Recursive Formulation of Repeated Games

Yuya Furusawa

2019/10/30

U-Tokyo, GSE

Repeated Game

Repeated Game

- In repeated games, the same stage-game is repeated T times by same players
  - If $T < \infty$, it is called "finite repeated game"
  - If $T = \infty$, it is called "infinite repeated game"
- We will assume "perfect monitoring" : the outcomes of all past periods are observed by all players
- We will label the stage game G
  - $N = \{1, \cdots, n\}$: the set of players
  - Actions in the stage-game are A_i, and $A = \times_{j \in N} A_j$
  - The stage game payoffs are given by : $u_i : A \to \mathbb{R}$
  - $G = \langle N, (A_i)_{i \in N}, (u_i)_{i \in N} \rangle$

Elements of Repeated Game

- All players discount future payoffs by common discount rate δ
- The action at time t is a_i^t, and its profile is a^t
- We will use $\overrightarrow{a} = \{a^t\}_{t=0}^T$ to denote the sequence of action profiles
- A history of action pairs $h^t = \{a^0, a^1, \ldots, a^t\}$
- A strategy σ_i for a player is a time 0 action and a sequence of functions $\{\sigma_i^t\}_{t=0}^{\infty}$, the tth component of which maps a history h^{t-1} into a time t action for player i
- A strategy profile $\sigma = \times_{i \in N} \sigma_i$
- The discounted payoff is

$$g_i(\overrightarrow{a}) = \sum_{t=0}^{T} \delta^t u_i^t(a^t)$$

Infinitely Repeated Games

- In infinite repeated games, we require $\delta < 1$, and can re-normalize the payoff function:

$$u_i(\overrightarrow{a}) = (1-\delta)\sum_{t=0}^{\infty} \delta^t g_i^t(a^t)$$

  - Suppose a player receives payoff x in each period. Then:

$$x + \delta x + \delta^2 x^2 + \cdots = \frac{1}{1-\delta}x$$

- We will denote v_i as the average payoff and call it "value"

Subgame Equilibria

- A definition of "subgame equilibria" as strategy profiles that satisfy the requirement that given the other player's strategy, each player wants to adhere to his strategy at each date $t \geq 0$ for all possible histories
- A strategy consists of a first-period action and a (continuation) strategy to be followed subsequently
- The continuation strategy profiles have associated present values $\tilde{v}_i$ too
- A subgame equilibrium consists of first period actions for all players chosen in light of players' (rational) expectations about the consequences of those choices for future utilities

Folk Theorem

- Folk Theorem(Abreu, Dutta, and Smith, 1994)
  Any payoff in the convex hull of the stage-game payoff, above minimax, is sustainable for high enough δ in the subgame equilibrium, provided at least two conditions are satisfied:
  - The stage-game has only finitely many pure strategies
  - One of the following two is satisfied
    - $n = 2$
    - $n \geq 3$, and no two players have identical interests

Example - Prisoner's Dilemma

- What is "convex hull of the stage-game payoff"?
- What is "minimax"?
- Let's consider the famous "Prisoner's Dilemma"!
  - $\rightarrow$ blackboard

Computing Values in Equilibrium

Reference

- Abreu, Pearce, and Stacchetti (APS) (1986, 1990)
- Cronshaw and Luenberger (1990)
- Judd, Yeltekin, and Conklin (JYC) (2003)
- Implementation in QuantEcon comes from JYC
- We will focus on very simple example : two-player prisoner's dilemma

Subgame Equilibrium Values

- A characterization of subgame equilibrium values $v_i (i = A, B)$ in terms of a first period action pair a_A, a_B and a pair of subgame perfect continuation values $\tilde{v}_i, \check{v}_i$ that satisfy recursions

$$v_i = (1-\delta) u_i(a_i, a_{-i}) + \delta \tilde{v}_i \geq (1-\delta) u_i(\check{a}_i, a_{-i}) + \delta \check{v}_i \quad (\dagger)$$

  - Here $\tilde{v}_i$ is the continuation value that player i receives if he adheres to a strategy that prescribes first period action a_i this period
  - $\check{v}_i$ is the continuation value in a subgame perfect equilibrium value prescribed if the player deviates from the strategy by choosing $\check{a}_i \neq a_i$.

APS Insight

- Equation (†) for each player i maps pairs of subgame perfect equilibrium continuation values $\tilde{v}_i, \check{v}_i$ into a single value v_i.
- Equation (†) characterizes all subgame perfect equilibrium values.
  - $\tilde{v}_i$ and $\check{v}_i$ each satisfy versions of this equation where they are now on the left side of the equation and another pair of continuation values are on the right side.
  - That $\check{v}_i$ is itself a subgame perfect equilibrium value captures the notion that it is the value associated with a strategy that is a credible threat that deters player i from deviating from the first-period action a_i.
- APS use this insight together with the backward induction inherent in equation (†) to characterize the set of subgame perfect equilibrium values $V \in \mathbb{R}^2$ as the largest fixed point of iterations of a set-to-set mapping

Formal Construction

- Let $\underline{u}_i = \min_{a \in A} u_i(a)$ and $\overline{u}_i = \max_{a \in A} u_i(a)$
- Let $u_i^*(a_{-i}) = \max_{a_i \in A_i} u_i(a_i, a_{-i})$
- The supergame payoffs are contained in the hypercube $\mathcal{W} = \times_{i \in N}[\underline{u}_i, \overline{u}_i]$
- Let $V^p \subset \mathcal{W}$ denote the set of all subgame perfect equilibrium payoffs
- The key to finding V^p is the construction of "self-generating" sets

Self-generation

- The concept of self-generations can be formulated using operator B^p, defined for $W^p \subset \mathcal{W}$:

$$B^p(W^p) = \bigcup_{(a,w) \in A \times W^p} \{(1-\delta)u(a) + \delta w | \forall i (IC_i \geq 0)\}$$

where

$$IC_i = ((1-\delta)u_i(a) + \delta w_i) - ((1-\delta)u_i^*(a_{-i}) + \delta \underline{w}_i)$$

and

$$\underline{w}_i = \inf_{w \in W^p} w_i$$

- A set W^p is self-generating if $W^p \subset B^p(W^p)$
- Cronshaw and Luenberger (1990) show that V^p is self-generating, repeated application of B^p converges to V^p

Public Randomization

- JYC use public randomization to assure that sets are convex
- A convenient property of convex sets (polytopes in particular) is that because we only need to keep track of extreme points, they can be represented easily inside a computer
- Public randomization enables players to coordinate by making their actions depend on a commonly observed public signal
- If W^p is the set of possible values, then $co(W^p)$ is the ones with public randomization
- Then $B^p(co(W^p))$ is the set of equilibrium values and $co(B^p(co(W^p)))$ is the set of ex-ante continuation velues with public randomization

Convergence with Public Randomization

- Let V be the set of equilibrium values with public randomization
- If B is
$$B(W) = co(B^p(co(W))),\ W \in \mathcal{W}$$
then,
    - B is monotone in W, that is, $B(W) \subset B(W')$ if $W \subset W'$
    - V is the largest fixed point of B
    - if $W_0 = \mathcal{W}$ and $W_{i+1} = B(W_i)$, then $V = \bigcap_i W_i$

Approximation of the Operator

- There are two kinds of convex polytope approximation of $B(\cdot)$: inner and outer approximation
  - $\rightarrow$ blackboard
- These approximation preserves the properties of $B(\cdot)$
  - it maps convex sets to convex sets
  - it is monotone
- We will employ numerical algorithm with outer approximation : outer hyperplane algorithm

Outer Hyperplane Algorithm : Step 1

- Initialize elements of the algorithm
  - Subgradients : $h_l \subset \mathbb{R}^2$ and $H = \{h_1, \cdots, h_L\}$
  - Vertices : $z_l \subset \mathbb{R}^2$ and $Z = \{z_1, \cdots, z_L\}$ such that $W = co(Z)$
- Hyperplane levels C are computed by $C = H \cdot Z'$
- There is not a unique way to pick the initial subgradients and hyperplane levels.
- In the note and QuantEcon library, we use a unit circle to pick them, and it often works well

Outer Hyperplane Algorithm : Step 2

- For each $h_l \in H$, solve the following linear programs

$$p(a) = \max_{w} h_l \cdot ((1-\delta)u(a) + \delta w)$$

  subject to

  - $(w_A, w_B) \in W$
  - $(1-\delta)u_A(a) + \delta w_A \geq (1-\delta)u_A^*(a_B) + \delta \underline{w}_A$
  - $(1-\delta)u_B(a) + \delta w_B \geq (1-\delta)u_B^*(a_A) + \delta \underline{w}_B$

  and let $w_l(a)$ be a w value which solves the above linear program

- Find a best action profile and corresponding values

$$a_l^* = \arg\max\{p_l(a) | a \in A\}$$

$$z_l^+ = (1-\delta)u(a_l^*) + \delta w_l(a_l^*)$$

Outer Hyperplane Algorithm : Step 3, 4, and 5

- Step 3
  - Collect set of vertices $Z^+ = \{z_l^+ | l = 1, \cdots, L\}$
  - Construct a new set W^+ through Z^+ with H by outer approximation
    - Compute new hyperplane levels C^+ by $H \times Z'$
- Step 4
  - If $d(W, W^+) > \epsilon$, return to step 2
    - Or check whether $d(C, C^+)$ is greater than ϵ or not
  - Otherwise, proceed
- Step 5
  - Set of vertices is described by Z and define $W^* = \text{co}(Z)$

Changing the Step 2

- $\max_w h_l \cdot ((1-\delta)u(a) + \delta w)$ produces the same optimal solution, w^*, as $\min_w -h_l \cdot ((1-\delta)u(a) + \delta w)$
- Additionally, $\min_w h_l \cdot (u(a) + \delta w)$ produces the same optimal solution, w^*, as $\min_w h_l \cdot w$
- $w \in W$ is equivalent to $H \cdot w \leq C$
- $(1-\delta)u_i(a) + \delta w_i \geq (1-\delta)u_i^*(a_{-i}) + \delta \underline{w}_i$ can be rewritten as $-\delta w_i \leq (1-\delta)(u_i(a) - u_i^*(a_{-i})) - \delta \underline{w}_i$

Changing the Step 2

- Then we can change the problem into the form

$$\min_{x} c^T x \quad \text{subject to} \quad Ax \leq b$$

where

$$x = \begin{bmatrix} w_1 \\ w_2 \end{bmatrix}, \quad c^T = \begin{bmatrix} -h_1 & -h_2 \end{bmatrix},$$

$$A = \begin{bmatrix} h_1^1 & h_2^1 \\ h_1^2 & h_2^2 \\ \dots & \\ h_1^N & h_2^N \\ -\delta & 0 \\ 0 & -\delta \end{bmatrix}, \quad b = \begin{bmatrix} c_1 \\ c_2 \\ \dots \\ c_N \\ (1-\delta)(u_1(a) - u_1^*(a_{-1})) - \delta \underline{w}_1 \\ (1-\delta)(u_2(a) - u_2^*(a_{-2})) - \delta \underline{w}_2 \end{bmatrix}$$