Computing first and second order approximations of DSGE models with DYNARE

Michel Juillard

CEPREMAP
DSGE models

$$E_t \{ f(y_{t+1}, y_t, y_{t-1}, u_t) \} = 0$$

$$u_t = \sigma \epsilon_t$$

$$E(\epsilon_t) = 0$$

$$E(\epsilon_t \epsilon_t') = \Sigma_\epsilon$$

$y$: vector of endogenous variables

$u$: vector of exogenous stochastic shocks

$\sigma$: stochastic scale variable

$\epsilon$: auxiliary random variables
Remarks

- The exogenous shocks may appear only at the current period
- There is no deterministic exogenous variables
- Not all variables are necessarily present with a lead and a lag
- Generalization to leads and lags on more than one period
Solution function

\[ y_t = g(y_{t-1}, u_t, \sigma) \]

Then,

\[ y_{t+1} = g(y_t, u_{t+1}, \sigma) \]
\[ g(g(y_{t-1}, u_t, \sigma), u_{t+1}, \sigma) \]
\[ F(y_{t-1}, u_t, u_{t+1}, \sigma) = f(g(g(y_{t-1}, u_t, \sigma), u_{t+1}, \sigma), g(y_{t-1}, u_t, \sigma), y_{t-1}, u_t) \]

\[ E_t \{ F(y_{t-1}, u_t, u_{t+1}, \sigma) \} = 0 \]
A deterministic steady state, $\bar{y}$, for the model satisfies

$$f(\bar{y}, \bar{y}, \bar{y}, 0) = 0$$

A model can have several steady states, but only one of them will be used for approximation. Furthermore,

$$\bar{y} = g(\bar{y}, 0, 0)$$
First order approximation

Around $\bar{y}$:

$$
E_t \left\{ F^{(1)}(y_{t-1}, u_t, u_{t+1}, \sigma) \right\} =
$$

$$
E_t \left\{ f(y, y, y, 0) + f_y (g_y (g_y \hat{y} + g_u u + g_\sigma \sigma) + g_u u' + g_\sigma \sigma) 
+ f_{y_0} (g_y \hat{y} + g_u u + g_\sigma \sigma) + f_y \hat{y} + f_u u \right\}
$$

$$= 0$$

with $\hat{y} = y_{t-1} - \bar{y}$, $u = u_t$, $u' = u_{t+1}$, $f_y = \frac{\partial f}{\partial y_{t+1}}$, $f_{y_0} = \frac{\partial f}{\partial y_t}$,

$f_y = \frac{\partial f}{\partial y_{t-1}}$, $f_u = \frac{\partial f}{\partial u_t}$, $g_y = \frac{\partial g}{\partial y_{t-1}}$, $g_u = \frac{\partial g}{\partial u_t}$, $g_\sigma = \frac{\partial g}{\partial \sigma}$.
Taking the expectation

\[ E_t \left\{ F^{(1)}(y_{t-1}, u_t, \bar{u}_{t+1}, \sigma) \right\} = \]
\[ f(\bar{y}, \bar{y}, \bar{y}, 0) + f_y^+ (g_y (g_y \hat{y} + g_u u + g_\sigma \sigma) + g_\sigma \sigma) \]
\[ + f_{y_0} (g_y \hat{y} + g_u u + g_\sigma \sigma) + f_{y_-} \hat{y} + f_u u \]
\[ = (f_y^+ g_y g_y + f_{y_0} g_y + f_{y_-}) \hat{y} + (f_y^+ g_y g_u + f_{y_0} g_u + f_u) u \]
\[ + (f_y^+ g_y g_\sigma + f_{y_0} g_\sigma) \sigma \]
\[ = 0 \]
Recovering $g_Y$

$$(f_{y_+}g_yg_y + f_{y_0}g_y + f_{y_-}) \hat{y} = 0$$

Structural state space representation:

$$\begin{bmatrix} 0 & f_{y_+} \\ I & 0 \end{bmatrix} \begin{bmatrix} I \\ g_y \end{bmatrix} g_y \hat{y} = \begin{bmatrix} -f_{y_-} & -f_{y_0} \\ 0 & I \end{bmatrix} \begin{bmatrix} I \\ g_y \end{bmatrix} \hat{y}$$

or

$$\begin{bmatrix} 0 & f_{y_+} \\ I & 0 \end{bmatrix} \begin{bmatrix} y_t - \bar{y} \\ y_{t+1} - \bar{y} \end{bmatrix} = \begin{bmatrix} -f_{y_-} & -f_{y_0} \\ 0 & I \end{bmatrix} \begin{bmatrix} y_{t-1} - \bar{y} \\ y_t - \bar{y} \end{bmatrix}$$
Structural state space representation

\[ D x_{t+1} = E x_t \]

with

\[
\begin{align*}
x_{t+1} &= \begin{bmatrix}
y_t - \bar{y} \\
y_{t+1} - \bar{y}
\end{bmatrix} \\
x_t &= \begin{bmatrix}
y_{t-1} - \bar{y} \\
y_t - \bar{y}
\end{bmatrix}
\end{align*}
\]

- There is an infinity of solutions but we want a unique stable one.
- Problem when \( D \) is singular.
Taking the real generalized Schur decomposition of the pencil $< E, D >$:

$$D = QTZ$$
$$E = QSZ$$

with $T$, upper triangular, $S$ quasi-upper triangular, $Q'Q = I$ and $Z'Z = I$. 
Generalized eigenvalues

\( \lambda_i \) solves

\[ \lambda_i D x_i = E x_i \]

For diagonal blocks on \( S \) of dimension 1 \times 1:

- \( T_{ii} \neq 0 \): \( \lambda_i = \frac{S_{ii}}{T_{ii}} \)
- \( T_{ii} = 0, S_{ii} > 0 \): \( \lambda = +\infty \)
- \( T_{ii} = 0, S_{ii} < 0 \): \( \lambda = -\infty \)
- \( T_{ii} = 0, S_{ii} = 0 \): \( \lambda \in \mathbb{C} \)
Applying the decomposition

\[
D \begin{bmatrix}
I \\
g_y
\end{bmatrix} \hat{g}_y \hat{y} = E \begin{bmatrix}
I \\
g_y
\end{bmatrix} \hat{y}
\]

\[
\begin{bmatrix}
T_{11} & T_{12} \\
0 & T_{22}
\end{bmatrix}
\begin{bmatrix}
Z_{11} & Z_{12} \\
Z_{21} & Z_{22}
\end{bmatrix}
\begin{bmatrix}
I \\
g_y
\end{bmatrix} \hat{y}
\]

= \begin{bmatrix}
S_{11} & S_{12} \\
0 & S_{22}
\end{bmatrix}
\begin{bmatrix}
Z_{11} & Z_{12} \\
Z_{21} & Z_{22}
\end{bmatrix}
\begin{bmatrix}
I \\
g_y
\end{bmatrix} \hat{y}
\]
Selecting stable trajectory

To exclude explosive trajectories, one imposes

\[ Z_{21} + Z_{22}g_y = 0 \]

\[ g_y = -Z_{22}^{-1}Z_{21} \]

A unique stable trajectory exists if \( Z_{22} \) is non-singular: there are as many roots larger than one in modulus as there are forward-looking variables in the model (Blanchard and Kahn condition) and the rank condition is satisfied.
Recovering $g_u$

\[ f_y g_y g_u + f_{y_0} g_u + f_u = 0 \]

\[ g_u = - (f_y g_y + f_{y_0})^{-1} f_u \]
Recovering $g_\sigma$

$$f_{y+}g_y g_\sigma + f_{y_0} g_\sigma = 0$$

$$g_\sigma = 0$$

Yet another manifestation of the certainty equivalence property of first order approximation.
First order approximated decision functions

\[ y_t = \bar{y} + g_y \hat{y} + g_u u \]

\[
E \{ y_t \} = \bar{y} \\
\Sigma_y = g_y \Sigma_y g'_y + \sigma^2 g_u \Sigma_{\epsilon} g'_u
\]

The variance is solved for with an algorithm for Lyapunov equations.
Second order approximation of the model

\[
E_t \left\{ F^{(2)}(y_{t-1}, u_t, u_{t+1}, \sigma) \right\} = \]

\[
E_t \left\{ F^{(1)}(y_{t-1}, u_t, u_{t+1}, \sigma) \right\} + 0.5 \left( F_{yy} \hat{y}_t \hat{y}_t + F_{uu}(u \otimes u) + F_{u'u'}(u' \otimes u') + F_{\sigma\sigma} \sigma^2 \right) \]

\[
+ F_{y'u}(\hat{y} \otimes u) + F_{y'u'}(\hat{y} \otimes u') + F_{y'\sigma} \hat{y} \sigma + F_{uu'}(u \otimes u) + F_{u\sigma} u \sigma + F_{u'u' \sigma} u' \sigma \right\} \]

\[
= E_t \left\{ F^{(1)}(y_{t-1}, u_t, u_{t+1}, \sigma) \right\} \]

\[
+ 0.5 \left( F_{yy} \hat{y}_t \hat{y}_t + F_{uu}(u \otimes u) + F_{u'u'}(\sigma^2 \overline{\Sigma}_\epsilon) + F_{\sigma\sigma} \sigma^2 \right) \]

\[
+ F_{y'u}(\hat{y} \otimes u) + F_{y'\sigma} \hat{y} \sigma + F_{u\sigma} u \sigma \]

\[
= 0 \]
Representing the second order derivatives

The second order derivatives of a vector of multivariate functions is a three dimensional object. We use the following notation

$$\frac{\partial^2 F}{\partial x \partial x} = \begin{bmatrix}
\frac{\partial^2 F_1}{\partial x_1 \partial x_1} & \frac{\partial^2 F_1}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 F_1}{\partial x_2 \partial x_1} & \cdots & \frac{\partial^2 F_1}{\partial x_n \partial x_n} \\
\frac{\partial^2 F_2}{\partial x_1 \partial x_1} & \frac{\partial^2 F_2}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 F_2}{\partial x_2 \partial x_1} & \cdots & \frac{\partial^2 F_2}{\partial x_n \partial x_n} \\
\vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\
\frac{\partial^2 F_m}{\partial x_1 \partial x_1} & \frac{\partial^2 F_m}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 F_m}{\partial x_2 \partial x_1} & \cdots & \frac{\partial^2 F_m}{\partial x_n \partial x_n}
\end{bmatrix}$$
Composition of two functions

Let

\[ y = g(s) \]
\[ f(y) = f(g(s)) \]

then,

\[ \frac{\partial^2 f}{\partial s \partial s} = \frac{\partial f}{\partial y} \frac{\partial^2 g}{\partial s \partial s} + \frac{\partial^2 f}{\partial y \partial y} \left( \frac{\partial g}{\partial s} \otimes \frac{\partial g}{\partial s} \right) \]
Recovering $g_{yy}$

\[ F_{y^- y^-} = f_{y_+} (g_{yy}(g_y \otimes g_y) + g_y g_{yy}) + f_{y_0} g_{yy} + B \]

\[ = 0 \]

where $B$ is a term that doesn’t contain second order derivatives of $g()$.

The equation can be rearranged:

\[ (f_{y_+} g_y + f_{y_0}) g_{yy} + f_{y_+} g_{yy} (g_y \otimes g_y) = -B \]

This is a Sylvester type of equation and must be solved with an appropriate algorithm.
Recovering $g_{yu}$

$$F_{y-u} = f_{y+} (g_{yy}(g_y \otimes g_u) + g_y g_{yu}) + f_{y0} g_{yu} + B$$
$$= 0$$

where $B$ is a term that doesn’t contain second order derivatives of $g()$. This is a standard linear problem:

$$g_{yu} = - (f_{y+} g_y + f_{y0})^{-1} (B + f_{y+} g_{yy} (g_y \otimes g_u))$$
Recovering $g_{uu}$

\[
F_{uu} = f_y + (g_{yy}(g_u \otimes g_u) + g_y g_{uu}) + f_{y_0} g_{uu} + B
\]

\[
= 0
\]

where $B$ is a term that doesn’t contain second order derivatives of $g()$. This is a standard linear problem:

\[
g_{uu} = - (f_{y+} g_y + f_{y_0})^{-1} (B + f_{y+} g_{yy}(g_u \otimes g_u))
\]
Recovering $g_{y\sigma}, g_{u\sigma}$

\[ F_{y\sigma} = f_y g_y g_{y\sigma} + f_{y0} g_{y\sigma} \]
\[ = 0 \]
\[ F_{u\sigma} = f_y g_y g_{u\sigma} + f_{y0} g_{u\sigma} \]
\[ = 0 \]

as $g_\sigma = 0$. Then

\[ g_{y\sigma} = g_{u\sigma} = 0 \]
Recovering $g_{\sigma\sigma}$

\[
F_{\sigma\sigma} + F_{u' u'} \Sigma_{\epsilon} = f_{y_+} (g_{\sigma\sigma} + g_y g_{\sigma\sigma}) + f_{y_0} g_{\sigma\sigma} \\
+ (f_{y_+ y_+} (g_u \otimes g_u) + f_{y_+ g_{uu}}) \tilde{\Sigma}_{\epsilon} \\
= 0
\]

taking into account $g_{\sigma} = 0$.

This is a standard linear problem:

\[
g_{\sigma\sigma} = - (f_{y_+} (I + g_y) + f_{y_0})^{-1} (f_{y_+ y_+} (g_u \otimes g_u) + f_{y_+ g_{uu}}) \tilde{\Sigma}_{\epsilon}
\]
Second order decision functions

\[ y_t = \bar{y} + 0.5g_{\sigma\sigma}\sigma^2 + g_y\hat{y} + g_u u + 0.5 (g_{yy}(\hat{y} \otimes \hat{y}) + g_{uu}(u \otimes u)) + g_{yu}(\hat{y} \otimes u) \]

We can fix \( \sigma = 1 \).

Second order accurate moments:

\[ \Sigma_y = g_y \Sigma_y g_y' + \sigma^2 g_u \Sigma \epsilon g_u' \]

\[ E\{y_t\} = (I - g_y)^{-1} \left( \bar{y} + 0.5 \left( g_{\sigma\sigma} + g_{yy} \bar{\Sigma}_y + g_{uu} \bar{\Sigma}_\epsilon \right) \right) \]
Stochastic versus deterministic SS

Deterministic steady state: the point where the agents decide to stay, in the absence of shocks, and ignoring future shocks.

Stochastic steady state: the point where the agents decide to stay, in the absence of shocks, but taking into account the likelihood of future shocks.

It is possible to compute a second order approximation around the stochastic steady state.
Further issues

- Impulse response functions depend of state at time of shocks and history of future shocks.
- For large shocks second order approximation simulation may explode
  - pruning algorithm (Sims)
  - truncate normal distribution (Judd)
DYNARE commands

Commands:

- check;
- shocks; . . . end;
- stoch_simul(options) variable list;

Options:

- order = 1,[2]
- solve_algo = 0,1,[2]
- dr_algo = [0],1
- irf = 0,...,[40],...
- noprint
Optimal Linear Regulator

Consider,

$$\max_{\{u\}_{t=0}^{\infty}} E_0 \sum_{t=0}^{\infty} \beta^t \left( y_t' W_{11} y_t + 2 y_t' W_{12} u_t + u_t' W_{22} u_t \right)$$

s.t.

$$A_+ E_t (y_{t+1}) + A_0 y_t + A_- y_{t-1} + B u_t + C e_t = 0$$

Lagrangian:

$$L = E_1 \sum_{t=1}^{\infty} \beta^{t-1} \left[ y_t' W_{11} y_t + 2 y_t' W_{12} u_t + u_t' W_{22} u_t + \lambda_t' (A_+ E_t (y_{t+1}) + A_0 y_t + A_- y_{t-1} + B u_t + C e_t) \right]$$
First order conditions

\[ \frac{\partial L}{\partial y_1} = 2W_{11}y_1 + 2W_{12}u_t + A'_0\lambda_1 + \beta A' E_1 (\lambda_2) \]
\[ = 0 \]
\[ \frac{\partial L}{\partial y_t} = 2W_{11}y_t + 2W_{12}u_t + \beta^{-1} A'_t \lambda_{t-1} + A'_0\lambda_t + \beta A' E_t (\lambda_{t+1}) \quad t = 2, \ldots \]
\[ = 0 \]
\[ \frac{\partial L}{\partial u_t} = 2W'_{12}y_t + 2W_{22}u_t + B' \lambda_t \quad t = 1, \ldots \]
\[ = 0 \]
\[ \frac{\partial L}{\partial \lambda_t} = A_+ E_t (y_{t+1}) + A_0 y_t + A_- y_{t-1} + B u_t + C e_t \]
\[ = 0 \]

One can write the first equation (for \( t = 1 \)) as the second one (for \( t > 1 \)) if and only if \( \lambda_0 = 0 \).
Augmented model

\[
2W_{11} y_t + 2W_{12} u_t + \beta^{-1} A'_+ \lambda_{t-1} + A'_0 \lambda_t + \beta A'_- E_t (\lambda_{t+1}) = 0 \\
2W'_{12} y_t + 2W_{22} u_t + B' \lambda_t = 0 \\
A_+ E_t (y_{t+1}) + A_0 y_t + A_- y_{t-1} + B u_t + C e_t = 0
\]

for \(y_0\) given and \(\lambda_0 = 0\).
Example: cgolr.mod

```
var y inf r;
varexo e_y e_inf;

parameters delta sigma alpha kappa;

delta = 0.44;
kappa = 0.18;
alpha = 0.48;
sigma = -0.06;

model(linear);
y = delta * y(-1) + (1-delta) * y(+1) + sigma * (r - inf(+1)) + e_y;
inf = alpha * inf(-1) + (1-alpha) * inf(+1) + kappa*y + e_inf;
end;
```
Example: *cgg_olr.mod* (continued)

shocks;
var e_y; stderr 0.63;
var e_inf; stderr 0.4;
end;

olr_inst r;
optim_weights;
y 1;
inf 1;
end;

olr;