener measure}}}(x_{(\dot{\tau},\ddot{\tau}]}) = a\exp\left[-\int_{\dot{\tau}}^{\ddot{\tau}} \underbrace{\varphi_\theta(x_t, y_{\dot{\tau}})}_{2^{-1}\left(\delta^2_\theta(x_t,y_t)+\partial_{x_t}\delta_\theta(x_t,y_t)\right)} \mathrm{d}t\right] \tag{6}$$

# Diffusion path augmentation.

Changing the dominating measure to $\mathrm{Leb} \times \mathbb{W}_{x_{\dot\tau}, x_{\ddot\tau}}$, obtain *augmented transition density*

$$\underbrace{\pi(x_{(\dot\tau,\ddot\tau]} | x_{\dot\tau}, y_{\dot\tau}, \theta)}_{\text{w.r.t. } \mathrm{Leb} \times \mathbb{W}_{x_{\dot\tau}, x_{\ddot\tau}}} = a \frac{\mathrm{d}\mathbb{X}_{x_{\dot\tau}, y_{\dot\tau}, \theta}}{\mathrm{d}\mathbb{W}_{x_{\dot\tau}}}(x_{(\dot\tau,\ddot\tau]}) \tag{7}$$

Switch to *non-centered parameterization* to ensure *irreducibility*:

$$\omega_\theta(x_t) = x_t - \eta_\theta(v_{\dot\tau}) - \frac{t - \dot\tau}{\ddot\tau - \dot\tau}(\eta_\theta(v_{\ddot\tau}) - \eta_\theta(v_{\dot\tau})), \qquad t \in [\dot\tau, \ddot\tau) \tag{8}$$

Such that $Z_{(x_{\dot\tau}, x_{\ddot\tau})} = \omega_\theta(X_{(x_{\dot\tau}, x_{\ddot\tau})})$ is a standard Brownian bridge under $\mathbb{W}_{x_{\dot\tau}, x_{\ddot\tau}} \circ \omega_\theta^{-1} = \mathbb{B}$. Now,

$$\underbrace{\pi(v_{\ddot\tau}, z_{(\dot\tau,\ddot\tau)} | v_{\dot\tau}, y_{\dot\tau}, \theta)}_{\text{w.r.t. } \mathrm{Leb} \times \mathbb{B}} = a \frac{\mathrm{d}\mathbb{X}_{x_{\dot\tau}, y_{\dot\tau}, \theta}}{\mathrm{d}\mathbb{W}_{x_{\dot\tau}}}(\omega_\theta^{-1}(z_{(\dot\tau,\ddot\tau)}), \eta_\theta(v_{\ddot\tau})) \tag{9}$$

# Infinite dimensional Gibbs sampler.

Put it all together:

$$\underbrace{\pi(\theta, \lambda, h, y|v_0)}_{\text{augmented posterior}} \propto \underbrace{\pi(v_s, h|v_0, y, \theta)}_{\text{aug trans density}} \underbrace{\pi(y|\lambda)}_{\text{regime prior}} \underbrace{\pi(\theta)\pi(\lambda)}_{\text{param prior}} \tag{10}$$

$$\underbrace{\pi(v_s, h|v_0, y, \theta)}_{\text{w.r.t. } (\mathrm{Leb} \times \mathbb{B})^{|\tau|-1}} = \prod_{\dot{\tau} \sim \ddot{\tau} \in \tau} \pi(v_{\ddot{\tau}}, z_{(\dot{\tau}, \ddot{\tau})}|v_{\dot{\tau}}, y_{\dot{\tau}}, \theta) \tag{11}$$

$$H = \underbrace{V_{\tau \setminus \{0,s\}} \cup Z_{[0,s] \setminus \tau}}_{\text{augmentation set}} \tag{12}$$

We can now define an *ergodic* Gibbs sampler:

$$(\text{missing}|\text{param}): \quad \pi(h, y|v_0, v_{\bar{s}}, \theta, \lambda) \propto \pi(h, v_{\bar{s}}|v_0, y, \theta)\pi(y|\lambda) \tag{13}$$

$$(\text{param}|\text{missing}): \quad \pi(\theta|v_0, v_{\bar{s}}, h, y) \propto \pi(h, v_{\bar{s}}|v_0, y, \theta)\pi(\theta) \tag{14}$$

The second update is of particular interest!

# What to do about the path integral?

The augmented transition density contains an integral over a rough path:

$$\pi(h, v_{\bar{s}} | v_0, y, \theta) = a \exp\left[ -\int_0^s \varphi_\theta(\omega_\theta^{-1}(z_t), y_t)\, \mathrm{d}t \right] \tag{15}$$

Can't evaluate in finite time! Multiple possible approaches...

- *Pseudo-marginal* method, using unbiased estimators of the exponentiated path integral.
- even more augmentation...
- Here: Combine *Barker's algorithm* with *Bernoulli factories*! Keeps the state space as is.

# Methods

# Barker's algorithm.

Let $\pi(a)$ be a target density. Propose update $a^\dagger$ according to $\kappa(a^\dagger|a)$. The *Metropolis algorithm* accepts with probability

$$\min\left[1, \frac{\kappa(a|a^\dagger)}{\kappa(a^\dagger|a)}\frac{\pi(a^\dagger)}{\pi(a)}\right] \tag{16}$$

But there are other options! Barker's algorithm accepts with probability

$$\frac{\kappa(a|a^\dagger)\pi(a^\dagger)}{\kappa(a^\dagger|a)\pi(a) + \kappa(a|a^\dagger)\pi(a^\dagger)} \tag{17}$$

This results in higher asymptotic variance for a given proposal! So why bother?

# Enter the 2-coin algorithm.

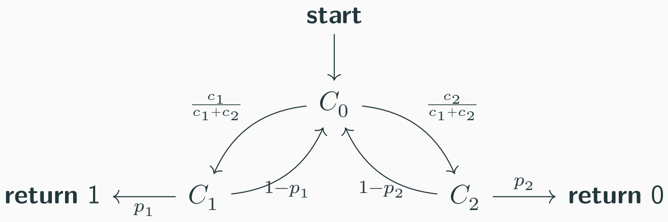



**Figure 6:** Probability flow diagram of the 2-coin algorithm.

Suppose we can generate coins with probability $p_1$ and $p_2$. Then, the 2-coin algorithm generates coins with odds

$$\frac{c_1 p_1}{c_1 p_1 + c_2 p_2} \tag{18}$$

This is an example of a *Bernoulli factory*.

**Notice!**

runtime $\to \infty$ as $p_1, p_2 \to 0$!

# 2-coin within Barker within Gibbs...

Assume there exist $\varphi_\theta^\downarrow$, $\varphi_\theta^\uparrow$ such that

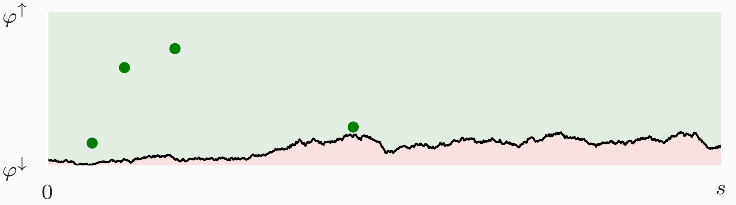
$$\varphi_\theta^\downarrow(z_{(\dot\tau,\ddot\tau)}, y_{\dot\tau}) \leq \varphi_\theta(\omega_\theta^{-1}(z_t), y_{\dot\tau}) \leq \varphi_\theta^\uparrow(z_{(\dot\tau,\ddot\tau)}, y_{\dot\tau}), \qquad t \in [\dot\tau, \ddot\tau) \tag{19}$$

Barker acceptance odds for parameter proposal $\theta^\dagger \sim \kappa(\theta^\dagger|\theta)$ update are

$$\frac{\alpha}{1-\alpha} = \underbrace{\frac{\pi(h, v_s|v_0, y, \theta^\dagger)}{\pi(h, v_s|v_0, y, \theta)}}_{\text{likelihood ratio}} \times \underbrace{\frac{\pi(\theta^\dagger)}{\pi(\theta)}}_{\text{prior odds}} \times \underbrace{\frac{\kappa(\theta|\theta^\dagger)}{\kappa(\theta^\dagger|\theta)}}_{\text{proposal odds}} \tag{20}$$

$$= \prod_{(\dot\tau \sim \ddot\tau)\in\tau} \frac{c_{\dot\tau}^\dagger}{c_{\dot\tau}} \underbrace{\frac{\exp\left[\int_{\dot\tau}^{\ddot\tau} \varphi_{\theta^\dagger}^\downarrow(z_{(\dot\tau,\ddot\tau)}, y_{\dot\tau}) - \varphi_{\theta^\dagger}(\omega_{\theta^\dagger}^{-1}(z_t), y_{\dot\tau})\,\mathrm{d}t\right]}{\exp\left[\int_{\dot\tau}^{\ddot\tau} \varphi_\theta^\downarrow(z_{(\dot\tau,\ddot\tau)}, y_{\dot\tau}) - \varphi_\theta(\omega_\theta^{-1}(z_t), y_{\dot\tau})\,\mathrm{d}t\right]}}_{\in(0,1)} \tag{21}$$

# Poisson coin within 2-coin within Barker within Gibbs.




Let $0 \leq f(t) \leq f^{\uparrow}$ for $t \in [0, s]$. Simulate a unit intensity Poisson process on $[0, s] \times [0, f^{\uparrow}]$. Then

$$\Pr\left[\text{all points above the graph of } f\right] = \exp\left[-\int_0^s f(t)\,\mathrm{d}t\right] \qquad (22)$$

So we only need to interpolate $f$ at a finite set of times. Apply this within a 2-coin algorithm to simulate coins with probability

$$\exp\left[\int_{\dot{\tau}}^{\ddot{\tau}} \varphi_\theta^{\downarrow}(z_{(\dot{\tau},\ddot{\tau})}, y_{\dot{\tau}}) - \varphi_\theta(\omega_\theta^{-1}(z_t), y_{\dot{\tau}})\,\mathrm{d}t\right] \qquad (23)$$

# Designing 2-Coin Algorithms

# Naive 2-coin algorithms don't scale.

Regardless of $\left|\theta^{\dagger} - \theta\right|$, under standard conditions

$$\lim_{s \to \infty} \exp\left[\int_0^s \varphi_\theta^{\downarrow}(z_{(\dot{\tau},\ddot{\tau})}, y_{\dot{\tau}}) - \varphi_\theta(\omega_\theta^{-1}(z_t), y_{\dot{\tau}})\,\mathrm{d}t\right] = 0 \tag{24}$$

almost surely, so the 2-coin algorithm slows down as the time series extends. But there are various two-coin algorithms resulting in the same coin probability...

# An alternative 2-coin algorithm.

Rearrange terms...

$$\begin{aligned}
&\frac{c^{\dagger}_{\dot\tau}}{c_{\dot\tau}}\frac{\exp\left[\int_{\dot\tau}^{\ddot\tau}\varphi^{\downarrow}_{\theta^\dagger}(z_{(\dot\tau,\ddot\tau)},y_{\dot\tau})-\varphi_{\theta^\dagger}(\omega^{-1}_{\theta^\dagger}(z_t),y_{\dot\tau})\,\mathrm{d}t\right]}{\exp\left[\int_{\dot\tau}^{\ddot\tau}\varphi^{\downarrow}_{\theta}(z_{(\dot\tau,\ddot\tau)},y_{\dot\tau})-\varphi_{\theta}(\omega^{-1}_{\theta}(z_t),y_{\dot\tau})\,\mathrm{d}t\right]}\\
&=\frac{c^{\dagger}_{\dot\tau}}{c_{\dot\tau}}\frac{\exp\left[-\int_{\dot\tau}^{\ddot\tau}0\vee\left(\varphi_{\theta^\dagger}(\omega^{-1}_{\theta^\dagger}(z_t),y_{\dot\tau})-\varphi_{\theta}(\omega^{-1}_{\theta}(z_t),y_{\dot\tau})\right)\mathrm{d}t\right]}{\exp\left[-\int_{\dot\tau}^{\ddot\tau}0\vee\left(\varphi_{\theta}(\omega^{-1}_{\theta}(z_t),y_{\dot\tau})-\varphi_{\theta^\dagger}(\omega^{-1}_{\theta^\dagger}(z_t),y_{\dot\tau})\right)\mathrm{d}t\right]}
\end{aligned}\tag{25}$$

By the *mean value theorem* and the *Cauchy-Schwarz* inequality,

$$\varphi_{\theta^\dagger}(\omega^{-1}_{\theta^\dagger}(z_t),y_{\dot\tau})-\varphi_{\theta}(\omega^{-1}_{\theta}(z_t),y_{\dot\tau}) \tag{26}$$

$$\leq \sup_{\mathrm{convhull}[\theta^\dagger,\theta],t}\left|\nabla_\theta\varphi_\theta(\omega^{-1}_{\theta}(z_t),y_{\dot\tau})\right|\left|\theta^\dagger-\theta\right| \tag{27}$$

$$\to 0 \qquad \text{as} \qquad \left|\theta^\dagger-\theta\right|\to 0 \tag{28}$$

# Bounding the new path integral.

We have to find

$$\sup_{\text{convhull}[\theta^\dagger,\theta],t} \left|\nabla_\theta \varphi_\theta(\omega_\theta^{-1}(z_t), y_{\dot{\tau}})\right| \tag{29}$$

$\nabla_\theta \varphi_\theta(\omega_\theta^{-1}(z_t), y_{\dot{\tau}})$ is usually not concave, so we take a symbolic approach. To find $\sup f(a)$, solve for

$$\sup \{f(a) : f'(a) = 0\} \tag{30}$$

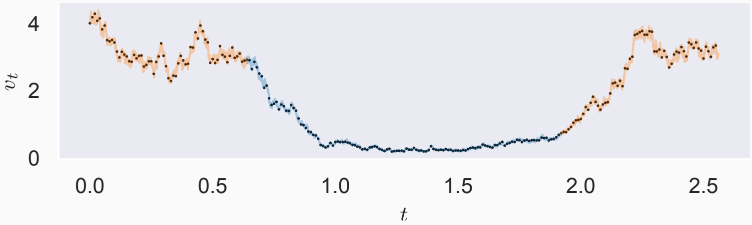
If $f'(a) = 0$ doesn't have an analytical solution, expand to $f = g + h$, and bound

$$\sup f \leq \sup g + \sup h \tag{31}$$

by finding roots of $g'$ and $h'$. Expressions are complicated even for simple models - use computer algebra systems to do the heavy lifting!

# Simulation Study

# Back to our data.




Consider a *generalized CIR model* with SDE

$$dV_t = \beta_{Y_t}(\mu_{Y_t} - V_t)\, dV_t + V_t^{3/4}\, dW_t \tag{32}$$

Where $V_t > 0$ almost surely. A priori

$$\beta_1, \beta_2, \mu_1, \mu_2 \sim \log \mathrm{N}\,[0, 1] \tag{33}$$

**Notice!**

Posterior is invariant to label inversions!

# Parameter traces.

# Regime inference.

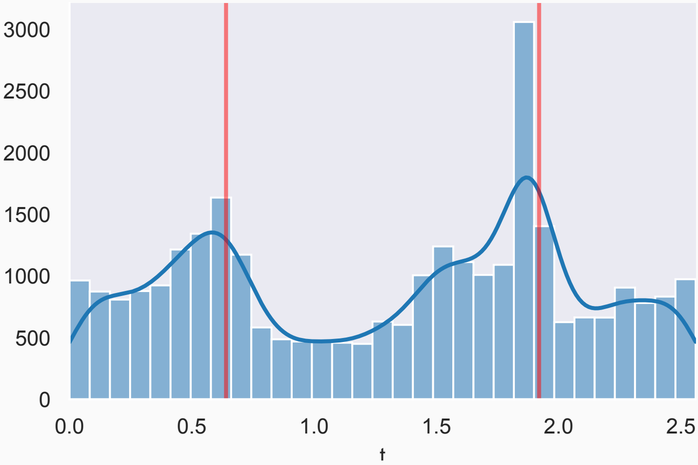



**Figure 8:** Posterior density of jump times in $Y$. Red lines correspond to the ground truth.

# In conclusion.

Our exact algorithm is slowed down...

- by dependence between $Y$ and $\theta$ due to Gibbs sampling.
- by large or variable drift, slowing down the 2-coin algorithm.

But other methods have the same downsides!

- integration of the posterior wrt $Y$ is intractable even for tractable diffusions, so some form of conditional updating is unavoidable.
- accuracy of approximate methods degrades when drift is variable.

# Outlook

# Open questions.

Work in progress...

- finish algorithm for general $\sigma_\theta(V_t, Y_t)$.
- apply to real data (misspecification!).
- benchmark against pseudo-marginal implementation.
- which rate of posterior contraction gives a scalable algorithm?
- MAP estimation for $Y$.
- try more than 2 states.

Important, but probably intractable...

- optimal scaling. Tradeoff between 2-coin and MCMC efficiency!

## Stay tuned for the pre-print!