

# Real Business Cycle (RBC) Model

Pavel Solís

CIDE

2025

# Introduction

- First RBC model dates back to Kydland and Prescott (1982)
  - Complications to increase persistence to shocks
- Hansen (1986) developed a basic RBC model with essential ingredients
- In RBC world, supply shocks generate economic fluctuations
- Neoclassical growth model as reference for economy's long-term behavior
  - Perfect competition and flexible prices
- Model describes behavior of two types of **representative agents**
  - A large number of identical consumers
  - A large number of identical firms

# Model Structure

- Assumptions about economy
- Agents' optimization problems
  - Households
  - Firms
- Model's equilibrium conditions
- Steady state
- Calibration
- Log-linearization
- Dynare implementation
- Impulse response functions

# Assumptions About Economy

- Closed economy
- Only one (physical) good (as consumption or capital)
- No financial sector (interest rate is a proxy for return on capital)
- No government
- No adjustment costs
- No population growth
- No productivity growth

## Households: Assumptions

- Continuum of mass 1 of identical HHs indexed by  $j \in [0, 1]$ 
  - Per capita variable is equal to aggregate of same variable
  - Dividing aggregate variable by 1 gives the per capita
  - Remove index  $j$  to simplify notation
- HHs are owners of factors of production: Capital and labor
  - Factor markets are perfectly competitive (no rigidities)
- Each HH maximizes an intertemporal utility function
  - HH decides how much to consume, work and save (investment in capital)
- Utility is intertemporally additively separable (no habit formation) into:
  - Consumption (utility) and labor (disutility)

## Households: Preferences

- Optimization problems have well-defined solutions when objective functions are concave and solution spaces are convex sets
- Preferences represented by a utility function:  $U(C_t, H_t)$ 
  - $U_C > 0$  (utility);  $U_H < 0$  (disutility);  $U_{CC} < 0, U_{HH} < 0$  (concave)
  - Common utility functions: logarithm, CRRA, CES
- Future utility in present terms via an intertemporal discount **factor**:  $\beta$ 
  - $\beta = 1/(1 + \theta)$ , where  $\theta$  is subjective **rate** of time preference
- Expectations operator conditional on information at time  $t$ :  $\mathbb{E}_t$
- Each period capital depreciates at rate:  $\delta$

## Firms: Assumptions

- Continuum of mass 1 of identical firms indexed by  $j \in [0, 1]$ 
  - Remove index  $j$  to simplify notation
- Each firm maximizes profits under perfect competition
- Technology represented by a production function:  $F(K_t, H_t)$ 
  - $F_K > 0, F_H > 0$  (increasing);  $F_{KK} < 0, F_{HH} < 0$  (concave)
  - Constant returns to scale:  $F(zK_t, zH_t) = zY_t$
  - Inada conditions:  
 $\lim_{K \rightarrow 0} F_K = \infty, \lim_{K \rightarrow \infty} F_K = 0, \lim_{H \rightarrow 0} F_H = \infty, \lim_{H \rightarrow \infty} F_H = 0$
  - Common production functions:  $AK$ , Cobb-Douglas, CES

# Handout

- Agents' optimization problems
  - Households
  - Firms
- First order conditions
- Model's equilibrium conditions
- Steady state
- Calibration
- Log-linearization
- Dynare implementation



## Variables: Household

- $C_t$ : Units of the good consumed
- $H_t$ : Hours dedicated to labor
- $K_t$ : Units of the good representing the capital stock
- $I_t$ : Units of the good invested as capital
- $T_t$ : Dividends or profits of the firm
- $w_t$ : Wage per hour of labor
- $r_t$ : Return per unit of capital
- $p_t$ : Aggregate price level
- $\beta, \gamma, \varphi, \delta$ : Parameters

## Variables: Firm

- $Y_t$ : Output
- $A_t$ : Productivity
- $V_t$ : Revenue
- $TC_t$ : Total cost
- $MC_t$ : Marginal cost
- $T_t$ : Profits of the firm
- $\alpha, \rho_A, \sigma_A$ : Parameters

## Technical: Unique and Stable Equilibrium

- Model in linearized form can be expressed as

$$A \begin{bmatrix} z_t \\ E_t x_{t+1} \end{bmatrix} = B \begin{bmatrix} z_{t-1} \\ x_t \end{bmatrix} + C \varepsilon_t$$

where

- $z_t$  is a vector of predetermined variables at time  $t$
- $x_t$  is a vector of forward-looking variables
- $\varepsilon_t$  is a vector of shocks
- $A$ ,  $B$  and  $C$  are matrices

## Technical: Unique and Stable Equilibrium

- Blanchard and Kahn (1980): Unique equilibrium and stability condition depends on magnitude of eigenvalues ( $\lambda$ ) of matrix  $B$ 
  - Eigenvalues that satisfy  $|\lambda| > 1$  are called unstable roots
  - Unstable roots arise in dynamic models with forward-looking variables
- If number of unstable roots equals number of forward-looking variables, system has a unique solution and a stable saddle path
  - Otherwise, indeterminacy: multiple solutions or no solution
  - Causes: Calibration errors, model misspecification, ill-behaved expectations

# Matlab

- Programming language developed by MathWorks
  - GUI: Command Window, Editor, Workspace, Current Folder
  - [Link](#) to quick reference guide
- Mainly used in engineering and economics for research and development
- Built around matrices and arrays, so it is easy to operate with them
- Variable names must start with a letter; followed by letters, digits and `_`
- Matlab (and Dynare) are case sensitive:  $x \neq X$ 
  - Beware: Some commands are Dynare-specific
- Write code in `.m` file and run it by typing its name in command window
  - `.mat` files store workspace variables

# Dynare

- Install Dynare from <http://www.dynare.org>
- Every time you start Matlab, add Dynare folder to Matlab path
  - Type: `addpath C:\dynare\x.y\matlab`
  - Set Path → [dynare installation directory] → matlab → Add Folder...
- Write your model in a text file and save it with extension `.mod`
  - Preamble (define variables)
  - Model (and steady state) block(s)
  - Shocks
  - Computations
- In Matlab, make sure current folder is where your `.mod` file is located
- Run it by typing `dynare filename.mod` in command window

## .mod Files

- All code lines must end with a semicolon ;
- Comments can be made with // or % for lines, /\* \*/ for sections of code
- Blocks (model, steady\_state\_model, shocks) closed using end; command
- Equations entered one after another, each one ended with ;
- Model needs to have as many equations as endogenous variables
- Variable and parameter names are case sensitive, must match preamble
- Default timing is t, so  $x_t$  is written as x
  - Use x(+1) for leads and x(-1) for lags (valid for any lead or lag)
- No need to enter expectation operator, Dynare adds it automatically

# Dynare Output

- Impulse response functions (IRFs)
  - What to look for? Scale, on impact effect relative to zero, periods, shape
- All of Dynare output is stored in data structures, most important is `oo_`
- Information about model can be found in structure `M_`
- Linearization can be done manually
  - Equations in model block already linearized
  - In linearized model, all variables have zero steady state
- Dynare can also solve non-linear models (approx. to higher orders)
  - Simulations use extended path methods