Lecture 12: Solving DSGE Models Part II
Numerical Solution Methods

Adam Hal Spencer

The University of Nottingham

Advanced Monetary Economics 2020

Roadmap

Motivation

- Analytical solution methods involve finding optimality conditions for the model, linearising and then guess and verify.
- Nice because this can lead to nice closed-form solutions for impulse responses that we can perform comparative statics on.
- **Limitation:** this only works for very simple models.
- When things get a little more complicated, we need to use a computer.

Numerical Solutions

- When you use a computer, instead of getting solutions for a general set of parameters as we did in the last lecture, you'll get a particular solution for a specific set of parameters.
- This is why we call it a numerical solution.
- E.g. say there are three different types of capital stocks that can be saved by agents in the model.
  - This is too many endogenous state variables to keep track of analytically.
  - We'll often **have to** solve numerically.

Computational Economics

- There's a whole branch of economics that deals with computational methods for models.
- This is an area that Wisconsin is pretty well-known for, so I learned a lot about it in graduate school.
- If we're finding numerical solutions, how do we interpret the numbers that come out of the model simulations?
- You'll typically be asking a quantitative research question rather than a qualitative one.
- E.g. if the fed funds rate increases by 1%, what will the response of output be?
  - Answer would be something like output decreases by x%.

Dynare

- In this lecture, I'm just going to focus on one small and accessible tool that you can use for solving relatively complicated DSGE models.
- Dynare: collection of Matlab libraries that make solving these models easy.
- Requires no programming experience.
- Used by central banks, international organisations and graduate students.
- I used it for my Masters thesis in Australia...

Dynare

ECONOMIC RECORD
THE ECONOMIC SOCIETY OF AUSTRALIA

Original Article

Effectiveness of the Australian Fiscal Stimulus Package: A DSGE Analysis*

Shuyun May Li, Adam Hal Spencer

First published: 12 November 2015 | https://doi.org/10.1111/1475-4932.12224 | Cited by: 2

Roadmap

General Perturbation Methods

- Dynare employs what are called perturbation methods.
- We are looking for policy functions that satisfy the FOCs of our DSGE model.
- FOC equations generally of the form

$$\mathbb{E}_t[f(y, \epsilon, p)] = 0$$

  where y is a vector of endogenous variables, ϵ is a vector of state variables and p is a vector of parameters.
- We seek a policy function of the form $y(\epsilon, p)$.

First and Higher Order Perturbation

- When we log-linearise our model, we employ a first order perturbation.
- Find the coefficients of a linear approximation to the policy function

$$y(\epsilon, p) = y_0(p) + y_1(p)(\epsilon - \bar{\epsilon})$$

- Higher order perturbation involves extra terms

$$y(\epsilon, p) = y_0(p) + y_1(p)(\epsilon - \bar{\epsilon}) + y_2(p)(\epsilon - \bar{\epsilon})^2 + y_3(p)(\epsilon - \bar{\epsilon})^3 + \ldots$$

- Dynare can do second and third order perturbations as well as first.
- Higher order means more moments of the shocks affect the endogenous outcomes.

Roadmap

Timing

- Timing convention is such that variables that are known at time t are dated $t-1$.
- Need to change the notation from what we've been using.
- E.g. in RBC model with capital
  - k_t denoted the capital we came into the period with.
  - k_{t+1} was our choice of control.
- The translation we'd have to make for input into Dynare would be
  - $k(t-1)$ denoted the capital we came into the period with.
  - $k(t)$ was our choice of control.

Model File: Overview

- The input into Dynare is a model file (.mod file).
- Has a very specific syntax that you need to follow.

Model File (1)

```
%List the endogenous variables
var a b c d;

%List the exogenous variables
varexo e f;

%List the parameters
parameters alpha_1 alpha_2 alpha_3 alpha_4;

%Specify the values for the parameters
alpha_1 = 1;
alpha_2 = 1;
alpha_3 = 1;
alpha_4 = 1;
```

Model File (2)

```
%Specify the model equations
model;
eq_{1};
eq_{2};
eq_{3};
eq_{4};
eq_{5};
eq_{6};
end;
```

Model File (3)

```
%Specify the initial values
initval;
a = a_initval;
b = b_initval;
c = c_initval;
d = d_initval;
e = e_initval;
f = f_initval;
end;
```

Model File (4)

```
%Give values to the shocks
shocks;
var e = 0.01^2;
var f = 0.01^2;
end;

stoch_simul;
```

Model File: Output

- The output from running this code will be impulse response functions for all the endogenous variables from shocks of size 1% to each of the e and f exogenous variables.
- You'll also get details regarding the policy functions (useful if you want to simulate the model).

Roadmap

Model Setup

- Consider a problem of the form

$$\max_{\{c_t, k_{t+1}\}_{t=0}^{\infty}} \mathbb{E}_0 \sum_{t=0} \beta^t \frac{c_t^{1-\sigma}}{1-\sigma}$$

subject to

$$k_{t+1} = a_t k_t^{\alpha} - c_t + (1-\delta) k_t$$
$$\log(a_t) = \rho \log(a_{t-1}) + \epsilon_t$$

Model Optimality Conditions

- Has optimality conditions of the form

$$
\begin{aligned}
c_t^{-\sigma} &= \beta \mathbb{E}_t [c_{t+1}^{-\sigma} (\alpha a_{t+1} k_{t+1}^{\alpha-1} + (1-\delta))] \\
k_{t+1} &= a_t k_t^{\alpha} - c_t + (1-\delta) k_t \\
\log(a_t) &= \rho \log(a_{t-1}) + \epsilon_t, \;\; \epsilon_t \sim N(0, \sigma_\epsilon)
\end{aligned}
$$

Model File (1)

```
%List the endogenous variables
var k a c;

%List the exogenous variables
varexo epsilon;

%List the parameters
parameters sigma beta alpha rho sigma_epsilon delta;

%Specify the values for the parameters
sigma = 1;
beta = 0.95;
alpha = 0.33;
rho = 0.95;
sigma_epsilon = 0.1;
delta = 0.15;
```

Model File (2)

```
%Specify the model equations
model;
c^(-sigma) = beta*c(+1)^(-sigma)* ...
(alpha*a(+1)*k^(alpha-1) + (1-delta));
k = a*k(-1)^(alpha) - c + (1-delta)*k(-1);
ln(a) = rho*ln(a(-1)) + epsilon
end;
```

Model File (3)

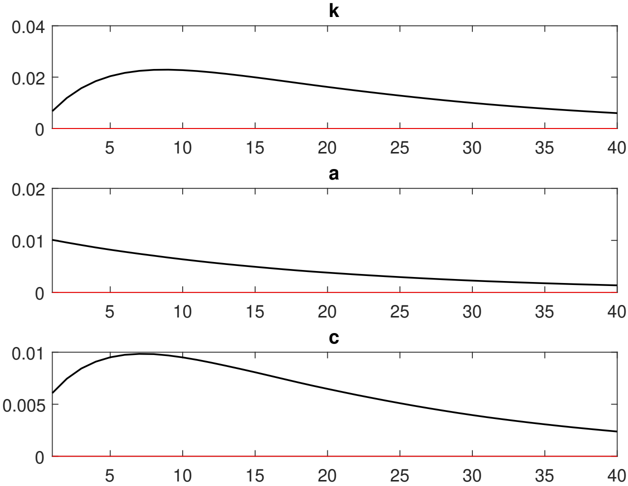
```
%Specify the initial values
initval;
k = (1/alpha*(1/beta - (1-delta)))^(1/(alpha-1)); ...
 %the steady state k
a = 1; %ss a
c = ((1/alpha*(1/beta - (1-delta))))...
^(alpha/(alpha-1)) - ...
delta*(1/alpha*(1/beta - (1-delta)))^(1/(alpha-1));
end;

shocks;
var epsilon = 0.01^2;
end;

stoch_simul;
```

Output: Impulse Response Functions

- Impulse response functions to 1% a_t shock.



Output: Policy Functions

POLICY AND TRANSITION FUNCTIONS

	k	a	c
Constant	2.070785	1.000000	0.960878
(correction)	0.000026	0	-0.000026
k(-1)	0.805219	0	0.247413
a(-1)	0.636330	0.950000	0.571613
epsilon	0.669821	1.000000	0.601698
k(-1),k(-1)	-0.012578	0	-0.020203
a(-1),k(-1)	0.095328	0	0.097172
a(-1),a(-1)	0.009847	-0.023750	-0.040046
epsilon,epsilon	0.363448	0.500000	0.272311
k(-1),epsilon	0.100346	0	0.102286
a(-1),epsilon	0.690552	0.950000	0.517391

- Notice the second order terms in the policy function given that the default is 2^{nd} order perturbation

Roadmap

Estimation

- For the examples we've considered here, we've used fixed parameter values in solving the models.
- A great feature of Dynare is that it can also estimate these parameters for you.
- If you collect macro data on observable variables related to the model, (e.g. inflation, output growth and consumption), you can use Bayesian estimation to get posteriors for the structural parameters.

Roadmap

Takeaways

- When we can't solve a model in its general form, we need to solve it numerically.
- Dynare is a great way to easily get acquainted with solving models with no programming experience.
- Useful to know if you intend to be a professional macroeconomist.