Autarky Equilibrium for Home Country

Need to find prices $\{p_1^H, p_2^H\}$, wages, $\{w^H\}$ and allocations$\{c_1^H, c_2^H; l_1^H, l_2^H;\ y_1^H, y_2^H\}$ such that

- **Consumer Optimization** holds

$$c_1^H = \frac{w^H L^H}{p_1^H}\left(\frac{\theta_1}{\theta_1 + \theta_2}\right); c_2^H = \frac{w^H L^H}{p_2^H}\left(\frac{\theta_2}{\theta_1 + \theta_2}\right)$$

- **Firm Optimization** holds for each good, $m = 1{,}2$

$$\text{Optimization for Prices:}\quad p_m^H = a_m^H w^H, \qquad \text{if } l_m > 0$$

$$\text{Production Functions:}\quad y_m^H = \frac{1}{a_m^H} l_m^H$$

- **Market clearing** holds for labor: $l_1^H + l_2^H = L^H$, and for goods: $c_1^H = y_1^H$, $c_2^H = y_2^H$

Solving Equilibrium on the Computer

Preliminary Step: Rewrite Equilibrium Equations to Equal Zero

- Production Functions

$$y_1^H - \frac{1}{a_1^H} l_1^H = 0$$

$$y_2^H - \frac{1}{a_2^H} l_2^H = 0$$

- Prices from Firm Problem

$$p_1^H - a_1^H w^H = 0$$

$$p_2^H - a_2^H w^H = 0$$

Solving Equilibrium on the Computer

Preliminary Step: Rewrite Equilibrium Equations to Equal Zero

- Consumer Problem Solutions

$$c_1^H - \frac{w^H L^H}{p_1^H}\left(\frac{\theta_1}{\theta_1 + \theta_2}\right) = 0$$

$$c_2^H - \frac{w^H L^H}{p_2^H}\left(\frac{\theta_2}{\theta_1 + \theta_2}\right) = 0$$

- Market Clearing for Goods and Labor

$$y_1^H - c_1^H = 0$$

$$y_2^H - c_2^H = 0$$

$$L^H - (l_1^H + l_2^H) = 0$$

Solving Equilibrium on the Computer

Step 1: Set values for the **Exogenous Parameters**

- Preference Paramters: θ_1, θ_2
- Labor Supply: L^H
- Technology Paramters: a_1^H, a_2^H

Solving Equilibrium on the Computer

Step 1: Set values for the **Exogenous Parameters**

- Preference Paramters: θ_1, θ_2
- Labor Supply: L^H
- Technology Paramters: a_1^H, a_2^H

Example Values:

$$\theta_1 = \theta_2 = 1$$

$$L^H = 10;\ \ a_1^H = 1, a_2^H = 2$$

Solving Equilibrium on the Computer

Step 2: Guess values for the **Endogenous Variables**

- Prices $\{p_1^H, p_2^H\}$, wages, $\{w^H\}$ and allocations$\{c_1^H, c_2^H; l_1^H, l_2^H;\ y_1^H, y_2^H\}$
- Guess doesn‘t need to be correct. Can help algorithm convergence if it‘s not horrible.

Solving Equilibrium on the Computer

Step 2: Guess values for the **Endogenous Variables**

- Prices $\{p_1^H, p_2^H\}$, wages, $\{w^H\}$ and allocations$\{c_1^H, c_2^H; l_1^H, l_2^H;\ y_1^H, y_2^H\}$
- Guess doesn‘t need to be correct. Can help algorithm convergence if it‘s not horrible.

Example Guess:

$$\mathbf{Normalize}\ \boldsymbol{w^H = 1}; \text{Guess: } p_1^H = p_2^H = 1$$

$$\text{Guess: } c_1^H = c_2^H = 1; l_1^H = l_2^H = 5; y_1^H = y_2^H = 5$$

IMPORTANT: For **General Equilibrium** models, normalize a price or wage. This means we won‘t update our guess for this price or wage, we will fix it. **Only relative prices matter.**

Solving Equilibrium on the Computer

Step 3: Define a vector **V** to keep track of how far each equilubrium equation is from zero

$$\boldsymbol{V} = \begin{bmatrix} y_1^H - \frac{1}{a_1^H} l_1^H \\ p_1^H - a_1^H w^H \\ c_1^H - \frac{w^H L^H}{p_1^H}\left(\frac{\theta_1}{\theta_1 + \theta_2}\right) \\ \dots \\ y_2^H - c_2^H \\ L^H - (l_1^H + l_2^H) \end{bmatrix}$$

Where each row of **V** corresponds to one of the equilibrium equations (order doesn‘t matter)

Solving Equilibrium on the Computer

Step 3: Define a vector **V** to keep track of how far each equilubrium equation is from zero

$$\boldsymbol{V} = \begin{bmatrix} y_1^H - \frac{1}{a_1^H} l_1^H \\ p_1^H - a_1^H w^H \\ c_1^H - \frac{w^H L^H}{p_1^H}\left(\frac{\theta_1}{\theta_1 + \theta_2}\right) \\ \ldots \\ y_2^H - c_2^H \\ \cancel{L^H - (l_1^H + l_2^H)} \end{bmatrix}$$

Where each row of **V** corresponds to one of the equilibrium equations

- **IMPORTANT**: For **General Equilibrium** models, leave out one of the market clearing conditions. One will hold automatically from **Walras‘ Law**.

Solving Equilibrium on the Computer

Step 4: (Done by Algorithm) Evaluate **V** at your guess, see how far it is from zero

- We define a function that, given any set of guesses, tells us how far $\boldsymbol{V}$ is from zero.

$$\textbf{Example: } V[1] \equiv y_1^H - \frac{1}{a_1^H} l_1^H = 5 - \left(\frac{1}{2}\right) 5 = 5 - 2.5 = 2.5$$

Step 5: (Done by Algorithm) Update guesses, repeat step 4 until all equations equal zero

When the algorithm converges, you found the equilibrium.

- If no convergence, need a better initial guess **or** have a typo in code

Details

Typically can't format your variables as you do here (with subscripts/superscripts)

- Need to come up with a name for each exogenous and endogenous variable

- **Examples:**

$$\text{Theta\_1} = \theta_1$$

$$\text{rho} = \rho$$

$$\text{c1\_H} = c_1^H$$

Details

Typically can’t format your variables as you do here (with subscripts/superscripts)

- Need to come up with a name for each exogenous and endogenous variable

- **Examples:**

$$\text{Theta\_1} = \theta_1$$

$$\text{rho} = \rho$$

$$\text{c1\_H} = c_1^H$$

Every language has different rules for how you can name variables. Typically can‘t include spaces or ^‘s. Often can‘t start a variable name with a number.

- **Not Allowed:** “c 1^H“ or “1theta“

More Details

Feed in guesses as a vector (in my code it's **x** in the function and **xstart** as the guess)

Example: $\boldsymbol{x} = [p_1, p_2, \ldots, y_2^H, l_1^H, l_2^H] = [1{,}1, \ldots, 5{,}5{,}5]$

- Note: **Exogenous parameters** and **Normalized prices** **don't** belong in the guess vector
- Don't need to worry about row vectors/column vectors in this class. Only doing element operations on them.
- Can call **x** and **V** whatever you want (in my code I use **y** instead of **V**), just need to use it consistently in the code