

# Estimating DSGE models using the Metropolis Algorithm

June 16, 2009

# The Plan

1. Talk about Bayesian methods
2. Solve a simple DSGE model
3. Simulate data
4. Estimate posterior by MCMC methods (Random Walk Metropolis Algorithm)
5. Identification

Most of today's lecture can be found in An and Schorfheide (2007)

Code and slides available at

[http://www.kris-nimark.net/TS\\_UPF\\_2009.html](http://www.kris-nimark.net/TS_UPF_2009.html)

## From last time

We can find the maximum likelihood estimate of a parameter vector  $\Theta$  from the state space system

$$\begin{aligned}X_t &= AX_{t-1} + Cu_t \\Z_t &= DX_t + v_t\end{aligned}$$

where  $A, C, D$  and  $E(v_t v_t')$  are functions of  $\theta$ .

- ▶ We are normally interested in more than point estimates: How confident are we about the estimated parameters?

## Inference with ML

One approach that works well if likelihood function is well behaved is to compute the Hessian (i.e. the matrix of second derivatives) of the likelihood function. The inverse of the negative of the Hessian gives an estimate of the covariance matrix of the parameter vector  $\theta$ , i.e.

$$E \left( \hat{\theta} - \theta \right) \left( \hat{\theta} - \theta \right)' \simeq \left[ -\frac{\partial^2 \mathcal{L}(Z | \theta)}{\partial \theta \partial \theta'} \Big|_{\theta = \hat{\theta}} \right]^{-1}$$

- ▶ In practice, this approach can be quite sensitive to step length when taking numerical derivatives

## Alternative approach: Bayesian methods

Parameters have distributions, just like the shocks

Frequentist vs Bayesian statistics

- ▶ Variance of estimator vs variance of parameter

Most of Bayesian econometrics consists of simulating distribution of parameters using numerical methods.

Bayesian methods have several advantages:

- ▶ Facilitates incorporating information from outside of sample
- ▶ Easy to compute confidence/probability intervals of functions of parameters

# The Random-Walk Metropolis Algorithm

1. Start with an arbitrary value  $\theta_0$
2. Update from  $\theta_j$  to  $\theta_{j+1}$  ( $j = 1, 2, \dots, J$ ) by
  - 2.1 Generate  $\theta^* \sim N(\theta_j, \Sigma)$
  - 2.2 Define

$$\alpha = \min \left( \frac{L(Y | \theta^*)}{L(Y | \theta_j)}, 1 \right) \quad (1)$$

- 2.3 Take

$$\theta_{j+1} = \left\{ \begin{array}{l} \theta^* \text{ with probability } \alpha \\ \theta_j \text{ otherwise} \end{array} \right\}$$

## Practical issues

How do we choose starting value  $\theta^{(0)}$ ?

- ▶ Asymptotically this shouldn't matter, but matters greatly in practice
- ▶ One (good) option is to first maximize the likelihood and choose the MLE of  $\Theta$  as  $\theta^{(0)}$

How do we choose covariance  $\Sigma$ ?

- ▶ Again, asymptotically this shouldn't matter, but matters greatly in practice
- ▶ Need to scale to achieve optimal acceptance ratio of 23%-40%
  - ▶ Scale for each element of  $\theta$
  - ▶ Scale overall variance

## A (very) simple DSGE model

$$\begin{aligned}x_t &= \rho x_{t-1} + u_t^x \\y_t &= E_t(y_{t+1}) - \frac{1}{\gamma} [r_t - E_t(\pi_{t+1})] + u_t^y \\\pi_t &= E_t(\pi_{t+1}) + \kappa [y_t - x_t] + u_t^\pi \\r_t &= \phi_\pi \pi_t\end{aligned}$$

We need to put model in state space form

## A (very) simple DSGE model

Substitute in the interest rate in the Euler equation

$$\begin{aligned}x_t &= \rho x_{t-1} + u_t^x \\y_t &= E_t(y_{t+1}) - \frac{1}{\gamma} [\phi_\pi \pi_t - E_t(\pi_{t+1})] + u_t^y \\\pi_t &= E_t(\pi_{t+1}) + \kappa [y_t - x_t] + u_t^\pi\end{aligned}$$

## Solving model using method of undetermined coefficients

Conjecture that model can be put in the form

$$x_t = \rho x_{t-1} + u_t^x$$

$$y_t = ax_t + u_t^y$$

$$\pi_t = bx_t + u_t^\pi$$

Why is this a good guess?

## Solving model using method of undetermined coefficients

Substitute in conjectured form of solution (ignoring the shocks  $u_t^y$  and  $u_t^\pi$  for now) into structural equation

$$\begin{aligned}ax_t &= a\rho x_t - \frac{1}{\gamma} [\phi_\pi b x_t - b\rho x_t] \\bx_t &= b\rho x_t + \kappa [ax_t - x_t]\end{aligned}$$

where we used that  $x_t = \rho x_{t-1} + u_t^x$  implies that  $E[x_{t+1} | x_t] = \rho x_t$

## Solving model using method of undetermined coefficients

Equate coefficients on right and left hand side

$$\begin{aligned}a &= a\rho - \frac{1}{\gamma}\phi_{\pi}b + \frac{1}{\gamma}b\rho \\ b &= b\rho + \kappa[a - 1]\end{aligned}$$

or

$$\begin{bmatrix} (1 - \rho) & \frac{1}{\gamma}(\phi_{\pi} - \rho) \\ -\kappa & (1 - \rho) \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} 0 \\ -\kappa \end{bmatrix}$$

## Solving model using method of undetermined coefficients

Solve for  $a$  and  $b$

$$\begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} (1 - \rho) & \frac{1}{\gamma} (\phi_\pi - \rho) \\ -\kappa & (1 - \rho) \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ -\kappa \end{bmatrix}$$

or

$$\begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} -\kappa \frac{\phi - \rho}{-c} \\ \kappa \gamma \frac{1 - \rho}{-c} \end{bmatrix}$$

where  $c = \gamma - \kappa\rho - 2\gamma\rho + \kappa\phi + \gamma\rho^2 < 0$

## A (very) simple DSGE model

$$\begin{aligned}x_t &= \rho x_{t-1} + u_t^x \\y_t &= -\kappa \frac{\rho - \phi_\pi}{c} x_t + u_t^y \\\pi_t &= \kappa \gamma \frac{\rho - 1}{c} x_t + u_t^\pi\end{aligned}$$

We then have all ingredients for state space form

## A (very) simple DSGE model

$$\begin{aligned}X_t &= AX_{t-1} + Cu_t \\Z_t &= DX_t + v_t\end{aligned}$$

where

$$\begin{aligned}X_t &= x_t, A = \rho, Cu_t = u_t^x \\Z_t &= \begin{bmatrix} y_t \\ \pi_t \end{bmatrix}, D = \begin{bmatrix} -\kappa \frac{\phi\pi - \rho}{-c} \\ \kappa\gamma \frac{1 - \rho}{-c} \end{bmatrix}, v_t = \begin{bmatrix} u_t^y \\ u_t^\pi \end{bmatrix}\end{aligned}$$

# The Random-Walk Metropolis Algorithm

What do we need for the algorithm?

1. Start with an arbitrary value  $\theta_0$
2. Update from  $\theta_j$  to  $\theta_{j+1}$  ( $j = 1, 2, \dots, J$ ) by

2.1 Generate  $\theta^* \sim N(\theta_j, \Sigma)$

2.2 Define

$$\alpha = \min \left( \frac{L(Y | \theta^*)}{L(Y | \theta_j)}, 1 \right) \quad (2)$$

2.3 Take

$$\theta_{j+1} = \left\{ \begin{array}{l} \theta^* \text{ with probability } \alpha \\ \theta_j \text{ otherwise} \end{array} \right\}$$

## The log likelihood function of a state space system

Remember from last time: For a given state space system

$$\begin{aligned} X_t &= AX_{t-1} + C\mathbf{u}_t \\ \underset{(p \times 1)}{Z_t} &= DX_t + \mathbf{v}_t \end{aligned}$$

we can evaluate the log likelihood by computing

$$\mathcal{L}(Z | \Theta) = -.5 \sum_{t=0}^T \left[ p \ln(2\pi) + \ln |\Omega_t| + \tilde{Z}_t' \Omega_t^{-1} \tilde{Z}_t \right]$$

where  $\tilde{Z}_t$  are the innovation from the Kalman filter and  $A, C, D$  and  $\Sigma_{vv}$  are (possibly non-linear) functions of  $\Theta$ .

## The parameter vector $\theta^0$ and the variance of random walk innovations in MCMC $\Sigma$

Parameterize the model according to

$$\begin{aligned}\theta &= \{\rho, \gamma, \kappa, \phi, \sigma_x, \sigma_y, \sigma_\pi, \} \\ &= \{0.9, 2, 0.1, 1.5, 1, 1, 1\}\end{aligned}$$

- ▶ Generate data from true model and  $T = 100$
- ▶ Set starting value  $\theta^{(0)} = \theta$
- ▶ Set covariance matrix of random walk increments in Metropolis Algorithm proportional to absolute values of true parameters

$$\Sigma = \varepsilon \times \text{diag}(\text{abs}(\theta))$$

# The Random-Walk Metropolis Algorithm

We now have all we need:

1. Start with an arbitrary value  $\theta_0$
2. Update from  $\theta_j$  to  $\theta_{j+1}$  ( $j = 1, 2, \dots, J$ ) by

2.1 Generate  $\theta^* \sim N(\theta_j, \Sigma)$

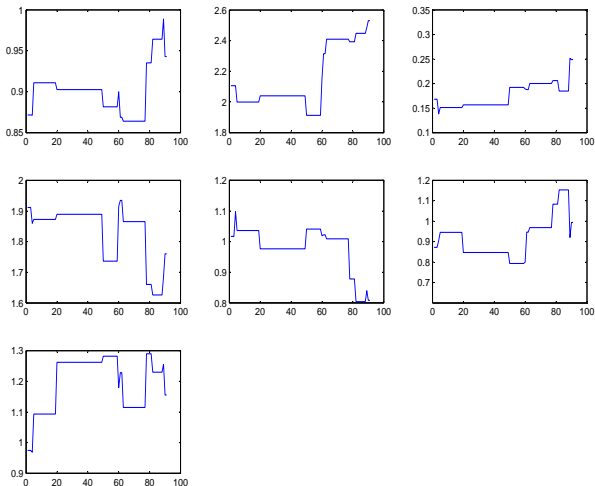
2.2 Define

$$\alpha = \min \left( \frac{L(Y | \theta^*)}{L(Y | \theta_j)}, 1 \right) \quad (3)$$

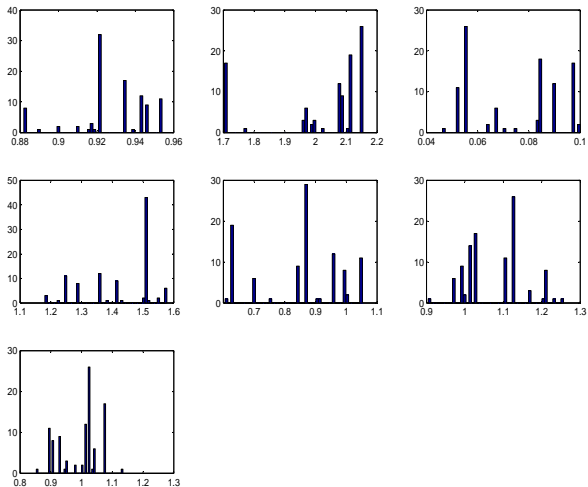
2.3 Take

$$\theta_{j+1} = \left\{ \begin{array}{l} \theta^* \text{ with probability } \alpha \\ \theta_j \text{ otherwise} \end{array} \right\}$$

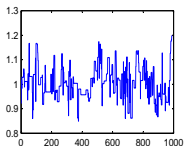
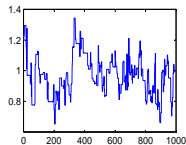
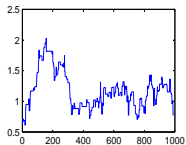
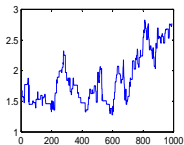
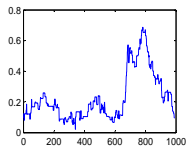
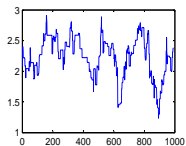
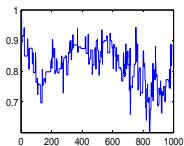
# The MCMC with $J=100$



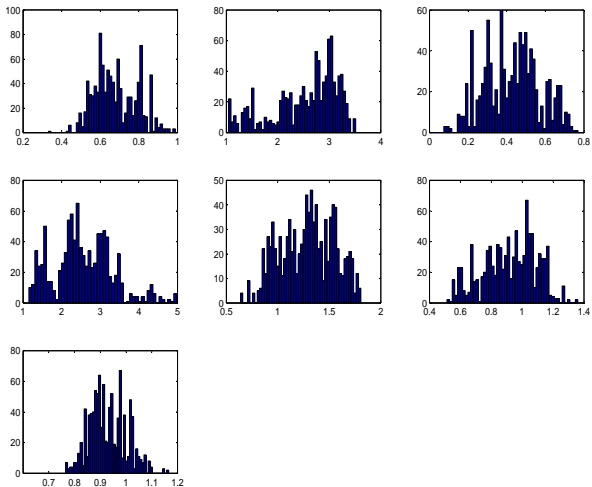
# The MCMC with $J=100$



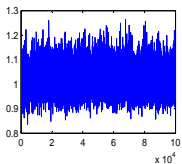
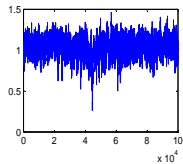
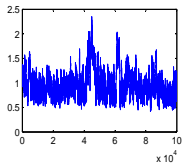
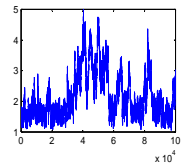
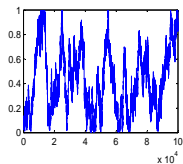
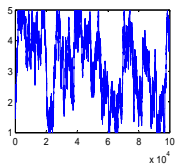
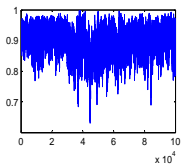
# The MCMC with $J=1000$



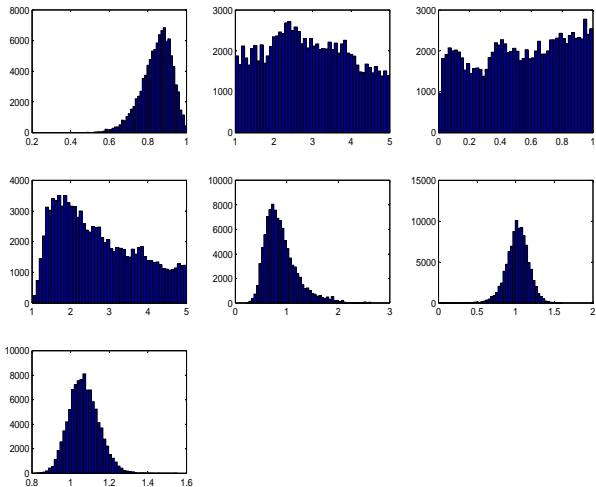
# The MCMC with $J=1000$



# The MCMC with $J=100000$



# The MCMC with $J=100000$

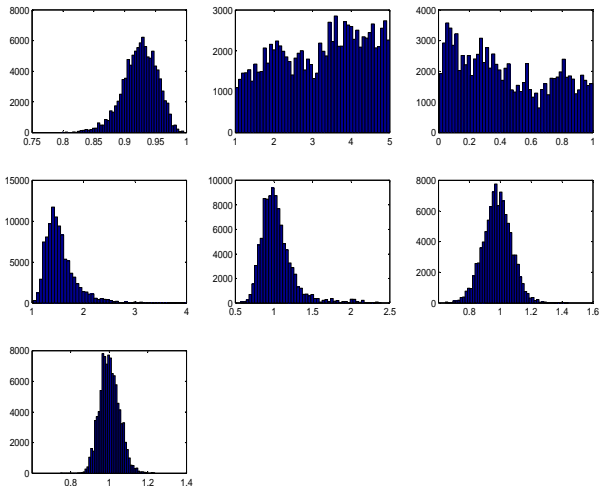


# Identification

Some posterior distribution look 'flat'

- ▶ Can be a short sample issue
- ▶ Can be a problem with mapping between parameters and likelihood

# The MCMC with $T=200m$ and $J=100000$



# Identification

Posterior distribution for  $\gamma$  and  $\kappa$  kappa still look 'flat'  
probably a true identification issue

- ▶ Little can be said about identification a priori

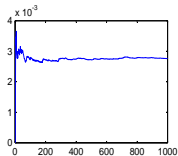
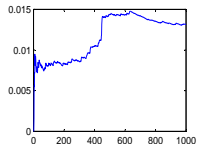
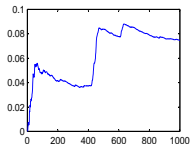
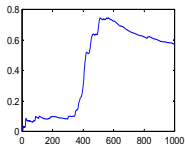
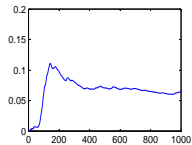
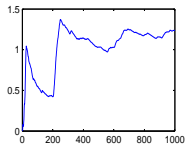
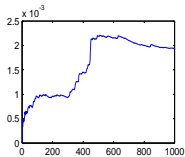
# Convergence

How many draws do we need?

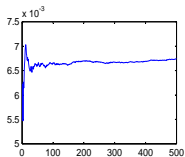
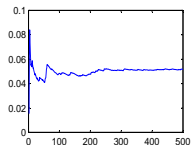
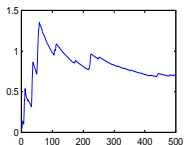
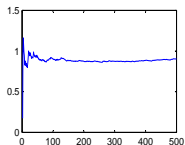
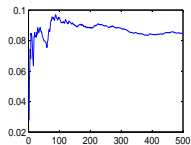
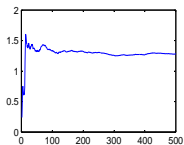
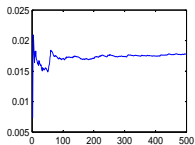
- ▶ Optimal  $J$  increases with number of parameters
- ▶ One informal check is to plot the diagonal of the recursive covariance matrix of the MCMC

$$\frac{1}{j} \sum_{i=0}^j \theta^{(i)} \theta'^{(i)} \text{ for } j = 1, 2, \dots, J$$

# Checking for convergence



# Checking for convergence J=500000



## Next time:

- ▶ Computing probability intervals of functions of  $\theta$
- ▶ Imposing prior information
- ▶ Recap of previous lectures