Term structure in DSGE models: some theoretical and empirical issues (II)

Gianni Amisano (ECB, U Brescia)

Frankfurt, 17 December 2009

Model solution by perturbation (I)

- See Schmitt-Grohè (2005)
- DSGE as a system of nonlinear expectation stochastic difference equation

$$
\begin{aligned}
E_t f(\mathbf{y}_{t+1}, \mathbf{y}_t, \mathbf{x}_{t+1}, \mathbf{x}_t) &= \underset{(n\times 1)}{\mathbf{0}}, n = n_x + n_y \\
\underset{(n_x\times 1)}{\mathbf{x}_t} &= \text{state variables (exog or pred)} \\
\underset{(n_y\times 1)}{\mathbf{y}_t} &= \text{control variables (endogs)}
\end{aligned}
$$

Initial condition $\mathbf{x}_0 + NPG$

Model solution by perturbation (II)

$$
\begin{aligned}
\mathbf{x}_t &= \begin{bmatrix} \mathbf{x}_{1t} \\ \mathbf{x}_{2t} \end{bmatrix} = \begin{bmatrix} \text{predetermined} \\ \text{exogs} \end{bmatrix} \\
n_{x_1} + n_{x_2} &= n_x \\
E_{t-1}\mathbf{x}_{1t} &= \mathbf{x}_{1t} \\
\mathbf{x}_{2t+1} &= \widetilde{H}(\mathbf{x}_{2t}, \sigma) + \sigma \widetilde{\boldsymbol{\eta}} \boldsymbol{\varepsilon}_t, \\
\boldsymbol{\varepsilon}_t &\backsim IID(\mathbf{0}, \mathbf{I}_{n_{x_2}})
\end{aligned}
$$

Model solution by perturbation (III)

Example of this representation: stochastic growth model

$$
\begin{aligned}
y_t &= c_t \\
\mathbf{x}_t &= \begin{bmatrix} k_t \\ a_t \end{bmatrix} \\
n_y &= 1, n_{x_1} = 1, n_{x_2} = 1
\end{aligned}
$$

$\Rightarrow$

$$
E_t f\left(y_{t+1}, y_t, \mathbf{x}_{t+1}, \mathbf{x}_t\right) = \underset{(3\times 1)}{\mathbf{0}} \Rightarrow
$$

$$
E_t \begin{bmatrix} \exp(-\gamma c_t) - \beta \exp(-\gamma c_t) \left\{ \begin{array}{c} \alpha \times \exp[a_{t+1} + \\ (\alpha - 1) k_{t+1}] + (1-\delta) \end{array} \right\} \\ \exp(k_{t+1}) + \exp(c_t) - \exp\left[a_t + \alpha k_t\right] - (1-\delta)\exp(k_t) \\ \exp(a_{t+1}) - \phi \exp(a_t) \end{bmatrix}
$$

$$
a_{t+1} = \phi a_t + \sigma \varepsilon_{t+1}
$$

Model solution by perturbation (IV)

Solution of the model

$$\begin{aligned} \mathbf{x}_{t+1} &= \widehat{H}(\mathbf{x}_t) + \sigma \boldsymbol{\eta} \varepsilon_t, \boldsymbol{\eta} = \begin{bmatrix} 0 \\ \widetilde{\boldsymbol{\eta}} \end{bmatrix} \\ \mathbf{y}_t &= \widehat{G}(\mathbf{x}_t) \end{aligned}$$

Model solution by perturbation (V)

Perturbation method: consider solution as function of state vector x_t and of parameter σ

$$\begin{aligned} \mathbf{x}_{t+1} &= H(\mathbf{x}_t, \sigma) + \sigma \boldsymbol{\eta} \boldsymbol{\varepsilon}_t, \boldsymbol{\eta} = \begin{bmatrix} 0 \\ \widetilde{\boldsymbol{\eta}} \end{bmatrix} \\ \mathbf{y}_t &= G(\mathbf{x}_t, \sigma) \end{aligned}$$

Model solution by perturbation (VI)

approximate with (linear) Taylor expansion around $(\bar{\mathbf{x}}, \overline{\sigma})$

$$
\begin{aligned}
H(\mathbf{x}_t, \sigma) &\approx H(\bar{\mathbf{x}}, \overline{\sigma}) + \mathbf{H}_\sigma(\sigma - \overline{\sigma}) + \mathbf{H}_\mathbf{x}(\mathbf{x}_t - \bar{\mathbf{x}}) \\
G(\mathbf{x}_t, \sigma) &\approx G(\bar{\mathbf{x}}, \overline{\sigma}) + \mathbf{G}_\sigma(\sigma - \overline{\sigma}) + \mathbf{G}_\mathbf{x}(\mathbf{x}_t - \bar{\mathbf{x}})
\end{aligned}
$$

Here unknowns are the derivatives.

Model solution by perturbation (VII)

To find them plug system solution into (**??**) to obtain

$$F(\mathbf{x},\sigma) = E_t f \left\{ \begin{array}{c} G[H(\mathbf{x},\sigma)+\sigma\boldsymbol{\eta}\boldsymbol{\varepsilon}'], \\ G(\mathbf{x},\sigma), H(\mathbf{x},\sigma)+\sigma\boldsymbol{\eta}\boldsymbol{\varepsilon}', \mathbf{x} \end{array} \right\}$$

$F(\mathbf{x},\sigma)$ must be zero for any $\mathbf{x},\sigma$, all its derivatives must be zero too

$$F_{[\mathbf{x}]^k,[\sigma]^j} = [\mathbf{0}]$$

Model solution by perturbation (VIII)

Approximate around non stochastic steady state (NSSS)

$$\mathbf{x}_t = \bar{\mathbf{x}}, \sigma = 0$$

such that

$$\begin{aligned} f\left(\bar{\mathbf{y}}, \bar{\mathbf{y}}, \bar{\mathbf{x}}, \bar{\mathbf{x}}\right) &= 0 \\ \bar{\mathbf{y}} &= G\left(\bar{\mathbf{x}}, 0\right) \\ \bar{\mathbf{x}} &= H\left(\bar{\mathbf{x}}, 0\right) \end{aligned}$$

Model solution by perturbation (IX)

Differentiate F wrt σ

$$
\begin{aligned}
\mathbf{F}_\sigma &= \\
E_t \left\{ \begin{array}{c} \mathbf{f}_{y'} [\mathbf{G}_x (\mathbf{H}_\sigma + \boldsymbol{\eta}\boldsymbol{\varepsilon}') + \mathbf{G}_\sigma] + \\ \mathbf{f}_y \mathbf{G}_\sigma + \mathbf{f}_{x'} (\mathbf{H}_\sigma + \boldsymbol{\eta}\boldsymbol{\varepsilon}') \end{array} \right\} &= \\
\mathbf{f}_{y'} [\mathbf{G}_x \mathbf{H}_\sigma + \mathbf{G}_\sigma] + \mathbf{f}_y \mathbf{G}_\sigma + \mathbf{f}_{x'} \mathbf{H}_\sigma &= \underset{(n \times 1)}{\mathbf{0}} \Rightarrow
\end{aligned}
$$

$$
\left[\begin{array}{cc} \underset{(n \times n_x)}{(\mathbf{f}_{y'} \mathbf{G}_x + \mathbf{f}_{x'})} & \underset{(n \times n_y)}{(\mathbf{f}_{y'} + \mathbf{f}_y)} \end{array} \right] \left[\begin{array}{c} \underset{(n_x \times 1)}{\mathbf{H}_\sigma} \\ \underset{(n_y \times 1)}{\mathbf{G}_\sigma} \end{array} \right] = \left[\begin{array}{c} \mathbf{0} \\ \mathbf{0} \end{array} \right]
$$

Model solution by perturbation (X)

Homogeneous system: will have unique solution

$$\begin{bmatrix} \underset{(n_x \times 1)}{\mathbf{H}_\sigma} \\ \underset{(n_y \times 1)}{\mathbf{G}_\sigma} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix} \tag{1}$$

Model solution by perturbation (XI)

Remarkable result

- approx are simpler than we thought! (good news)
- certainty equivalence principle holds up to FO effects (good news)
- linear approx gives no role to scale of uncertainty (risk premia, welfare analysis) (bad news?)

Model solution by perturbation (XII)

To find $\mathbf{G}_x$ and $\mathbf{H}_x$, differentiate F wrt $\mathbf{x}$

$$
\begin{aligned}
\mathbf{F}_x &= E_t\left\{\mathbf{f}_{y'}\left[\mathbf{G}_x\mathbf{H}_x\right] + \mathbf{f}_y\mathbf{G}_x + \mathbf{f}_{x'}\mathbf{H}_x + \mathbf{f}_x\right\} = \\
&= \mathbf{f}_{y'}\mathbf{G}_x\mathbf{H}_x + \mathbf{f}_y\mathbf{G}_x + \mathbf{f}_{x'}\mathbf{H}_x + \mathbf{f}_x = \underset{(n\times 1)}{\mathbf{0}}
\end{aligned}
$$

$$
\begin{aligned}
\begin{bmatrix} \mathbf{f}_{x'} & \mathbf{f}_{y'} \end{bmatrix}\begin{bmatrix} \mathbf{I}_{n_x} \\ \mathbf{G}_x \end{bmatrix}\mathbf{H}_x &= -\begin{bmatrix} \mathbf{f}_x & \mathbf{f}_y \end{bmatrix}\begin{bmatrix} \mathbf{I}_{n_x} \\ \mathbf{G}_x \end{bmatrix} \Rightarrow \\
\underset{(n\times n)}{\mathbf{A}}\underset{(n\times n_x)}{\mathbf{Z}}\underset{(n_x\times n_x)}{\tilde{}} &= \underset{(n\times n)}{\mathbf{B}}\underset{(n\times n_x)}{\mathbf{Z}}, \text{(GENERALISED EIG)} \\
\mathbf{A} &= \begin{bmatrix} \mathbf{f}_{x'} & \mathbf{f}_{y'} \end{bmatrix}, \mathbf{B} = -\begin{bmatrix} \mathbf{f}_x & \mathbf{f}_y \end{bmatrix}, \\
\mathbf{Z} &= \begin{bmatrix} \mathbf{I}_{n_x} \\ \mathbf{G}_x \end{bmatrix}\mathbf{P}, \mathbf{H}_x\mathbf{P} = \mathbf{P}\tilde{}
\end{aligned}
$$

Model solution by perturbation (XIII)

Non linear system. Can be solved by finding generalised eigenvalues-vectors

$$
\begin{aligned}
\mathbf{A}\left[\underset{(n\times n_x)}{\mathbf{V}_1}\ \underset{(n\times n_y)}{\mathbf{V}_2}\right]\left[\begin{array}{cc}\underset{(n_x\times n_x)}{\mathbf{D}_{11}} & [\mathbf{0}] \\ [\mathbf{0}] & \underset{(n_y\times n_y)}{\mathbf{D}_{22}}\end{array}\right] &= \mathbf{B}[\mathbf{V}_1\mathbf{V}_2] \Rightarrow \\
\mathbf{A}\mathbf{V}_1\mathbf{D}_{11} &= \mathbf{B}\mathbf{V}_1 \Rightarrow \\
\left[\begin{array}{c}\mathbf{I}_{n_x} \\ \mathbf{G}_x\end{array}\right]\mathbf{P} &= \mathbf{V}_1 = \left[\begin{array}{c}\mathbf{V}_{11} \\ \mathbf{V}_{12}\end{array}\right], \\
\tilde{} &= \mathbf{D}_{11} \Rightarrow \\
\mathbf{G}_x &= \mathbf{V}_{12}\mathbf{V}_{11}^{-1}, \\
\mathbf{H}_x &= \mathbf{V}_{11}\mathbf{D}_{11}\mathbf{V}_{11}^{-1}
\end{aligned}
$$

Model solution by perturbation (XIV)

Remarks

- need obtain (numerically) $\mathbf{f}_x, \mathbf{f}_{x'}, \mathbf{f}_y, \mathbf{f}_{y'}$: use Matlab symbolic Toolbox
- can use alternative Schur decomposition
- can use higher order approximations (Fernandex and Villaverde, An and Schorfheide, Amisano and Tristani): second order approximation requires solution of additional linear system.

Model solution by perturbation (XV)

- Higher order: same approach. take second order derivatives
- use first order derivates as inputs
- solve linear system

Model in state space (I)

- Dynamic system

$$
\begin{aligned}
\text{(measurement equation) } \mathbf{y}_t^o &= G(\mathbf{x}_t, \mathbf{v}_t, \boldsymbol{\theta}) \\
\text{(state equation)} \mathbf{x}_t &= H(\mathbf{x}_{t-1}, \mathbf{w}_t, \boldsymbol{\theta})
\end{aligned}
$$

Model in state space (II)

- Linear

$$\begin{aligned} \mathbf{y}_t^o &= \bar{\mathbf{y}} + \mathbf{G}_x \mathbf{x}_t + \mathbf{D}\mathbf{v}_t \\ \mathbf{x}_t &= \mathbf{H}_x \mathbf{x}_{t-1} + \mathbf{B}\mathbf{w}_t \end{aligned}$$

- quadratic

$$\begin{aligned} \mathbf{y}_t^o &= \frac{1}{2}\mathbf{g}_{\sigma\sigma} + \mathbf{G}_x \mathbf{x}_t + \frac{1}{2}\mathbf{G}_{xx} vec(\mathbf{x}_t \mathbf{x}_t^{'}) + \mathbf{D}\mathbf{v}_t \\ \mathbf{x}_t &= \frac{1}{2}\mathbf{h}_{\sigma\sigma} + \mathbf{H}_x \mathbf{x}_{t-1} + \mathbf{H}_{xx} vec(\mathbf{x}_{t-1} \mathbf{x}_{t-1}^{'}) + \mathbf{B}\mathbf{v}_t \end{aligned}$$

Filtering problem

- projection

$$p(\mathbf{x}_{t+1}|\underline{\mathbf{y}}_t^o, \boldsymbol{\theta}) = \int p(\mathbf{x}_{t+1}|\mathbf{x}_t, \boldsymbol{\theta}) p(\mathbf{x}_t|\underline{\mathbf{y}}_t^o, \boldsymbol{\theta}) d\mathbf{x}_t$$

- update

$$\begin{aligned} p(\mathbf{x}_{t+1}|\underline{\mathbf{y}}_{t+1}^o, \boldsymbol{\theta}) &= \frac{p(\mathbf{x}_{t+1}|\underline{\mathbf{y}}_t^o, \boldsymbol{\theta}) p(\mathbf{y}_{t+1}^o|\mathbf{x}_{t+1}, \boldsymbol{\theta})}{p(\mathbf{y}_{t+1}^o|\underline{\mathbf{y}}_t^o, \boldsymbol{\theta})} \\ p(\mathbf{y}_{t+1}^o|\underline{\mathbf{y}}_t^o, \boldsymbol{\theta}) &= \int p(\mathbf{x}_{t+1}|\underline{\mathbf{y}}_t^o, \boldsymbol{\theta}) p(\mathbf{y}_{t+1}^o|\mathbf{x}_{t+1}, \boldsymbol{\theta}) d\mathbf{x}_{t+1} \end{aligned}$$

Integration steps easy only under very special circumstances (KF, Hamilton filter are examples)

Sequential MC methods

Sequential Monte Carlo (SMC) methods.

- Arulampalam *et al*. (2002) , IEEE
- Doucet *et al*. (2001)
- Fernandez-Villaverde and Rubio-Ramirez (2004)
- An and Schorfheide (2005)
- Amisano and Tristani (forthcoming), (2009a), (2009b)

$\Rightarrow$Filtering by simulation.

Particle filter (I)

Simplest way: Particle Filter (PF)
Intuition: compute the likelihood $p(\mathbf{y}^o_{t+1}|\underline{\mathbf{y}}^o_t, \boldsymbol{\theta})$ by:

1. drawing large number of realisations from distribution of $\mathbf{x}_{t+1}$ conditioned on $\underline{\mathbf{y}}^o_t$
2. assigning them weight determined by their "distance " from (compatibility with) $\mathbf{y}^o_{t+1}$.

Particle filter (II)

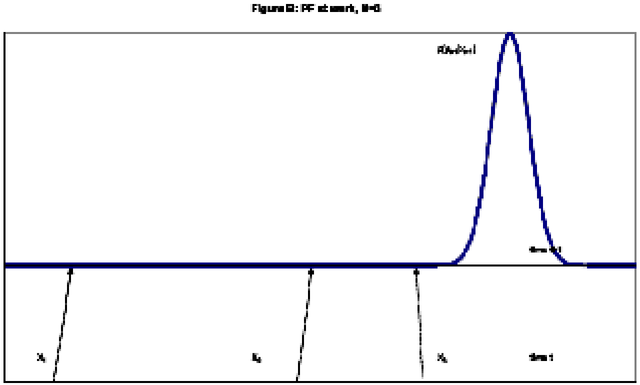
How does PF work?
Spse N draws to approximate $p(\mathbf{x}_t|\underline{\mathbf{y}}_t^o, \boldsymbol{\theta})$ (*swarm of particles*):

$$\left(\mathbf{x}_t^{(i)}, w_t^{(i)}\right), i = 1, 2, ..., N$$

weight $w_t^{(i)}$ in case $\mathbf{x}_t^{(i)}$ drawn from $q(\mathbf{x}_t)$ (Importance sampling):

$$w_t^{(i)} = \frac{p(\mathbf{x}_t^{(i)}|\underline{\mathbf{y}}_t^o, \boldsymbol{\theta})}{q(\mathbf{x}_t^{(i)})}$$

Particle filter (III)



One slide crash course in Bayesian inference

Bayes theorem

$$p(\boldsymbol{\theta}|\mathbf{y}) = \frac{p(\boldsymbol{\theta}) \times p(\mathbf{y}|\boldsymbol{\theta})}{p(\boldsymbol{\theta})}$$

draw from joint posterior using MCMC algorithms

Metropolis-Hastings algorithm

- draw from arbitrary distribution

$$\boldsymbol{\theta}^* \backsim q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(i-1)}, \mathbf{V})$$

- compute

$$\alpha(\boldsymbol{\theta}^{(i)}, \boldsymbol{\theta}^{(i-1)}) = \max\left\{\frac{p(\boldsymbol{\theta}^*)p(\mathbf{y}|\boldsymbol{\theta}^*)}{p(\boldsymbol{\theta}^{(i-1)})p(\mathbf{y}|\boldsymbol{\theta}^{(i-1)})}, 1\right\}$$

- draw $u \backsim U(0,1)$ and set $\boldsymbol{\theta}^{(i)} = \boldsymbol{\theta}^*$ if $u < \alpha(\boldsymbol{\theta}^{(i)}, \boldsymbol{\theta}^{(i-1)})$,
- otherwise set $\boldsymbol{\theta}^{(i)} = \boldsymbol{\theta}^{(i-1)}$