Econ 420
Fall, 2022
Li, Hao
UBC

Lecture 10d. Optimal Dynamics

1. Dynamic programming versus optimal control again

- Solution method is different.
  - Dynamic programming uses backward induction and Bellman equation to capture forward looking.
  - Optimal control uses shadow value of state and Hamiltonian function to accomplish the same.

- Solution form is different.
  - Dynamic programming yields value function and the optimal policy as functions of state.
  - Optimal control yields state, co-state, and optimal control as functions of time.

- Solution implementation is different.
  - Dynamic programming is closed loop: complex but reliable.
  - Optimal control is open loop: simple but unreliable.

2. Optimal dynamics in dynamic programming

- Bellman equation:

$$V(x) = \max_{y \in \Gamma(x)} F(x, y) + \beta V(y).$$

  – Solution to Bellman equation yields optimal policy $y = g(x)$ as by product.

- Given any y_0, optimal policy g generates optimal path of state.
  - Define function g^t inductively by

$$g^t(x) = g(g^{t-1}(x))$$

    for each $t = 2, \ldots, \infty$, with $g^1 = g$.

  - Optimal dynamics is

$$y_t = g^t(y_0).$$

    $t = 1, \ldots, \infty$.

- Optimal growth: $\max \sum_{t=0}^{\infty} \beta^t \ln c_t$ subject to $k_{t+1} = k_t^\alpha - c_t$, where $\alpha \in (0,1)$ and $k_0 > 0$ is given.

  – Bellman equation:

$$V(k) = \max_{\tilde{k} \in [0, k^\alpha]} \ln(k^\alpha - \tilde{k}) + \beta V(\tilde{k}).$$

  – Optimal policy is $g(k) = \alpha\beta k^\alpha$.

- Optimal growth dynamics.
  - First order difference equation: $k_{t+1} = \alpha\beta k_t^\alpha$.
  - k_t converges to $k^* = (\alpha\beta)^{1/(1-\alpha)}$.
  - Illustration.

3. Dynamic programming and Euler equation

- Consider $\max_{\{y_t\}_{t=1}^{\infty}} \beta^t F(y_t, y_{t+1})$ subject to $y_{t+1} \in \Gamma(y_t)$ for all $t = 0, 1, \ldots, \infty$, with y_0 given.
  - If $\{y_t^*\}_{t=1}^{\infty}$ is solution, and if y_{t+1} is interior to $\Gamma(y_t)$ for all $t = 0, 1, \ldots, \infty$, then

$$F_y(y_t^*, y_{t+1}^*) + \beta F_x(y_{t+1}^*, y_{t+2}^*) = 0.$$

  - Above second order difference equation is Euler equation in discrete-time model.

- Proof.

  – No profitable arbitrage: each y^*_{t+1} solves

$$\max_{y_{t+1}} F(y^*_t, y_{t+1}) + \beta F(y_{t+1}, y^*_{t+2})$$

  subject to $y_{t+1} \in \Gamma(y^*_t)$ and $y^*_{t+2} \in \Gamma(y_{t+1})$.

- Alternative proof using Bellman equation

$$V(x) = \max_{y \in \Gamma(x)} F(x, y) + \beta V(y).$$

  - First order condition:

$$F_y(x, y) + \beta V'(y) = 0.$$

  - Envelope condition:

$$V'(x) = F_x(x, y).$$

- Solving Euler equation.
  - Solving second order difference equation needs two boundary conditions.
  - Besides $y_0^* = y_0$, also need transversality condition

$$\lim_{T\to\infty} \beta^T F_x(y_T^*, y_{T+1}^*)\ y_T^* = 0.$$

- Euler equation and transversality condition are together sufficient for optimality, if feasible states are non-negative, and immediate function $F(x, y)$ is increasing in x and concave in (x, y).
  - Proof.

- Optimal growth: $\max \sum_{t=0}^{\infty} \beta^t \ln c_t$ subject to $k_{t+1} = k_t^\alpha - c_t$, where $\alpha \in (0,1)$ and $k_0 > 0$ is given.

  - Euler equation:

$$-\frac{1}{k_t^\alpha - k_{t+1}} + \beta \frac{\alpha k_{t+1}^{\alpha-1}}{k_{t+1}^\alpha - k_{t+2}} = 0.$$

  - Verify $k_{t+1} = \alpha\beta k_t^\alpha$ satisfies Euler equation.

  - $\ln(k^\alpha - \tilde{k})$ is increasing in k and concave in $(k, \tilde{k})$.

  - Transversality condition is satisfied.