Lecture 2

Solving Real Business Cycle Models

Maarten De Ridder

London School of Economics

Winter Term 2023

This term

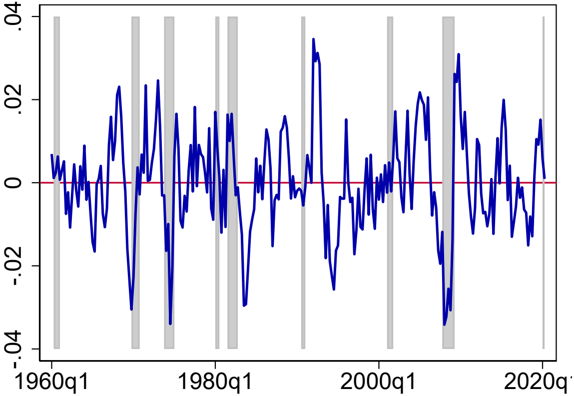
Part I: Shocking theory of the business cycle (weeks 1-6)

- Introduction to business cycles ✓
- Real Business Cycle (RBC) Model $\Leftarrow$
- New Keynesian DSGE Models

Part II: Perspectives on business cycles and steady states (weeks 7-10)

- Persistent effects of recessions
- Aggregate shocks? Firm-heterogeneity and the business cycle
- Interesting steady states: firms, productivity, market power

Last week



Real TFP for the U.S. 1960-2020 - Deviations from HP Trend

Source: Fernald (FRBSF)

Last week

	GDP	Consum.	Invest.	Unempl.	Product.
GDP	1				
Consumption	0.89	1			
Investment	0.88	0.70	1		
Unemployment	-0.88	-0.82	-0.74	1	
Productivity	**0.79**	**0.74**	**0.76**	**-0.56**	1

Correlation matrix for the U.S. 1960-2020 - Deviations from HP Trend

Source: Fred, Fernald (FRBSF)

Baseline RBC Model

Ingredients (slight change from last lecture):

- Representative household:
  Dynamic optimization of consumption, labor, capital
- Representative firm:
  Static opt. of rented capital, labor inputs
- No frictions:
  Investments, labor, prices and wages have no adjustment costs
- Total factor productivity is subject to exogenous shocks

Competitive equilibrium: example

Model with log utility, Cobb Douglas production function:

$$
\begin{aligned}
C_t^{-1} &= \beta \mathrm{E}_t \left[(1 + r_{t+1} - \delta) C_{t+1}^{-1} \right] \\
L_t &= (W_t / C_t)^{\eta} \\
W_t &= Z_t (1-\alpha) K_t^{\alpha} L_t^{-\alpha} \\
r_t &= Z_t \alpha K_t^{\alpha-1} L_t^{1-\alpha} \\
Y_t &= Z_t K_t^{\alpha} L_t^{1-\alpha} \\
Y_t &= C_t + I_t \\
K_{t+1} &= (1-\delta) K_t + I_t \\
Z_t &= Z_{t-1}^{\rho} \exp(\epsilon_t)
\end{aligned}
$$

Solution

Did we solve the model yet? No!

- Model is solved when **endogenous variables** are expressed as function of **exogenous variables**
- In other words: we need to find the **policy functions**
- System of non-linear difference equations $\Rightarrow$ hard to solve
  - This lecture: a new **solution method** to help us out

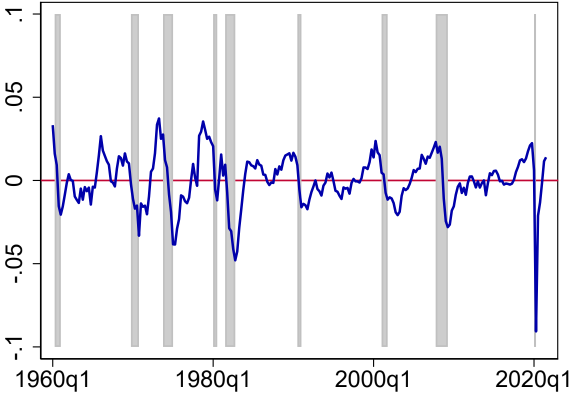
Today

- Tool: local approximation (a.k.a perturbation)
- Solve the model using Matlab plug-in Dynare
- Tool: Log-linearization
- Check for determinacy and existence: Blanchard Kahn conditions

Today

- **Tool: local approximation (a.k.a perturbation)**
- Solve the model using Matlab plug-in Dynare
- Tool: Log-linearization
- Check for determinacy and existence: Blanchard Kahn conditions

What we care about



Real GDP for the U.S. 1960-2021 - Deviations from HP Trend

Source: Fernald (FRBSF)

Local approximation

Non-linear systems of stochastic difference equations $\Rightarrow$ hard to solve

- However: interested in fluctuations around the **steady state**
- Small fluctuations: local approximation a.k.a. perturbation
- Approximate **policy functions** around known point (steady state)

Perturbation

Perturbation is a **local solution method** that relies on:

- Taylor series (1st order, 2nd order, ..) around steady state
- Take derivatives (for Taylor approximations) of the policy functions ..
  - .. without explicitly finding the policy functions **(!)**

Definitions

RBC models can be written in general form:

$$\mathbb{E}_t \left[f(y_{t+1}, y_t, y_{t-1}, u_t) \right] = 0$$

- y: vector of endogenous control and state variables
  - y_{t+1}: subset of variables enters with a lead (consumption, interest)
  - y_{t-1}: subset are predetermined (states: capital, productivity)
- u: vector of exogenous stochastic shocks, $\mathbb{E}(u_t) = 0$

Goal: solve the model / find the policy function g

$$\begin{aligned}
y_t &= g(y_{t-1}, u_t) \\
y_{t+1} &= g(y_t, u_{t+1}) \Rightarrow y_{t+1} = g(g(y_{t-1}, u_t), u_{t+1}) \\
F(y_{t-1}, u_t, u_{t+1}) &= f(g(g(y_{t-1}, u_t), u_{t+1}), g(y_{t-1}, u_t), y_{t-1}, u_t)
\end{aligned}$$

Example: simple RBC model

$$
\begin{aligned}
& \max_{\{C_t, K_t\}_{t=0}^{\infty}} \mathrm{E}_t \sum_{t=0}^{\infty} \beta^t \left(\frac{C_t^{1-\gamma}}{1-\gamma} \right) \\
\text{such that } K_t + C_t &= Z_t K_{t-1}^{\alpha} + (1-\delta) K_{t-1} \\
& K_{-1} \text{ given} \\
\ln Z_t &= \epsilon_t
\end{aligned}
$$

ϵ_t is the disturbance with mean 0, variance 1.

Euler equation:

$$
C_t^{-\gamma} = \beta \mathbb{E}_t \left(\left[\alpha Z_{t+1} K_t^{\alpha-1} + 1 - \delta \right] C_{t+1}^{-\gamma} \right)
$$

Note: timing convention; capital determined at $t-1$ called K_{t-1}

Example: simple RBC model

Inserting the budget constraint in the Euler, we end up with just two equations:

$$0 = \mathbb{E}_t \left(-1 + \beta \left[\alpha Z_{t+1} K_t^{\alpha-1} + 1 - \delta \right] \frac{\left(Z_t K_{t-1}^{\alpha} + (1-\delta) K_{t-1} - K_t \right)^{\gamma}}{\left(Z_{t+1} K_t^{\alpha} + (1-\delta) K_t - K_{t+1} \right)^{\gamma}} \right)$$

$$0 = -\ln Z_t + \epsilon_t$$

The solution has the form

$$K_t = K_t(K_{t-1}, \epsilon_t)$$

Taylor's theorem

Theorem *Let $k \geq 1$ be an integer and let function $g : \mathbb{R} \to \mathbb{R}$ be k times differentiable at $x \in \mathbb{R}$. Then there exists a function $h_k : \mathbb{R} \to \mathbb{R}$ s.t.*

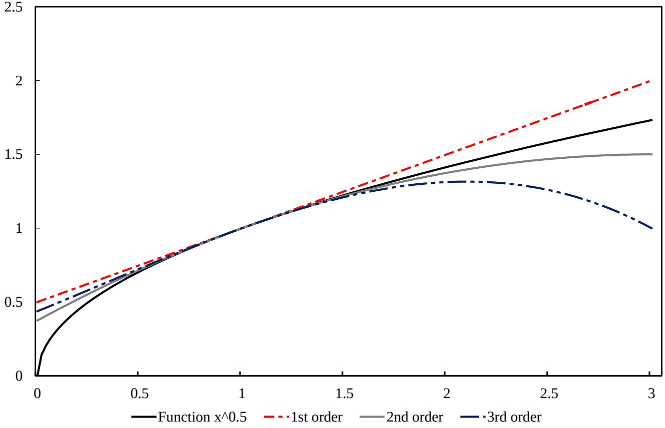
$$g(x_t) = g(x) + g'(x)(x_t - x) + \frac{g''(x)}{2}(x_t - x)^2 + \cdots + \frac{g^{(k)}(x)}{k!}(x_t - x)^k + h_k(x)(x_t - x)^k,$$

and $\lim_{x_t \to x} h_k(x) = 0$.

In our application?

Why do we talk about Taylor **approximations**?

Taylor approximation



Function $y = \sqrt{x}$ and Taylor series approximations around 1

Perturbation - idea

Goal: approximate the policy function $g(.)$ using a Taylor approximation.

$$
\begin{aligned}
g(y_{t-1}, u_t) \quad &\approx \quad g(y,0) + g'_y(y,0)(y_{t-1} - y) + g'_u(y,0)u_t \text{ (1st order)} \\
&+ \quad g''_{yy}(y,0)(y_{t-1} - y)^2/2 + g''_{uu}(y,0)(u_t)^2/2 + g''_{yu}(y,0)(y_{t-1} - y)(u_t) \text{ (2nd)} \\
&+ \quad ..
\end{aligned}
$$

Steps:

1. Find the model's steady state
2. Find the coefficients of the Taylor approximation polynomial

Perturbation - procedure

$$
\begin{aligned}
g(y_{t-1}, u_t) &\approx g(y,0) + g'_y(y,0)(y_{t-1} - y) + g'_u(y,0)u_t \\
&+ g''_{yy}(y,0)(y_{t-1} - y)^2/2 + g''_{uu}(y,0)(u_t)^2/2 + g''_{yu}(y,0)(y_{t-1} - y)(u_t) \\
&+ ..
\end{aligned}
$$

Step 1: Find the model's steady state

Definition of the deterministic steady state:

$$f(y, y, y, 0) = 0$$

which is the fixed point of the policy function:

$$g(y, 0) = y$$

$\Rightarrow$ most cases: use a **numerical solver** (e.g. fsolve)

Perturbation - procedure

$$
\begin{aligned}
g(y_{t-1}, u_t) &\approx g(y,0) + g'_y(y,0)(y_{t-1} - y) + g'_u(y,0)u_t \\
&+ g''_{yy}(y,0)(y_{t-1} - y)^2/2 + g''_{uu}(y,0)(u_t)^2/2 + g''_{yu}(y,0)(y_{t-1} - y)(u_t) \\
&+ ..
\end{aligned}
$$

Step 2: Find the coefficients of the policy function's Taylor approximation

Approach:

- Calculate each coefficient sequentially:
- Obtain steady-state derivatives of $g(y)$ from derivatives of $F(y, 0, 0)$

Linear coefficients

Linear terms appear in the first-order approximation:

$$g(y_{t-1}, u_t) \approx g(y, 0) + g'_y(y, 0)(y_{t-1} - y) + g'_u(y, 0)u_t + ..$$

Use the derivative of the model's system of equations:

$$\begin{aligned}
\mathbb{E}_t\left[F(y_{t-1}, u_t, u_{t+1})\right] &= \mathbb{E}_t\left[f(y_{t+1}, y_t, y_{t-1}, u_t)\right] \\
&= \mathbb{E}_t\left[f(g(g(y_{t-1}, u_t), u_{t+1}), g(y_{t-1}, u_t), y_{t-1}, u_t)\right]
\end{aligned}$$

Derivatives:

$$\mathbb{E}_t\left[F'_y\right] = \mathbb{E}_t\left[\frac{\partial f(y_{t+\mathbf{1}}, y_t, y_{t-\mathbf{1}}, u_{t+\mathbf{1}})}{\partial y_{t+\mathbf{1}}}\frac{\partial g(y_t, u_{t+\mathbf{1}})}{\partial y_t}\frac{\partial g(y_{t-\mathbf{1}}, u_t)}{\partial y_{t-\mathbf{1}}} + \frac{\partial f(.)}{\partial y_t}\frac{\partial g(y_{t-\mathbf{1}}, u_t)}{\partial y_{t-\mathbf{1}}} + \frac{\partial f(.)}{\partial y_{t-\mathbf{1}}}\right]$$

$$\mathbb{E}_t\left[F'_u\right] = \mathbb{E}_t\left[\frac{\partial f(y_{t+\mathbf{1}}, y_t, y_{t-\mathbf{1}}, u_{t+\mathbf{1}})}{\partial y_{t+\mathbf{1}}}\frac{\partial g(y_t, u_{t+\mathbf{1}})}{\partial y_t}\frac{\partial g(y_{t-\mathbf{1}}, u_t)}{\partial u_t} + \frac{\partial f(.)}{\partial y_t}\frac{\partial g(y_{t-\mathbf{1}}, u_t)}{\partial u_t} + \frac{\partial f(.)}{\partial u_t}\right]$$

Linear coefficients

$$g(y_{t-1}, u_t) \approx g(y, 0) + g'_y(y, 0)(y_{t-1} - y) + g'_u(y, 0)u_t + ..$$

Evaluate the derivatives at the steady state:

$$F'_y(y, 0, 0) \quad = \quad \underbrace{\frac{\partial f(y, y, y, 0)}{\partial y_{t+1}}}_{f_1} \underbrace{\frac{\partial g(y, 0)}{\partial y_t} \frac{\partial g(y, 0)}{\partial y_{t-1}}}_{[g'_y(y,0)]^2} + \underbrace{\frac{\partial f(.)}{\partial y_t}}_{f_2} \underbrace{\frac{\partial g(y, 0)}{\partial y_{t-1}}}_{g'_y(y,0)} + \underbrace{\frac{\partial f(.)}{\partial y_{t-1}}}_{f_3} \Bigg|_{y_i = y} = 0$$

$$\Rightarrow \quad f_1 [g'_y(y, 0)]^2 + f_2 g'_y(y, 0) + f_3 = 0$$

This uses these (important) properties:

$$\mathbb{E}_t F(y_{t-1}, u_t, u_{t+1}) = 0 \;\forall\; y_{t-1}, u_t, u_{t+1}$$

which yields:

$$\mathbb{E}_t F'_y(y_{t-1}, u_t, u_{t+1}) = 0 \text{ , in turn yielding } \mathbb{E}_t F'_u(y_{t-1}, u_t, u_{t+1}) = 0$$

Hence: solution to second-order equation gives first derivatives of policy function

Calculating coefficients g_y'

To find g_y' we solve for the roots of second order equation: :

$$f_1[g_y'(y,0)]^2 + f_2 g_y'(y,0) + f_3 = 0$$

In our setup (concavity of utility and production functions, stationary prod.):

- We will get two roots (λ_1, λ_2)
- Where one root 'explosive' ($\mid \lambda \mid > 1$), one is stable ($\mid \lambda \mid < 1$)
- This is the case if the model satisfies **Blanchard-Kahn** conditions (lec. 3)
- We (the software) choose the stable root and set $g_y' = \lambda$

Linear coefficients

$$g(y_{t-1}, u_t) \approx g(y, 0) + g_y'(y, 0)(y_{t-1} - y) + g_u'(y, 0)u_t + ..$$

Now for the derivative with respect to the shock:

$$\begin{aligned}
F_u'(y, 0, 0) &= \underbrace{\frac{\partial f(y, y, y, 0)}{\partial y_{t+1}}}_{f_1} \underbrace{\frac{\partial g(y, 0)}{\partial y_t} \frac{\partial g(y, 0)}{\partial u_t}}_{g_u'(y,0)g_y'(y,0)} + \underbrace{\frac{\partial f(.)}{\partial y_t}}_{f_2} \underbrace{\frac{\partial g(y, 0)}{\partial u_t}}_{g_y'(y,0)} + \underbrace{\frac{\partial f(.)}{\partial u_t}}_{f_4} = 0 \\
&\Rightarrow \quad f_1 g_u'(y, 0) g_y'(y, 0) + f_2 g_u'(y, 0) + f_4 = 0
\end{aligned}$$

Hence: straightforward to solve for $g_u'(y, 0)$ once you have solved for $g_y'(y, 0)$

Impulse response function

From the first-order approximation, we get:

$$y_t = y + g'_y \cdot (y_{t-1} - y) + g'_u u_t$$

Impulse response function: **path** of y_t following a shock at steady state

$$\begin{aligned}
y_t - y &= g'_u u_t \\
y_{t+1} - y &= g'_y (g'_u u_t) \\
y_{t+2} - y &= (g'_y)^2 (g'_u u_t) \\
.. &= .. \\
y_{t+s} - y &= (g'_y)^s (g'_u u_t)
\end{aligned}$$

Second-order approximation

$$g(y_{t-1}, u_t) \approx (..) + g''_{yy}(y_{t-1} - y)^2/2 + g''_{uu}(u_t)^2/2 + g''_{yu}(y_{t-1} - y)(u_t)$$

Second order: take 2nd derivative of model's equations

$$\mathbb{E}_t \left[F''_y \right] = \frac{\partial}{\partial y_{t-1}} \mathbb{E} \left[\frac{\partial f(y_{t+1}, y_t, y_{t-1}, u_{t+1})}{\partial y_{t+1}} \frac{\partial g(y_t, u_{t+1})}{\partial y_t} \frac{\partial g(y_{t-1}, u_t)}{\partial y_{t-1}} + \frac{\partial f(.)}{\partial y_t} \frac{\partial g(y_{t-1}, u_t)}{\partial y_{t-1}} + \frac{\partial f(.)}{\partial y_{t-1}} \right]$$

Evaluated at the steady state (all evaluated at $y_{t+1} = y_t = y_{t-1} = y$):

$$\begin{aligned}
& F''_y(y, 0, 0) \\
= \quad & (g'_y)^2 \left(\frac{\partial^2 f(y,y,y,0)}{\partial y_{t+1} \partial y_{t+1}} (g'_y)^2 + \frac{\partial^2 f(y,y,y,0)}{\partial y_{t+1} \partial y_t} g'_y + \frac{\partial^2 f(y,y,y,0)}{\partial y_{t+1} \partial y_{t-1}} \right) + \frac{\partial f(y,y,y,0)}{\partial y_{t+1}} \left(2 g'_y g''_{yy} \right) \\
+ \quad & g'_y \left(\frac{\partial^2 f(y,y,y,0)}{\partial y_t \partial y_{t+1}} (g'_y)^2 + \frac{\partial^2 f(y,y,y,0)}{\partial y_t \partial y_t} (g'_y) + \frac{\partial^2 f(y,y,y,0)}{\partial y_t \partial y_{t-1}} \right) + \frac{\partial f(y,y,y,0)}{\partial y_t} g''_{yy} \\
+ \quad & g'_y \left(\frac{\partial^2 f(y,y,y,0)}{\partial y_{t-1} \partial y_{t+1}} (g'_y)^2 + \frac{\partial^2 f(y,y,y,0)}{\partial y_{t-1} \partial y_t} (g'_y) + \frac{\partial^2 f(y,y,y,0)}{\partial y_{t-1} \partial y_{t-1}} \right) + \frac{\partial f(y,y,y,0)}{\partial y_{t-1}} = 0
\end{aligned}$$

Note that: g''_{yy} only appears **linearly**; straightforward to solve using g'_y.

Second-order approximation

$$g(y_{t-1}, u_t) \approx (..) + g''_{yy}(y_{t-1} - y)^2/2 + g''_{uu}(u_t)^2/2 + g''_{yu}(y_{t-1} - y)(u_t)$$

Second order: take 2nd derivative of model's equations

$$\mathbb{E}_t\left[F''_y\right] = \frac{\partial}{\partial y_{t-1}}\mathbb{E}\left[\frac{\partial f(y_{t+1}, y_t, y_{t-1}, u_{t+1})}{\partial y_{t+1}}\frac{\partial g(y_t, u_{t+1})}{\partial y_t}\frac{\partial g(y_{t-1}, u_t)}{\partial y_{t-1}} + \frac{\partial f(.)}{\partial y_t}\frac{\partial g(y_{t-1}, u_t)}{\partial y_{t-1}} + \frac{\partial f(.)}{\partial y_{t-1}}\right]$$

Evaluated at the steady state:

$$\begin{aligned}
& F''_y(y, 0, 0) \\
= \quad & (g'_y)^2\left(\frac{\partial^2 f(y,y,y,0)}{\partial y_{t+1}\partial y_{t+1}}(g'_y)^2 + \frac{\partial^2 f(y,y,y,0)}{\partial y_{t+1}\partial y_t}g'_y + \frac{\partial^2 f(y,y,y,0)}{\partial y_{t+1}\partial y_{t-1}}\right) + \frac{\partial f(y,y,y,0)}{\partial y_{t+1}}\left(2g'_y g''_{yy}\right) \\
+ \quad & g'_y\left(\frac{\partial^2 f(y,y,y,0)}{\partial y_t\partial y_{t+1}}(g'_y)^2 + \frac{\partial^2 f(y,y,y,0)}{\partial y_t\partial y_t}(g'_y) + \frac{\partial^2 f(y,y,y,0)}{\partial y_t\partial y_{t-1}}\right) + \frac{\partial f(y,y,y,0)}{\partial y_t}g''_{yy} \\
+ \quad & g'_y\left(\frac{\partial^2 f(y,y,y,0)}{\partial y_{t-1}\partial y_{t+1}}(g'_y)^2 + \frac{\partial^2 f(y,y,y,0)}{\partial y_{t-1}\partial y_t}(g'_y) + \frac{\partial^2 f(y,y,y,0)}{\partial y_{t-1}\partial y_{t-1}}\right) + \frac{\partial f(y,y,y,0)}{\partial y_{t-1}} = 0
\end{aligned}$$

Note that: g''_{yy} only appears **linearly**; straightforward to solve using g'_y.

Why higher order?

First-order approximation: removes **curvature** from the model

- Curvature matters for welfare, risk premia: how uncertainty enters model
- Example: strictly concave contemporaneous utility function U:

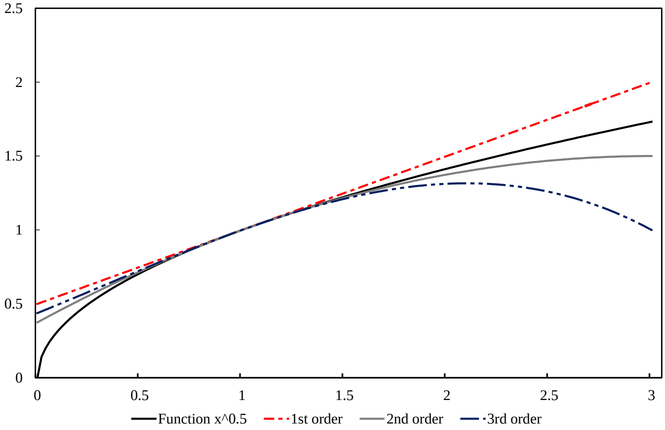
$$\sum_{s=t}^{\infty} \beta^{s-t} \mathbb{E}_t U(C_{t+s}) < \sum_{s=t}^{\infty} \beta^{s-t} U(\mathbb{E}_t C_{t+s})$$

  Linear approximation around C:

$$\sum_{s=t}^{\infty} \beta^{s-t} \mathbb{E}_t \left[U(C) + U_c'(C)(C_{t+s} - C) \right] = \sum_{s=t}^{\infty} \beta^{s-t} \left[U(C) + U_c'(C)(\mathbb{E}_t C_{t+s} - C) \right]$$

- Hence: analysis with first-order approximation is **certainty equivalent**

Why higher order?



Function $y = \sqrt{x}$ and Taylor series approximations around 1

Note: Jensen's inequality

In general, be careful with expectations operator: **Jensen's Inequality**

$$\begin{aligned}
\mathbb{E}f(x) &\leq f(\mathbb{E}[x]) \text{ (concave function)} \\
\mathbb{E}f(x) &\geq f(\mathbb{E}[x]) \text{ (convex function)} \\
\mathbb{E}f(x) &= f(\mathbb{E}[x]) \text{ (linear function)}
\end{aligned}$$

Example: the Euler equation:

$$C_t^{-1} = \beta \mathrm{E}_t \left[(1 + r_{t+1}) C_{t+1}^{-1} \right]$$

$$\mathrm{E}_t \left(C_{t+1} / C_t \right) = \beta \mathrm{E}_t \left[(1 + r_{t+1}) \right] \text{ ✗}$$

More on perturbation

An intuitive and complete description:

Wouter Den Haan's notes (link)

Today

- Tool: local approximation (a.k.a perturbation)
- **Solve the model using Matlab plug-in Dynare**
- Tool: Log-linearization
- Check for determinacy and existence: Blanchard Kahn conditions

Solving an RBC model in practice

Steps:

1. Solve first order conditions, collect constraints to define equilibrium
2. **Calibrate** the model: assign numerical values to parameters
3. Solve for the steady state, either by hand or through **solver**
4. Perform perturbation to approximate policy functions $\Rightarrow$ **Dynare**
5. Plot/analyze **impulse response functions** to shocks

Competitive equilibrium: example

Definition: sequence for the combination of quantities and prices $\{C_t, L_t, L_t^s, A_t, K_t, I_t, Y_t, Z_t\}$, $\{W_t, r_t\}$ such that

- Households solve utility maximization problem
- Firms choose profit-maximizing labor and investment
- Technology constraints:
  capital accumulation, production function, productivity process
- Factor ($L_t^s = L_t$; $A_t = K_t$) and goods markets ($Y_t = C_t + I_t$) clear

Solving an RBC model in practice

Steps:

1. Solve first order conditions, collect constraints, define equilibrium ✓
2. **Calibrate** the model: assign numerical values to parameters
3. Solve for the steady state, either by hand or through **solver**
4. Perform perturbation to approximate policy functions $\Rightarrow$ **Dynare**
5. Plot/analyze **impulse response functions** to shocks

Calibration

Parameter	Description	Value
β	Discount factor	0.99
δ	Capital depreciation rate	0.10
α	Capital share in production	0.33
η	Frisch elasticity of labor supply	0.25
ρ	Persistence of productivity	0.50

Solving an RBC model in practice

Steps:

1. Solve first order conditions, collect constraints, define equilibrium ✓
2. **Calibrate** the model: assign numerical values to parameters ✓
3. Solve for the steady state, either by hand or through **solver**
4. Perform perturbation to approximate policy functions ⇒ **Dynare**
5. Plot/analyze **impulse response functions** to shocks

Our example

$$
\begin{aligned}
C_t^{-1} &= \beta \mathrm{E}_t \left[(1 + r_{t+1} - \delta) C_{t+1}^{-1} \right] \\
L_t &= (W_t / C_t)^{\eta} \\
W_t &= Z_t (1-\alpha) K_t^{\alpha} L_t^{-\alpha} \\
r_t &= Z_t \alpha K_t^{\alpha-1} L_t^{1-\alpha} \\
Y_t &= Z_t K_t^{\alpha} L_t^{1-\alpha} \\
Y_t &= C_t + I_t \\
K_{t+1} &= (1-\delta) K_t + I_t \\
Z_t &= Z_{t-1}^{\rho} \exp\left(\epsilon_t\right)
\end{aligned}
$$

Our example: simplify by taking out prices

Labor market equilibrium:

$$L_t = (W_t/C_t)^{\eta} \quad W_t = Z_t(1-\alpha)K_t^{\alpha}L_t^{-\alpha}$$

$$\Rightarrow C_t L_t^{1/\eta} = (1-\alpha)\underbrace{Y_t/L_t}_{Z_t K_t^{\alpha} L_t^{-\alpha}}$$

Interest rate appears as first order condition and in Euler. Use:

$$r_{t+1} = Z_{t+1}\alpha K_{t+1}^{\alpha-1}L_{t+1}^{1-\alpha} = \alpha\frac{Y_{t+1}}{K_{t+1}}$$

Euler becomes:

$$\frac{1}{C_{t+1}} = \beta \mathrm{E}_t\left[(Z_{t+1}\alpha K_{t+1}^{\alpha-1}L_{t+1}^{1-\alpha}+1-\delta)\frac{1}{C_{t+1}}\right]$$

Competitive equilibrium: new system

$$
\begin{aligned}
C_t^{-1} &= \beta \mathrm{E}_t \left[(\alpha Y_{t+1}/K_{t+1} + 1 - \delta) C_{t+1}^{-1} \right] \\
C_t L_t^{1/\eta} &= (1-\alpha)\, Y_t / L_t \\
Y_t &= Z_t K_t^{\alpha} L_t^{1-\alpha} \\
Y_t &= C_t + I_t \\
K_{t+1} &= (1-\delta) K_t + I_t \\
Z_t &= Z_{t-1}^{\rho} \exp\left(\epsilon_t\right)
\end{aligned}
$$

Example: find steady state

Deterministic steady state: 'ignore time subscripts, expectations'

Note that $\mathrm{E}_t \epsilon_t = 0$

$$\begin{aligned}
\bar{Z} = \bar{Z}^{\rho} \quad &\Rightarrow \quad \bar{Z} = 1 \\
\bar{Y} = \bar{K}^{\alpha} \bar{L}^{1-\alpha} \quad &\Rightarrow \quad \bar{Y}/\bar{K} = \bar{K}^{\alpha-1} \bar{L}^{1-\alpha} \text{ and } \bar{Y}/\bar{L} = \bar{K}^{\alpha} \bar{L}^{-\alpha} \\
\bar{C}^{-1} = \beta \mathrm{E}_t (\alpha \bar{Y}/\bar{K} + 1 - \delta) \bar{C}^{-1} \quad &\Rightarrow \quad \bar{K} = \left(\frac{\alpha}{\beta^{-1} - (1-\delta)} \right)^{\frac{1}{1-\alpha}} \bar{L} \\
\bar{C} \bar{L}^{1/\eta} = (1-\alpha)\, \bar{Y}/\bar{L} \quad &\Rightarrow \quad \bar{C} = (1-\alpha) \bar{L}^{-1/\eta} \bar{K}^{\alpha} \bar{L}^{-\alpha} \\
\bar{K} = (1-\delta)\bar{K} + \bar{I} \quad &\Rightarrow \quad \bar{I} = \delta \bar{K} \\
\bar{Y} \quad &= \quad \bar{C} + \bar{I}
\end{aligned}$$

(no simple closed-form expression here for labor;
though you can get one with some work: insert steady state K/L ratio into wage, insert that into labor supply equation, use resource constraint to substitute-out consumption)

Find values: RBC_steadystate.m on Moodle

```
% set parameters
param_values.beta = 0.99;
param_values.delta = 0.10;
param_values.alpha = 0.33;
param_values.eta = 0.25;
param_values.rho = 0.50;

% solve for steady state
f = @(x) steadystate_solver(x,param_values);
steadystate = fsolve(f,ones(6,1));

% define function
function [y] = steadystate_solver(x,par)

y = zeros(6,1);
Z = x(1);
Y = x(2);
K = x(3);
C = x(4);
I = x(5);
L = x(6);

% steady state function
y(1) = 1 - Z ;
y(2) = K^par.alpha * L^(1-par.alpha) - Y ;
y(3) = (par.alpha/(par.beta^(-1)+par.delta-1))^(1/(1-par.alpha))*L - K ;
y(4) = (1-par.alpha)*L^(-1/par.eta)*K^par.alpha*L^(-par.alpha) - C ;
y(5) = par.delta*K - I ;
y(6) = Y - I - C ;
end
```

Example: find steady state

Variable	Description	Steady state
Z	Productivity	1.00
Y	Output	1.70
K	Capital	5.10
C	Consumption	1.19
I	Investment	0.51
L	Labor	0.99

Solving an RBC model in practice

Steps:

1. Solve first order conditions, collect constraints, define equilibrium ✓
2. **Calibrate** the model: assign numerical values to parameters ✓
3. Solve for the steady state, either by hand or through **solver** ✓
4. Perform perturbation to approximate policy functions $\Rightarrow$ **Dynare**
5. Plot/analyze **impulse response functions** to shocks

Dynare

Dynare is one of the primary tools used to solve DSGE models

- Free software for perturbation solutions and more
  - many options
  - Mainly used with Matlab..
- You MUST know what it is doing
- Once you do, its a very useful tool
- Hundreds of models ready made, including large central banks
  - www.macromodelbase.com

Where/how to get Dynare:

There is also a video (from last year) on Echo360 (via Moodle)

- See the readme file in Moodle under Problem Set 2:
- Download at `www.dynare.org`
- Install the .exe file *and* in Matlab set path to `.../Dynare/Matlab`
- Run one of the example files in Moodle to check if it works.

What does Dynare do?

- Main file type is a $*.mod$ file
- Into this file you specify:
    - Variables of your model
    - Parameters and their values
    - Model equations (linearized or not)
    - Initial values (ideally steady state)
    - Solution method (1st or higher order)
        - Many other options (IRFs, simulations, moments etc.)
- We will go over an example in the Q&A session.
    - Also see the scanned manual from Miao (2014) on Moodle

Policy functions

- In our kind of models, Dynare generates following policy functions

$$k_t = \overline{k} + a_{kk}(k_{t-1} - \overline{k}) + a_{kz}(z_{t-1} - \overline{z}) + a_{k\epsilon}\epsilon_t$$

- i.e. it splits structural shocks into
  - past value and
  - innovation
  - i.e. if $z_t = 1 - \rho + \rho z_{t-1} + \epsilon_t$ then $a_{kz} = \rho a_{k\epsilon}$

Dynare blocks

A Dynare file has several blocks:

1. list of variables
2. list of exogenous shocks
3. list of model parameters and their values
4. model block (optimality conditions)
5. shock properties
6. initial values
7. solution (and other) commands

Dynare example: see Moodle (baseline_RBC.mod)

```
% preamble
var C Y K L I Z;
varexo eps;
parameters beta delta eta alpha rho ;

% assign parameter values
beta = 0.99;
delta = 0.10;
alpha = 0.33;
eta = 0.25;
rho = 0.50;

model;
% equations of the model
C^(-1) = beta*(1+(alpha*Y(+1)/K+(1-delta)))*(C(+1))^(-1);
C = (1-alpha)*Y*L^(-1-1/eta);
Y = Z*K(-1)^alpha*L^(1-alpha);
Y = C + I ;
K = (1-delta)*K(-1) + I;
Z = Z(-1)^rho*exp(eps);
end;
```

Policy Functions (1st order)

Dynare approximates policy functions as we saw previously:

$$y_t \quad = \quad y + g'_y(y_{t-1} - y) + g'_u u_t$$

Note: only pre-determined (state) variables are capital and productivity.

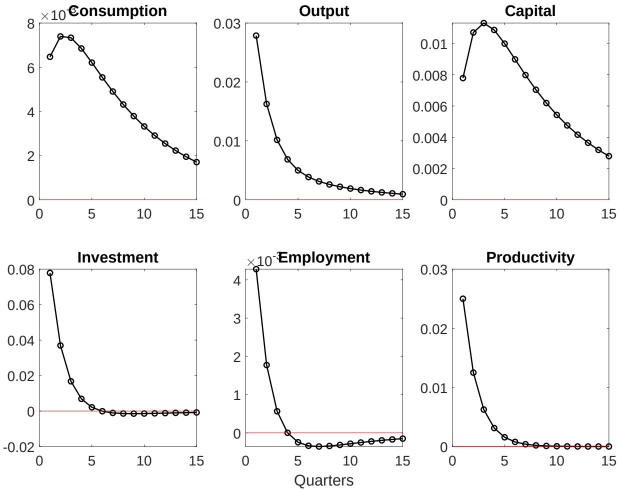
Results:

	C	Y	K	Z	I	L
Constant	1.191883	1.702021	5.101378	1	0.510138	0.9912
K(-1)	0.124799	0.099562	0.874762	0	-0.02524	-0.00916
Z(-1)	0.154249	0.948608	0.794359	0.5	0.794359	0.084832
eps	0.308498	1.897216	1.588717	1	1.588717	0.169664

Solving an RBC model in practice

1. Solve first order conditions, collect constraints, define equilibrium ✓
2. **Calibrate** the model: assign numerical values to parameters ✓
3. Solve for the steady state, either by hand or through **solver** ✓
4. Perform perturbation to approximate policy functions ⇒ **Dynare** ✓
5. Plot/analyze **impulse response functions** to shocks

Impulse responses



Impulse responses to 2.5% productivity shock (in **log** dev. from steady state)

Today

- Tool: local approximation (a.k.a perturbation)
- Solve the model using Matlab plug-in Dynare
- **Tool: Log-linearization**
- Check for determinacy and existence: Blanchard Kahn conditions

Linearization

Perturbation is now a standard way to solve RBC models. Requires:

- Closed-form solution for first order conditions, constraints
- Solution for the steady state (numerically)

Before computer packages, approach was to linearize by hand

- Linearize non-linear system of difference equations (Taylor)
  - System of equations in difference from steady state:

$$\widehat{X_t} = X_t - X \text{ , such that } \widehat{X} = 0$$

- Solve linearized system of equations (e.g. method of undet. coeff.)
- **Identical** results to first-order perturbation

Log-linearization

Now, linearization by hand still useful when:

- Model doesn't have closed form first order condition, steady state.
- Can ease interpretation and back-of-envelope analysis
- Powerful when combined with Dynare
  - Linearize the model by hand, enter into Dynare
  - Let Dynare solve policy functions and impulse response functions

Common: **percent** deviations from the steady state (**log-linearize**)

$$\widehat{x}_t \equiv \frac{X_t - X}{X} \approx \ln\left(\frac{X_t}{X}\right)$$

Log-linearization

Steps:

1. Calculate the point around which to approximate (steady state)

2. Write the system in terms of log deviations from the steady state

$$\begin{aligned} X_t &= e^{\log X_t} \\ &= X\left(e^{x_t - x}\right) \\ &= X\left(e^{\widehat{x_t}}\right) \end{aligned}$$

   where $\widehat{x_t} = x_t - x$, $\log X_t \equiv x_t$

3. Take a Taylor approximation around the steady state

$$\widehat{x_t} = 0 \text{ for all } x_t$$

Log-linearization: example

$$C_t L_t^{1/\eta} = (1-\alpha)\, Y_t / L_t$$

1. Calculate the point around which to approximate (steady state)

$$C L^{1/\eta} = (1-\alpha)\, Y / L$$

2. Write the equation in terms of log deviations from the steady state

$$\begin{aligned} 1 &= (1-\alpha) Y e^{\widehat{y}_t} L^{-1} e^{-\widehat{l}_t} C^{-1} e^{-\widehat{c}_t} L^{-1/\eta} e^{-\eta^{-1}\widehat{l}_t} \\ &= \left(e^{\widehat{y}_t - \widehat{l}_t - \widehat{c}_t - \eta^{-1}\widehat{l}_t} \right) \underbrace{(1-\alpha)\, Y L^{-1} C^{-1} L^{-1/\eta}}_{=1} \end{aligned}$$

3. Take a Taylor approximation around the steady state

$$\begin{aligned} 1 &\approx 1 + \left. \frac{\partial e^{\widehat{y}_t - \widehat{l}_t - \widehat{c}_t - \eta^{-1}\widehat{l}_t}}{\partial \widehat{y}_t} \right|_{\widehat{y}_t = \widehat{c}_t = \widehat{l}_t = 0} (\widehat{y}_t - 0) + \left. \frac{\partial ..}{\partial \widehat{c}_t} \right|_{..} (\widehat{c}_t - 0) + \left. \frac{\partial ..}{\partial \widehat{l}_t} \right|_{.} \left(\widehat{l}_t - 0 \right) \\ 0 &= \widehat{y}_t - \widehat{l}_t - \widehat{c}_t - \eta^{-1}\widehat{l}_t \end{aligned}$$

Log-linearization: another example

$$C_t^{-1} = \beta \mathrm{E}_t \left[(\alpha Y_{t+1}/K_{t+1} + 1 - \delta) C_{t+1}^{-1} \right]$$

1. Calculate the point around which to approximate (steady state)

$$1 = \beta \left(\alpha Y/K + 1 - \delta \right)$$

2. Write the equation in terms of log deviations from the steady state

$$\begin{aligned} 1 &= \beta \mathrm{E}_t \left[(\alpha Y_{t+1}/K_{t+1} + 1 - \delta) C_{t+1}^{-1} \right] C_t \\ &= \beta \mathrm{E}_t \left[(\alpha Y/K e^{\widehat{y}_{t+1} - \widehat{k}_{t+1}} + 1 - \delta) C^{-1} e^{-\widehat{c}_{t+1}} \right] C e^{\widehat{c}_t} \\ &= \beta \mathrm{E}_t \left[(\alpha Y/K e^{\widehat{y}_{t+1} - \widehat{k}_{t+1}} + 1 - \delta) e^{\widehat{c}_t - \widehat{c}_{t+1}} \right] \end{aligned}$$

Log-linearization: another example

$$1 = \beta \mathrm{E}_t \left[(\alpha Y/K e^{\widehat{y}_{t+1} - \widehat{k}_{t+1}} + 1 - \delta) e^{\widehat{c}_t - \widehat{c}_{t+1}} \right]$$

3. Take a Taylor approximation around the steady state

$$\begin{aligned}
1 &\approx 1 + \mathrm{E}_t \beta (\alpha \frac{Y}{K} e^0 + 1 - \delta) e^0 (\widehat{c}_t - \widehat{c}_{t+1}) + \beta (\alpha \frac{Y}{K} e^0) e^0 (\widehat{y}_{t+1} - \widehat{k}_{t+1}) \\
0 &= \mathrm{E}_t \beta \underbrace{\left(\alpha \frac{Y}{K} + 1 - \delta \right)}_{=1/\beta \text{ in steady state}} (\widehat{c}_t - \widehat{c}_{t+1}) + \underbrace{\beta \alpha \frac{Y}{K}}_{1-\beta(1-\delta)} (\widehat{y}_{t+1} - \widehat{k}_{t+1}) \\
&\Rightarrow \mathrm{E}_t \widehat{c}_{t+1} - \widehat{c}_t = (1 - \beta[1-\delta]) \mathrm{E}_t (\widehat{y}_{t+1} - \widehat{k}_{t+1})
\end{aligned}$$

Competitive equilibrium log-linearized

The two examples and the remaining equations (derive yourself!)

$$\begin{aligned}
\mathrm{E}_t \widehat{c}_{t+1} - \widehat{c}_t &= (1-\beta[1-\delta])\mathrm{E}_t(\widehat{y}_{t+1} - \widehat{k}_{t+1}) \\
\widehat{c}_t + \eta^{-1}\widehat{l}_t &= \widehat{y}_t - \widehat{l}_t \\
\widehat{y}_t &= \widehat{z}_t + \alpha\widehat{k}_t + (1-\alpha)\widehat{l}_t \\
\widehat{y}_t &= \widehat{c}_t(1 - I/Y) + \widehat{i}_t I/Y \\
\widehat{k_{t+1}} &= (1-\delta)\widehat{k}_t + \delta\widehat{i}_t \\
\widehat{z}_t &= \rho\widehat{z}_{t-1} + \epsilon_t
\end{aligned}$$

Dynare: see problemset

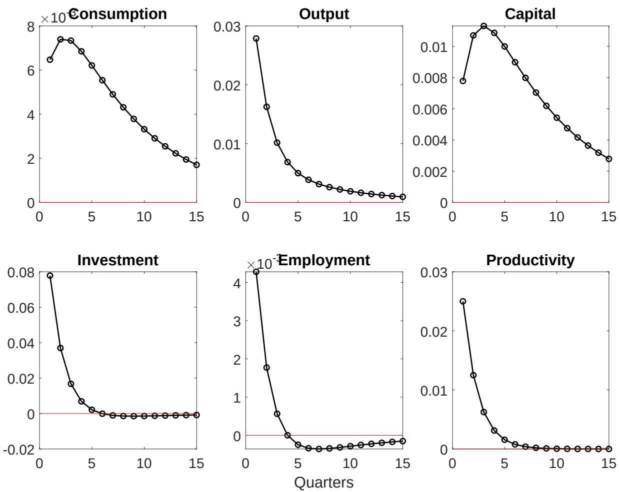
Linearized model:

```
model
c(+1) -c= (1-beta*(1-delta))*(y(+1)-k);
c = y-l*(1+1/eta);
y = z + alpha*k(-1) + (1-alpha)*l ;
y = c*(1-I_ss/Y_ss) + i*(I_ss/Y_ss);
k = (1-delta)*k(-1) + delta*i ;
z = rho*z(-1) + eps;
end;
```

Original model:

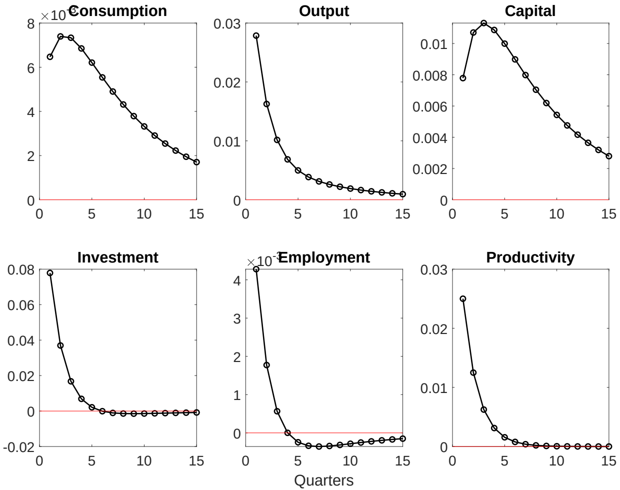
```
model;

C^(-1) = beta*(alpha*Y(+1)/K+1-delta)*(C(+1))^(-1);
C = (1-alpha)*Y*L^(-1-1/eta);
Y = Z*K(-1)^alpha*L^(1-alpha);
Y = C + I ;
K = (1-delta)*K(-1) + I;
Z = Z(-1)^rho*exp(eps);
end;
```

Impulse responses: log-linearized model



Impulse responses to 2.5% productivity shock

Impulse responses: original model



Impulse responses to 2.5% productivity shock (**in log dev. from steady state**)

Today

- Tool: local approximation (a.k.a perturbation)
- Solve the model using Matlab plug-in Dynare
- Tool: Log-linearization
- **Check for determinacy, existence: Blanchard Kahn conditions**

Are we done yet?

So far: found the (approximate) policy functions using software

- Didn't think about existence, uniqueness of a solution
- Our approach 'worked' because of the parameters I picked

Blanchard Kahn (1980) conditions determine

- Whether solutions exist and are unique
- **Next lecture**

What did we do?

- Derived the main tool used in business cycle analysis: perturbation $\checkmark$
- Brief look at Dynare and perturbation in practice (problem set!) $\checkmark$
- Learned how to log-linearize first order conditions, constraints $\checkmark$
- Derived the Blanchard Kahn condition for uniqueness, existence $\checkmark$