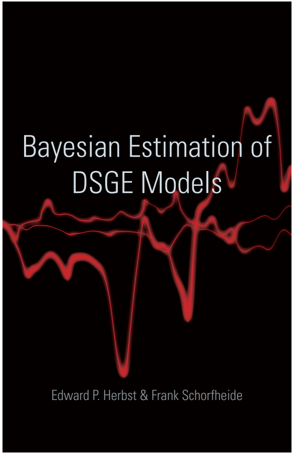
Bayesian Computations for DSGE Models

Frank Schorfheide*

*University of Pennsylvania, PIER, CEPR, and NBER

October 23, 2017

This Lecture is Based on



https://web.sas.upenn.edu/schorf/
companion-web-site-bayesian-estimation-of-dsge-models

Some Background

- DSGE model: dynamic model of the macroeconomy, indexed by θ – vector of preference and technology parameters. Used for forecasting, policy experiments, interpreting past events.
- Ingredients of Bayesian Analysis:
  - Likelihood function $p(Y|\theta)$
  - Prior density $p(\theta)$
  - Marginal data density $p(Y) = \int p(Y|\theta)p(\theta)d\phi$
- Bayes Theorem:

$$p(\theta|Y) = \frac{p(Y|\theta)p(\theta)}{p(Y)} \propto p(Y|\theta)p(\theta)$$

Some Background

- Implementation: usually by generating a sequence of draws (not necessarily iid) from posterior

$$\theta^i \sim p(\theta|Y), \quad i = 1, \ldots, N$$

- Algorithms: direct sampling, accept/reject sampling, importance sampling, Markov chain Monte Carlo sampling, sequential Monte Carlo sampling...

- Draws can then be transformed into objects of interest, $h(\theta^i)$, and under suitable conditions a Monte Carlo average of the form

$$\bar{h}_N = \frac{1}{N} \sum_{i=1}^{N} h(\theta^i) \approx \mathbb{E}_\pi[h] \int h(\theta) p(\theta|Y) d\theta.$$

- Strong law of large numbers (SLLN), central limit theorem (CLT)...

Bayesian Inference – Decision Making

- The posterior expected loss of decision $\delta(\cdot)$:

$$\rho\big(\delta(\cdot)|Y\big) = \int_{\Theta} L\big(\theta, \delta(Y)\big) p(\theta|Y) d\theta.$$

- Bayes decision minimizes the posterior expected loss:

$$\delta^*(Y) = \text{argmin}_d \; \rho\big(\delta(\cdot)|Y\big).$$

- Approximate $\rho\big(\delta(\cdot)|Y\big)$ by a Monte Carlo average

$$\bar{\rho}_N\big(\delta(\cdot)|Y\big) = \frac{1}{N}\sum_{i=1}^{N} L\big(\theta^i, \delta(\cdot)\big).$$

- Then compute

$$\delta^*_N(Y) = \text{argmin}_d \; \bar{\rho}_N\big(\delta(\cdot)|Y\big).$$

Bayesian Inference

- Point estimation:
  - Quadratic loss: posterior mean
  - Absolute error loss: posterior median
- Interval/Set estimation $\mathbb{P}_{\pi}\{\theta \in C(Y)\} = 1 - \alpha$:
  - highest posterior density sets
  - equal-tail-probability intervals

Computational Challenges

- Numerical solution of model leads to
  state-space representation $\Longrightarrow$ likelihood approximation $\Longrightarrow$ posterior sampler.
- "Standard" approach for *(linearized) models*
  - Model solution: log-linearize and use linear rational expectations system solver.
  - Evaluation of $p(Y|\theta)$: Kalman filter
  - Posterior draws θ^i: MCMC
- Book reviews the "standard approach", but also studies more recently developed sequential Monte Carlo (SMC) techniques.

Sequential Monte Carlo (SMC) Methods

SMC can help to

- Approximate the likelihood function (particle filtering): Gordon, Salmond, and Smith (1993) ... Fernandez-Villaverde and Rubio-Ramirez (2007)
- approximate the posterior of θ: Chopin (2002) ... Durham and Geweke (2013) ... Creal (2007), Herbst and Schorfheide (2014)
- or both: SMC^2: Chopin, Jacob, and Papaspiliopoulos (2012) ... Herbst and Schorfheide (2015)

Review – Importance Sampling

Importance Sampling

- Approximate $\pi(\cdot)$ by using a different, tractable density $g(\theta)$ that is easy to sample from.
- For more general problems, posterior density may be unnormalized. So we write

$$\pi(\theta) = \frac{p(Y|\theta)p(\theta)}{p(Y)} = \frac{f(\theta)}{\int f(\theta)d\theta}.$$

- Importance sampling is based on the identity

$$E_\pi[h(\theta)] = \int h(\theta)\pi(\theta)d\theta = \frac{\int_\Theta h(\theta)\frac{f(\theta)}{g(\theta)}g(\theta)d\theta}{\int_\Theta \frac{f(\theta)}{g(\theta)}g(\theta)d\theta}.$$

- (Unnormalized) importance weight:

$$w(\theta) = \frac{f(\theta)}{g(\theta)}.$$

Importance Sampling

1. For $i = 1$ to N, draw $\theta^i \stackrel{iid}{\sim} g(\theta)$ and compute the unnormalized importance weights

$$w^i = w(\theta^i) = \frac{f(\theta^i)}{g(\theta^i)}.$$

2. Compute the normalized importance weights

$$W^i = \frac{w^i}{\frac{1}{N}\sum_{i=1}^N w^i}.$$

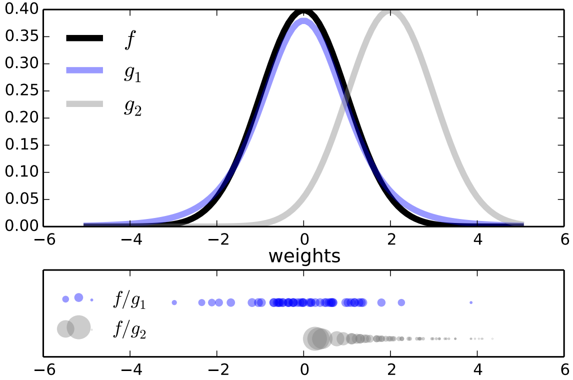
An approximation of $\mathbb{E}_\pi[h(\theta)]$ is given by

$$\bar{h}_N = \frac{1}{N}\sum_{i=1}^N W^i h(\theta^i).$$

Illustration

If θ^i's are draws from $g(\cdot)$ then

$$\mathbb{E}_\pi[h] \approx \frac{\frac{1}{N}\sum_{i=1}^N h(\theta^i)w(\theta^i)}{\frac{1}{N}\sum_{i=1}^N w(\theta^i)}, \quad w(\theta) = \frac{f(\theta)}{g(\theta)}.$$



Accuracy

- Since we are generating *iid* draws from $g(\theta)$, it's fairly straightforward to derive a CLT:

$$\sqrt{N}(\bar{h}_N - \mathbb{E}_\pi[h]) \Longrightarrow N\big(0, \Omega(h)\big), \quad \text{where} \quad \Omega(h) = \mathbb{V}_g[(\pi/g)(h - \mathbb{E}_\pi[h])].$$

- Using a crude approximation (see, e.g., Liu (2008)), we can factorize $\Omega(h)$ as follows:

$$\Omega(h) \approx \mathbb{V}_\pi[h]\big(1 + \mathbb{V}_g[\pi/g]\big).$$

  The approximation highlights that the larger the variance of the importance weights, the less accurate the Monte Carlo approximation relative to the accuracy that could be achieved with an *iid* sample from the posterior.

- Users often monitor

$$ESS = N\frac{\mathbb{V}_\pi[h]}{\Omega(h)} \approx \frac{N}{1 + \mathbb{V}_g[\pi/g]}.$$

Likelihood Approximation

State-Space Representation and Likelihood

- Measurement Equation:

$$y_t = \Psi(s_t; \theta) \underbrace{+u_t}_{\text{optional}}$$

- State transition:

$$s_t = \Phi(s_{t-1}, \epsilon_t; \theta)$$

- Joint density for the observations and latent states:

$$p(Y_{1:T}, S_{1:T}|\theta) = \prod_{t=1}^{T} p(y_t, s_t|Y_{1:t-1}, S_{1:t-1}, \theta) = \prod_{t=1}^{T} p(y_t|s_t, \theta)p(s_t|s_{t-1}, \theta).$$

- Need to compute the marginal $p(Y_{1:T}|\theta)$.

Filtering - General Idea

- State-space representation of nonlinear DSGE model

$$\begin{aligned} \text{Measurement Eq.} &: \quad y_t = \Psi(s_t, t; \theta) + u_t, \quad u_t \sim F_u(\cdot;\theta) \\ \text{State Transition} &: \quad s_t = \Phi(s_{t-1}, \epsilon_t; \theta), \quad \epsilon_t \sim F_\epsilon(\cdot;\theta). \end{aligned}$$

- Likelihood function:

$$p(Y_{1:T}|\theta) = \prod_{t=1}^{T} p(y_t|Y_{1:t-1}, \theta)$$

- A filter generates a sequence of conditional distributions $s_t|Y_{1:t}$.
- Iterations:
  - Initialization at time $t-1$: $p(s_{t-1}|Y_{1:t-1}, \theta)$
  - Forecasting t given $t-1$:
    1. Transition equation: $p(s_t|Y_{1:t-1}, \theta) = \int p(s_t|s_{t-1}, Y_{1:t-1}, \theta) p(s_{t-1}|Y_{1:t-1}, \theta) ds_{t-1}$
    2. Measurement equation: $p(y_t|Y_{1:t-1}, \theta) = \int p(y_t|s_t, Y_{1:t-1}, \theta) p(s_t|Y_{1:t-1}, \theta) ds_t$
  - Updating with Bayes theorem. Once y_t becomes available:

$$p(s_t|Y_{1:t}, \theta) = p(s_t|y_t, Y_{1:t-1}, \theta) = \frac{p(y_t|s_t, Y_{1:t-1}, \theta) p(s_t|Y_{1:t-1}, \theta)}{p(y_t|Y_{1:t-1}, \theta)}$$

Conditional Distributions for Kalman Filter (Linear Gaussian State-Space Model)

	Distribution	Mean and Variance
$s_{t-1}\|(Y_{1:t-1},\theta)$	$N\left(\bar{s}_{t-1\|t-1}, P_{t-1\|t-1}\right)$	Given from Iteration $t-1$
$s_t\|(Y_{1:t-1},\theta)$	$N\left(\bar{s}_{t\|t-1}, P_{t\|t-1}\right)$	$\bar{s}_{t\|t-1} = \Phi_1 \bar{s}_{t-1\|t-1}$ $P_{t\|t-1} = \Phi_1 P_{t-1\|t-1}\Phi_1' + \Phi_\epsilon \Sigma_\epsilon \Phi_\epsilon'$
$y_t\|(Y_{1:t-1},\theta)$	$N\left(\bar{y}_{t\|t-1}, F_{t\|t-1}\right)$	$\bar{y}_{t\|t-1} = \Psi_0 + \Psi_1 t + \Psi_2 \bar{s}_{t\|t-1}$ $F_{t\|t-1} = \Psi_2 P_{t\|t-1}\Psi_2' + \Sigma_u$
$s_t\|(Y_{1:t},\theta)$	$N\left(\bar{s}_{t\|t}, P_{t\|t}\right)$	$\bar{s}_{t\|t} = \bar{s}_{t\|t-1} + P_{t\|t-1}\Psi_2' F_{t\|t-1}^{-1}(y_t - \bar{y}_{t\|t-1})$ $P_{t\|t} = P_{t\|t-1} - P_{t\|t-1}\Psi_2' F_{t\|t-1}^{-1}\Psi_2 P_{t\|t-1}$

Bootstrap Particle Filter

1. **Initialization.** Draw the initial particles from the distribution $s_0^j \stackrel{iid}{\sim} p(s_0)$ and set $W_0^j = 1$, $j = 1, \ldots, M$.
2. **Recursion.** For $t = 1, \ldots, T$:
   1. **Forecasting s_t.** Propagate the period $t-1$ particles $\{s_{t-1}^j, W_{t-1}^j\}$ by iterating the state-transition equation forward:

      $$\tilde{s}_t^j = \Phi(s_{t-1}^j, \epsilon_t^j; \theta), \quad \epsilon_t^j \sim F_\epsilon(\cdot; \theta). \tag{1}$$

      An approximation of $\mathbb{E}[h(s_t)|Y_{1:t-1}, \theta]$ is given by

      $$\hat{h}_{t,M} = \frac{1}{M} \sum_{j=1}^{M} h(\tilde{s}_t^j) W_{t-1}^j. \tag{2}$$

Bootstrap Particle Filter

1. **Initialization.**
2. **Recursion.** For $t = 1, \ldots, T$:
   1. **Forecasting s_t.**
   2. **Forecasting y_t.** Define the incremental weights

$$\tilde{w}_t^j = p(y_t|\tilde{s}_t^j, \theta). \tag{3}$$

The predictive density $p(y_t|Y_{1:t-1}, \theta)$ can be approximated by

$$\hat{p}(y_t|Y_{1:t-1}, \theta) = \frac{1}{M}\sum_{j=1}^{M} \tilde{w}_t^j W_{t-1}^j. \tag{4}$$

If the measurement errors are $N(0, \Sigma_u)$ then the incremental weights take the form

$$\tilde{w}_t^j = (2\pi)^{-n/2}|\Sigma_u|^{-1/2}\exp\left\{-\frac{1}{2}\big(y_t - \Psi(\tilde{s}_t^j, t; \theta)\big)'\Sigma_u^{-1}\big(y_t - \Psi(\tilde{s}_t^j, t; \theta)\big)\right\}, \tag{5}$$

where n here denotes the dimension of y_t.

Bootstrap Particle Filter

1. **Initialization.**
2. **Recursion.** For $t = 1, \ldots, T$:
   1. **Forecasting s_t.**
   2. **Forecasting y_t.** Define the incremental weights

$$\tilde{w}_t^j = p(y_t|\tilde{s}_t^j, \theta). \tag{6}$$

   3. **Updating.** Define the normalized weights

$$\tilde{W}_t^j = \frac{\tilde{w}_t^j W_{t-1}^j}{\frac{1}{M}\sum_{j=1}^M \tilde{w}_t^j W_{t-1}^j}. \tag{7}$$

An approximation of $\mathbb{E}[h(s_t)|Y_{1:t}, \theta]$ is given by

$$\tilde{h}_{t,M} = \frac{1}{M}\sum_{j=1}^M h(\tilde{s}_t^j)\tilde{W}_t^j. \tag{8}$$

Bootstrap Particle Filter

1. **Initialization.**
2. **Recursion.** For $t = 1, \dots, T$:
   1. **Forecasting s_t.**
   2. **Forecasting y_t.**
   3. **Updating.**
   4. **Selection (Optional).** Resample the particles via multinomial resampling. Let $\{s_t^j\}_{j=1}^M$ denote M iid draws from a multinomial distribution characterized by support points and weights $\{\tilde{s}_t^j, \tilde{W}_t^j\}$ and set $W_t^j = 1$ for $j =, 1 \dots, M$.
   An approximation of $\mathbb{E}[h(s_t)|Y_{1:t}, \theta]$ is given by

$$\bar{h}_{t,M} = \frac{1}{M} \sum_{j=1}^{M} h(s_t^j) W_t^j. \tag{9}$$

Likelihood Approximation

- The approximation of the log likelihood function is given by

$$\ln \hat{p}(Y_{1:T}|\theta) = \sum_{t=1}^{T} \ln \left(\frac{1}{M} \sum_{j=1}^{M} \tilde{w}_t^j W_{t-1}^j \right). \tag{10}$$

- One can show that the approximation of the likelihood function is unbiased.
- This implies that the approximation of the log likelihood function is downward biased.

The Role of Measurement Errors

- Measurement errors may not be intrinsic to DSGE model.
- Bootstrap filter needs non-degenerate $p(y_t|s_t, \theta)$ for incremental weights to be well defined.
- Decreasing the measurement error variance Σ_u, holding everything else fixed, increases the variance of the particle weights, and reduces the accuracy of Monte Carlo approximation.

Generic Particle Filter

1. **Initialization.** Same as BS PF
2. **Recursion.** For $t = 1, \ldots, T$:
   1. **Forecasting s_t.** Draw $\tilde{s}_t^j$ from density $g_t(\tilde{s}_t | s_{t-1}^j, \theta)$ and define

$$\omega_t^j = \frac{p(\tilde{s}_t^j | s_{t-1}^j, \theta)}{g_t(\tilde{s}_t^j | s_{t-1}^j, \theta)}. \quad (11)$$

      An approximation of $\mathbb{E}[h(s_t) | Y_{1:t-1}, \theta]$ is given by

$$\hat{h}_{t,M} = \frac{1}{M} \sum_{j=1}^{M} h(\tilde{s}_t^j) \omega_t^j W_{t-1}^j. \quad (12)$$

   2. **Forecasting y_t.** Define the incremental weights

$$\tilde{w}_t^j = p(y_t | \tilde{s}_t^j, \theta) \omega_t^j. \quad (13)$$

      The predictive density $p(y_t | Y_{1:t-1}, \theta)$ can be approximated by

$$\hat{p}(y_t | Y_{1:t-1}, \theta) = \frac{1}{M} \sum_{j=1}^{M} \tilde{w}_t^j W_{t-1}^j. \quad (14)$$

   3. (...)

Adapting the Generic PF

- Conditionally-optimal importance distribution:

$$g_t(\tilde{s}_t|s^j_{t-1}) = p(\tilde{s}_t|y_t, s^j_{t-1}).$$

  This is the posterior of s_t given s^j_{t-1}. Typically infeasible, but a good benchmark.

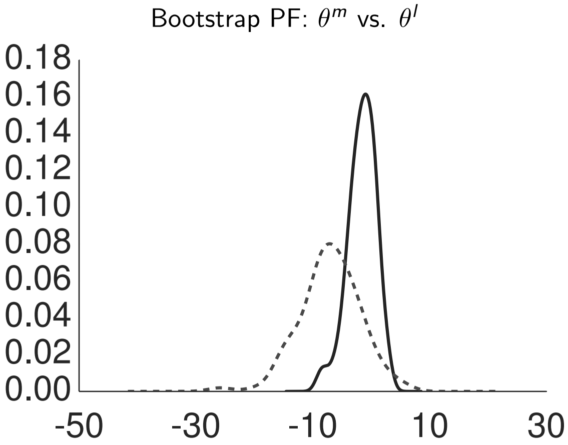
- Approximately conditionally-optimal distributions: from linearize version of DSGE model or approximate nonlinear filters.

- Conditionally-linear models: do Kalman filter updating on a subvector of s_t. Example:

$$\begin{aligned} y_t &= \Psi_0(m_t) + \Psi_1(m_t)t + \Psi_2(m_t)s_t + u_t, \quad u_t \sim N(0, \Sigma_u), \\ s_t &= \Phi_0(m_t) + \Phi_1(m_t)s_{t-1} + \Phi_\epsilon(m_t)\epsilon_t, \quad \epsilon_t \sim N(0, \Sigma_\epsilon), \end{aligned}$$

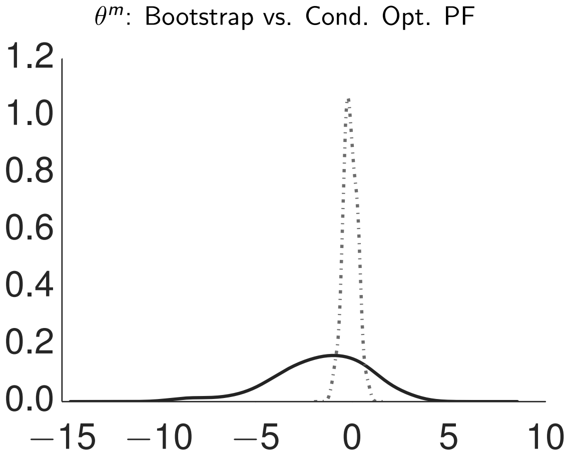
  where m_t follows a discrete Markov-switching process.

Distribution of Log-Likelihood Approximation Errors



Notes: Density estimate of $\hat{\Delta}_1 = \ln \hat{p}(Y_{1:T}|\theta) - \ln p(Y_{1:T}|\theta)$ based on $N_{run} = 100$ runs of the PF. Solid line is $\theta = \theta^m$; dashed line is $\theta = \theta^l$ ($M = 40,000$).

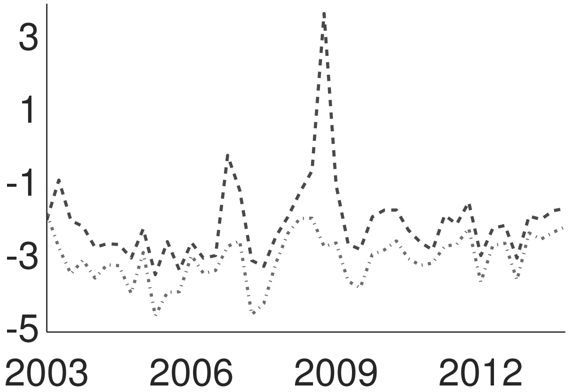
Distribution of Log-Likelihood Approximation Errors



Notes: Density estimate of $\hat{\Delta}_1 = \ln \hat{p}(Y_{1:T}|\theta) - \ln p(Y_{1:T}|\theta)$ based on $N_{run} = 100$ runs of the PF. Solid line is bootstrap particle filter ($M = 40,000$); dotted line is conditionally optimal particle filter ($M = 400$).

Great Recession and Beyond

Log Standard Dev of Log-Likelihood Increments



Notes: Solid lines represent results from Kalman filter. Dashed lines correspond to bootstrap particle filter ($M = 40,000$) and dotted lines correspond to conditionally-optimal particle filter ($M = 400$). Results are based on $N_{run} = 100$ runs of the filters.

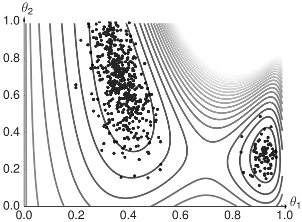
Posterior Sampling

Implementation: Sampling from Posterior

- DSGE model posteriors are often non-elliptical, e.g., multimodal posteriors may arise

because it is difficult to

- disentangle internal and external propagation mechanisms;
- disentangle the relative importance of shocks.



- Economic Example: is wage growth persistent because
  1. wage setters find it very costly to adjust wages?
  2. exogenous shocks affect the substitutability of labor inputs and hence markups?

Sampling from Posterior

- If posterior distributions are irregular, standard MCMC methods can be inaccurate (examples will follow).
- SMC samplers often generate more precise approximations of posteriors in the same amount of time.
- SMC can be parallelized.
- SMC = importance sampling on steroids
- For now we assume that the likelihood evaluation is exact.

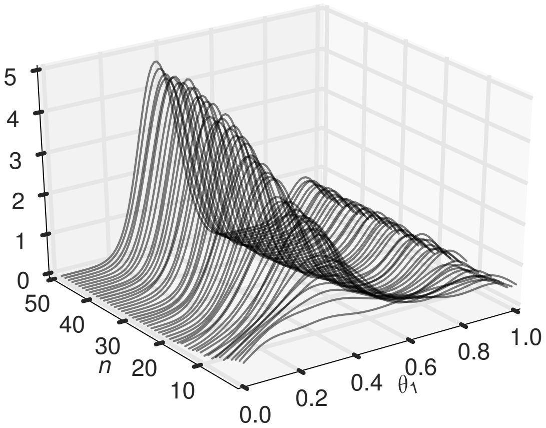
From Importance Sampling to Sequential Importance Sampling

- In general, it's hard to construct a good proposal density $g(\theta)$,
- especially if the posterior has several peaks and valleys.
- Idea - Part 1: it might be easier to find a proposal density for

$$\pi_n(\theta) = \frac{[p(Y|\theta)]^{\phi_n} p(\theta)}{\int [p(Y|\theta)]^{\phi_n} p(\theta) d\theta} = \frac{f_n(\theta)}{Z_n}.$$

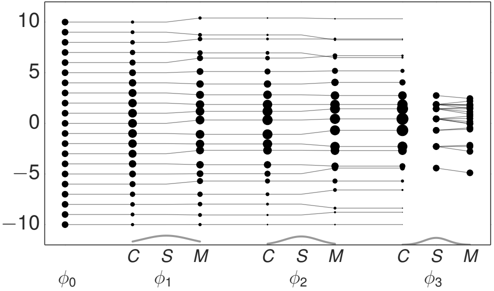
  at least if ϕ_n is close to zero.
- Idea - Part 2: We can try to turn a proposal density for π_n into a proposal density for π_{n+1} and iterate, letting $\phi_n \longrightarrow \phi_N = 1$.

Illustration: Tempered Posteriors of θ_1



$$\pi_n(\theta) = \frac{[p(Y|\theta)]^{\phi_n} p(\theta)}{\int [p(Y|\theta)]^{\phi_n} p(\theta) d\theta} = \frac{f_n(\theta)}{Z_n}, \quad \phi_n = \left(\frac{n}{N_\phi}\right)^\lambda$$

SMC Algorithm: A Graphical Illustration



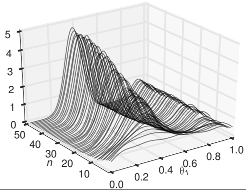
- $\pi_n(\theta)$ is represented by a swarm of particles $\{\theta_n^i, W_n^i\}_{i=1}^N$:

$$\bar{h}_{n,N} = \frac{1}{N}\sum_{i=1}^{N} W_n^i h(\theta_n^i) \xrightarrow{a.s.} \mathbb{E}_{\pi_n}[h(\theta_n)].$$

- C is Correction; S is Selection; and M is Mutation.

Remarks

- Correction Step:
  - reweight particles from iteration $n-1$ to create importance sampling approximation of $\mathbb{E}_{\pi_n}[h(\theta)]$
- Selection Step: the resampling of the particles
  - (good) equalizes the particle weights and thereby increases accuracy of subsequent importance sampling approximations;
  - (not good) adds a bit of noise to the MC approximation.
- Mutation Step: changes particle values
  - adapts particles to posterior $\pi_n(\theta)$;
  - imagine we don't do it: then we would be using draws from prior $p(\theta)$ to approximate posterior $\pi(\theta)$, which can't be good!



More on Transition Kernel in Mutation Step

- Transition kernel $K_n(\theta|\hat{\theta}_{n-1};\zeta_n)$: generated by running M steps of a Metropolis-Hastings algorithm.
- Lessons from DSGE model MCMC:
  - blocking of parameters can reduces persistence of Markov chain;
  - mixture proposal density avoids "getting stuck."
- Blocking: Partition the parameter vector θ_n into N_{blocks} equally sized blocks, denoted by $\theta_{n,b}$, $b = 1, \dots, N_{blocks}$. (We generate the blocks for $n = 1, \dots, N_\phi$ randomly prior to running the SMC algorithm.)
- Example: random walk proposal density:

$$\vartheta_b | (\theta^i_{n,b,m-1}, \theta^i_{n,-b,m}, \Sigma^*_{n,b}) \sim N\left(\theta^i_{n,b,m-1}, c_n^2 \Sigma^*_{n,b}\right).$$

Application: Estimation of Smets and Wouters (2007) Model

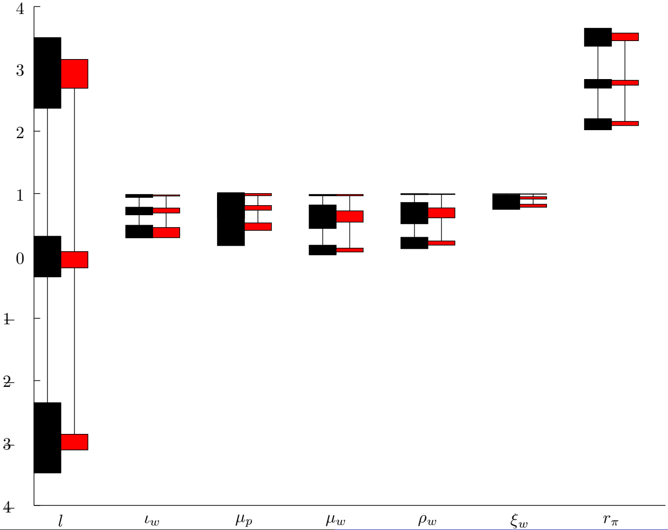
- Benchmark macro model, has been estimated many (many) times.
- "Core" of many larger-scale models.
- 36 estimated parameters.
- RWMH: 10 million draws (5 million discarded); SMC: 500 stages with 12,000 particles.
- We run the RWM (using a particular version of a parallelized MCMC) and the SMC algorithm on 24 processors for the same amount of time.
- We estimate the SW model twenty times using RWM and SMC and get essentially identical results.

Application: Estimation of Smets and Wouters (2007) Model

- More interesting question: how does quality of posterior simulators change as one makes the priors more diffuse?

- Replace Beta by Uniform distributions; increase variances of parameters with Gamma and Normal prior by factor of 3.

SW Model with DIFFUSE Prior: Estimation stability RWH (black) versus SMC (red)



A Measure of Effective Number of Draws

- Suppose we could generate *iid* N_{eff} draws from posterior, then

$$\hat{\mathbb{E}}_\pi[\theta] \stackrel{approx}{\sim} N\left(\mathbb{E}_\pi[\theta], \frac{1}{N_{eff}}\mathbb{V}_\pi[\theta]\right).$$

- We can measure the variance of $\hat{\mathbb{E}}_\pi[\theta]$ by running SMC and RWM algorithm repeatedly.

- Then,

$$N_{eff} \approx \frac{\mathbb{V}_\pi[\theta]}{\mathbb{V}\left[\hat{\mathbb{E}}_\pi[\theta]\right]}$$

Effective Number of Draws

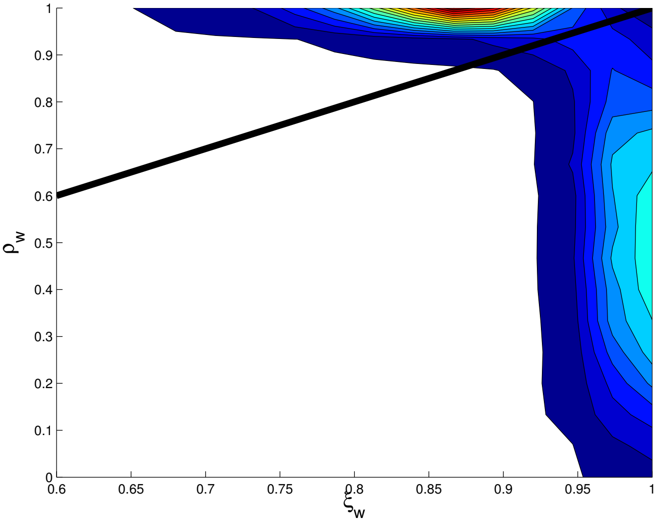
	SMC			RWMH		
Parameter	Mean	STD(Mean)	N_{eff}	Mean	STD(Mean)	N_{eff}
σ_l	3.06	0.04	1058	3.04	0.15	60
l	-0.06	0.07	732	-0.01	0.16	177
ι_p	0.11	0.00	637	0.12	0.02	19
h	0.70	0.00	522	0.69	0.03	5
Φ	1.71	0.01	514	1.69	0.04	10
r_π	2.78	0.02	507	2.76	0.03	159
ρ_b	0.19	0.01	440	0.21	0.08	3
φ	8.12	0.16	266	7.98	1.03	6
σ_p	0.14	0.00	126	0.15	0.04	1
ξ_p	0.72	0.01	91	0.73	0.03	5
ι_w	0.73	0.02	87	0.72	0.03	36
μ_p	0.77	0.02	77	0.80	0.10	3
ρ_w	0.69	0.04	49	0.69	0.09	11
μ_w	0.63	0.05	49	0.63	0.09	11
ξ_w	0.93	0.01	43	0.93	0.02	8

A Closer Look at the Posterior: Two Modes

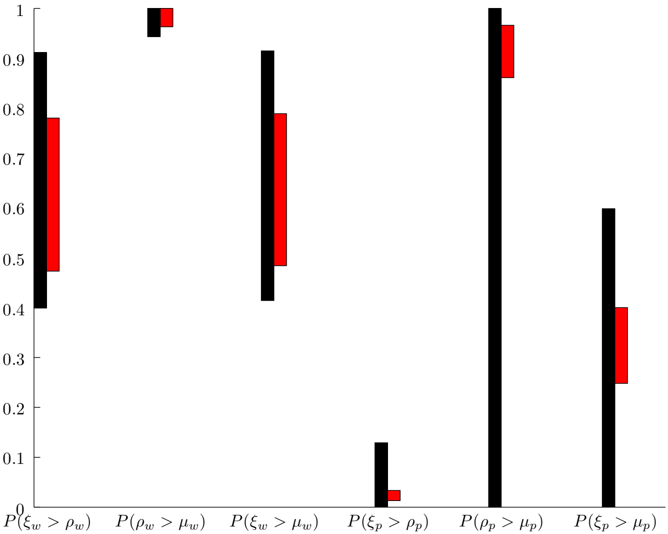
Parameter	Mode 1	Mode 2
ξ_w	0.844	0.962
ι_w	0.812	0.918
ρ_w	0.997	0.394
μ_w	0.978	0.267
Log Posterior	-804.14	-803.51

- Mode 1 implies that wage persistence is driven by extremely exogenous persistent wage markup shocks.
- Mode 2 implies that wage persistence is driven by endogenous amplification of shocks through the wage Calvo and indexation parameter.
- SMC is able to capture the two modes.

A Closer Look at the Posterior: Internal ξ_w versus External ρ_w Propagation



Stability of Posterior Computations: RWH (black) versus SMC (red)



Embedding PF Likelihoods into Posterior Samplers

- Particle MCMC, SMC^2.
- Distinguish between:
  - $\{p(Y|\theta), p(\theta|Y), p(Y)\}$, which are related according to:

$$p(\theta|Y) = \frac{p(Y|\theta)p(\theta)}{p(Y)}, \quad p(Y) = \int p(Y|\theta)p(\theta)d\theta$$

  - $\{\hat{p}(Y|\theta), \hat{p}(\theta|Y), \hat{p}(Y)\}$, which are related according to:

$$\hat{p}(\theta|Y) = \frac{\hat{p}(Y|\theta)p(\theta)}{\hat{p}(Y)}, \quad \hat{p}(Y) = \int \hat{p}(Y|\theta)p(\theta)d\theta.$$

- Surprising result (Andrieu, Docet, and Holenstein, 2010): under certain conditions we can replace $p(Y|\theta)$ by $\hat{p}(Y|\theta)$ and still obtain draws from $p(\theta|Y)$.