

Algorithm

Nicolas Fernandez-Arias

November 21, 2018

1 Outline

At the highest level, the algorithm for computing the equilibrium of the economy consists of the following iterative scheme.

1. Guess final goods labor g
 - (a) Guess growth rate L^F
 - i. Guess R&D wages $w(q, m)$
 - A. Guess entrant innovation effort $z^E(q, m)$
 - B. Compute V using HACT method by Moll et al.
 - C. Compute W using same method
 - D. Aggregate individual policies from computation of W to compute implied $\tilde{z}^E(q, m)$.
 - E. If not converged, update guess and return to (1aiA)
 - ii. Compute wage $\tilde{w}(q, m) = \bar{w} - \nu W(q, m)$
 - If not converged, update guess and return to (1ai)
 - (b) Compute stationary distribution, aggregate policy functions to implied \tilde{L}^F
 - If not converged, update guess and return to (1a)
2. Using stationary distribution and policy functions, compute implied \tilde{g}
 - If not converged, update guess and return to (1)

2 Finite difference solution of HJBs

2.1 Incumbent

2.1.1 Explicit method

Define Δ_t

2.1.2 Semi-implicit method

In order to make this work, need to be smart about what grid points we use for q . Essentially, need it to be possible to compute V^+ without explicit reference to q , only to the index i_q . This can be achieved by having the points on the grid log-spaced; specifically, $q_{i+1} = q_i * (1 + \lambda)^{1/m}$ for some $m \geq 1$. Set $m > 1$ to make the grid finer.

Discretization:

$$\begin{aligned} \frac{V_{i,j}^{n+1} - V_{i,j}^n}{\Delta_t} + (\rho - g)V_{i,j}^{n+1} &= f_{i,j}^n + \xi_{i,j}^n V_{i,j}^{n+1} + \eta_{i,j} V_{i-1,j}^{n+1} + \gamma_{i,j} V_{i,j+1}^{n+1} \\ &\quad + \sigma_{i,j} (V^{n+1})_{i,j}^+ \end{aligned}$$

with

$$\begin{aligned} f_{i,j}^n &= \pi_{i,j} + x_{i,j}^n (-\bar{w} z_{i,j}^n) + (1 - x_{i,j}^n) (-w_{i,j} z_{i,j}^n) \\ \xi_{i,j}^n &= -gq/\Delta_i^q - (\chi_I z_{i,j}^n + \chi_E z_{i,j}^E) \phi(z_{i,j}^n + z_{i,j}^E) h(q_i) - \nu(z_{i,j}^E + (1 - x_{i,j}^n) z_{i,j}^n) / \Delta_i^m \\ \eta_{i,j}^n &= gq/\Delta_i^q \\ \gamma_{i,j}^n &= \nu(z_{i,j}^E + (1 - x_{i,j}^n) z_{i,j}^n) / \Delta_i^m \\ \sigma_{i,j}^n &= \chi_I z_{i,j}^n \phi(z_{i,j}^n + z_{i,j}^E) h(q_i) \\ \Delta_i^q &= q_i - q_{i-1} = (1 - (1 + \lambda)^{-1/m}) q_i \\ \Delta_i^m &= m_{i+1} - m_i \end{aligned}$$

Further, we compute $(V^{n+1})_{i,j}^+$, extrapolating linearly when necessary:

$$(V^{n+1})_{i,j}^+ = \begin{cases} V_{i+m,0}^{n+1} & i < I - (m - 1) \\ V_{i,0}^{n+1} + \frac{\lambda}{(1+\lambda)^{1/m-1}} (V_{i+1,0}^{n+1} - V_{i,0}^{n+1}) & I - (m - 1) \leq i < I \\ V_{i,0}^{n+1} + \frac{\lambda}{1-(1+\lambda)^{-1/m}} (V_{i,0}^{n+1} - V_{i-1,0}^{n+1}) & i = I \end{cases}$$