Computing Equilibria of Repeated And Dynamic Games

Şevin Yeltekin

Carnegie Mellon University

ICE 2012

July 2012
Introduction

- Repeated and dynamic games have been used to model dynamic interactions in:
  - Industrial organization,
  - Principal-agent contracts,
  - Social insurance problems,
  - Political economy games,
  - Macroeconomic policy-making.
Introduction

- Repeated and dynamic games have been used to model dynamic interactions in:
  - Industrial organization,
  - Principal-agent contracts,
  - Social insurance problems,
  - Political economy games,
  - Macroeconomic policy-making.
Introduction

- Repeated and dynamic games have been used to model dynamic interactions in:
  - Industrial organization,
  - Principal-agent contracts,
  - Social insurance problems,
  - Political economy games,
  - Macroeconomic policy-making.
Introduction

• Repeated and dynamic games have been used to model dynamic interactions in:
  • Industrial organization,
  • Principal-agent contracts,
  • Social insurance problems,
  • Political economy games,
  • Macroeconomic policy-making.
Introduction

- Repeated and dynamic games have been used to model dynamic interactions in:
  - Industrial organization,
  - Principal-agent contracts,
  - Social insurance problems,
  - Political economy games,
  - Macroeconomic policy-making.
Introduction

• Repeated and dynamic games have been used to model dynamic interactions in:
  • Industrial organization,
  • Principal-agent contracts,
  • Social insurance problems,
  • Political economy games,
  • Macroeconomic policy-making.
Introduction

• These problems are difficult to analyze unless severe simplifying assumptions are made:

  • Equilibrium selection
  • Functional form (cost, technology, preferences)
  • Size of discounting
Introduction

- These problems are difficult to analyze unless severe simplifying assumptions are made:
  - Equilibrium selection
  - Functional form (cost, technology, preferences)
  - Size of discounting
Introduction

• These problems are difficult to analyze unless severe simplifying assumptions are made:
  
  • Equilibrium selection
  
  • Functional form (cost, technology, preferences)
  
  • Size of discounting
Introduction

- These problems are difficult to analyze unless severe simplifying assumptions are made:
  - Equilibrium selection
  - Functional form (cost, technology, preferences)
  - Size of discounting
Goal

- Examine *entire set* of pure-strategy equilibrium values in repeated and dynamic games
- Propose a general algorithm for computation that can handle
  - large state spaces,
  - flexible functional forms,
  - any discounting,
  - flexible informational assumptions.
Approach


- APS show that set of equilibrium payoffs a fixed point of an operator similar to Bellman operator in DP.

- APS method not directly implementable on a computer. Requires approximation of arbitrary sets.

- Our method allows for
  - parsimonious representation of sets/correspondences on a computer
  - preserves monotonicity of underlying operator.
Contributions

• Develop a general algorithm that
  • computes pure-strategy equilibrium value sets of repeated and dynamic games,
  • provides upper and lower bounds for equilibrium values and hence computational error bounds,
  • computes equilibrium strategies.

• Based on: Judd-Yeltekin-Conklin (2003), Sleet and Yeltekin(2003), Yeltekin-Judd (2011)
REPEATED GAMES
Stage Game

- $A_i$ – player $i$’s action space, $i = 1, \ldots, N$
- $A = \times_{i=1}^{N} A_i$ – action profiles
- $\Pi_i(a)$ – Player $i$ payoff, $i = 1, \ldots, N$
- Maximal and minimal payoffs

$$\underline{\Pi}_i \equiv \min_{a \in A} \Pi_i(a), \quad \bar{\Pi}_i \equiv \max_{a \in A} \Pi_i(a)$$
Supergame $G^\infty$

- **Action space:** $A^\infty$

- $h_t$: t-period history: $\{a_s\}_{s=0}^{t-1}$ with $a_s \in A$

- **Set of t-period histories:** $H_t$

- **Preferences:**

  $$w_i(a^\infty) = \frac{1 - \delta}{\delta} E_0 \sum_{t=1}^{\infty} \delta^t \Pi_i(a_t).$$

- **Strategies:** $\{\sigma_{i,t}\}_{t=0}^{\infty}$ with $\sigma_{i,t} : H_t \rightarrow A_i$.

- **Subgame Perfect Equilibrium Payoffs**

  $$V^* \subset \mathcal{W} = \times_{i=1}^{N} [\Pi_i, \bar{\Pi}_i]$$
Example 1: Prisoner’s Dilemma

- Static game: player 1 (2) chooses row (column)

<table>
<thead>
<tr>
<th></th>
<th>Left</th>
<th>Right</th>
</tr>
</thead>
<tbody>
<tr>
<td>Up</td>
<td>4, 4</td>
<td>0, 6</td>
</tr>
<tr>
<td>Down</td>
<td>6, 0</td>
<td>0, 0</td>
</tr>
</tbody>
</table>

- Static Nash equilibrium
  - (Down, Right) with payoff (0, 0)

- Suppose $\delta$ is close to 1
- $G^\infty$ includes (Up, Left) forever with payoff (4, 4)
  - Rational if all believe a deviation causes permanent reversion to (Down, Right)

- This is just one of many equilibria.
Static Equilibrium

- Static game

<table>
<thead>
<tr>
<th></th>
<th>11, c₁₁</th>
<th>12, c₁₂</th>
</tr>
</thead>
<tbody>
<tr>
<td>21, c₂₁</td>
<td>22, c₂₂</td>
<td></td>
</tr>
</tbody>
</table>

$b_{ij}$ ($c_{ij}$) is player 1’s (2’s) return if player 1 (2) plays $i$ ($j$).
Recursive Formulation

- Each SPE payoff vector is supported by
  - profile of actions consistent with Nash today
  - continuation payoffs that are SPE payoffs

- Each stage of subgame perfect equilibrium of $G^\infty$ is a static equilibrium to some one-shot game $A$, augmented by values from $\delta V^*$:

$$
\begin{array}{c|c}
\delta^* b_{11} + \delta u_{11}, \delta^* c_{11} + \delta w_{11} & \delta^* b_{12} + \delta u_{12}, \delta^* c_{12} + \delta w_{12} \\
\delta^* b_{21} + \delta u_{21}, \delta^* c_{21} + \delta w_{21} & \delta^* b_{22} + \delta u_{22}, \delta^* c_{22} + \delta w_{22}
\end{array}
$$

$$\delta^* = 1 - \delta$$
Steps: Computing the Equilibrium Value Set

1. Define an operator that maps today’s equilibrium values to tomorrow’s.

2. Show operator is monotone and equilibrium payoff set is its largest fixed point. [Requires some work. We use Tarski’s FP theorem.]

3. Define approximation for operator and sets that
   - Represent sets parsimoniously on computer
   - Preserve monotonicity of operator

4. Define appropriately chosen initial set, apply operator until convergence.
Step 1: Operator

\( B^* : \mathcal{P} \to \mathcal{P} \).

- Let \( \mathcal{W} \in \mathcal{P} \).

\[
B^*(\mathcal{W}) = \bigcup_{(a,w)} \left\{ (1 - \delta) \Pi(a) + \delta w \right\}
\]

subject to:

\[
w \in \mathcal{W}
\]

and for each \( \forall i \in N, \forall \tilde{a} \in A_i \)

\[
(1 - \delta) \Pi_i(a) + \delta w_i \geq \Pi_i(\tilde{a}, a_{-i}) + \delta w_i
\]

where \( w_i = \min \{ w_i \mid w \in \mathcal{W} \} \).
Step 2: Self-generation

A set $\mathcal{W}$ is self-generating if:

$$\mathcal{W} \subseteq B^*(\mathcal{W})$$

An extension of the arguments in APS establishes the following:

- Any self-generating set is contained within $V^*$,
- $V^*$ itself is self-generating.
Step 2: Factorization

\[ b \in B^*(\mathcal{W}) \] if there is an action profile \( a \) and continuation payoff \( w \in \mathcal{W} \), s.t.

- \( b \) is value of playing \( a \) today and receiving continuation value \( w \),
- for each \( i \), player \( i \) will choose to play \( a_i \)
- punishment value drawn from set \( \mathcal{W} \).
Step 2: Properties of $B^*$

- Monotonicity: $B^*$ is monotone in the set inclusion ordering:
  \[
  \text{If } W_1 \subseteq W_2, \text{ then } B^*(W_1) \subseteq B^*(W_2)
  \]

- Compactness: $B^*$ preserves compactness.

- Implications:
  1) $V^*$ is the maximal fixed point of the mapping $B^*$;
  2) $V^*$ can be obtained by repeatedly applying $B^*$ to any set that contains $V^*$. 
Step 3: Approximation

- $V^*$ is not necessarily a convex set
  - We need to approximate both $V^*$ and the correspondence $B^*(W)$
  - As a first step, use public randomization to convexify the equilibrium value set.
Step 3: Public randomization

- Public lottery with support contained in $\mathcal{W}$.
- Public lottery specifies continuation values for the next period
  - Lottery determines Nash equilibrium for next period.
  - Strategies now condition on histories of actions and lottery outcomes.
- Modified operator:

$$B(W) = B(co(\mathcal{W})) = co(B^*(co(\mathcal{W}))),$$

where $W = co(\mathcal{W})$
- $V$ equilibrium value set of supergame with public randomization.
- $B$ is monotone and $V$ is the largest fixed point of $B$. 
Step B: Approximations

- Modified operator $B$ preserves monotonicity and compactness.

- Produces a sequence of convex sets that converge to equilibrium.

- Two approximations:
  - outer approximation
  - inner approximation
Piecewise-Linear **Inner** Approximation

- Suppose we have $M$ points $Z = \{(x_1, y_1), \ldots, (x_M, y_M)\}$ on the boundary of a convex set $W$.

- The convex hull of $Z$, $co(Z)$, is contained in $W$ and has a piecewise linear boundary.

- Since $co(Z) \subseteq W$, we will call $co(Z)$ the inner approximation to $W$ generated by $Z$. 
Suppose we have \( n \) points \( Z = \{(x_1, y_1), \ldots, (x_n, y_n)\} \) on the boundary of a convex set \( W \).

The convex hull of \( Z \), \( \text{co}(Z) \), is contained in \( W \) and has a piecewise linear boundary, as illustrated by the polygon in Figure 1.

Since \( \text{co}(Z) \subseteq W \), we will call \( \text{co}(Z) \) the inner approximation to \( W \) generated by \( Z \).

**Inner approximations**
Piecewise-Linear **Outer** Approximation

- Suppose we have
  - $M$ points $Z = \{(x_1, y_1), ..., (x_M, y_M)\}$ on the boundary of $W$, and
  - corresponding set of subgradients, $R = \{(s_1, t_1), ..., (s_M, t_M)\}$;
- Therefore,
  - the plane $s_i x + t_i y = s_i x_i + t_i y_i$ is tangent to $W$ at $(x_i, y_i)$, and
  - the vector $(s_i, t_i)$ with base at $(x_i, y_i)$ points away from $W$. 

Computing Equilibria of Repeated And Dynamic Games
Outer approximation

Suppose we have $n$ points $Z = \{(x_1, y_1), \ldots, (x_n, y_n)\}$ on the boundary of $W$, and the corresponding set of subgradients, $\mathcal{R} = \{(s_1, t_1), \ldots, (s_n, t_n)\}$; therefore, the plane $s_i x + t_i y = s_i x_i + t_i y_i$ is tangent to $W$ at $(x_i, y_i)$, and the vector $(s_i, t_i)$ with a point away from $W$.

A convex set and supporting hyperplanes
Key Properties of Approximations

**Definition**

Let $B^I(W)$ be an inner approximation of $B(W)$ and $B^O(W)$ be an outer approximation of $B(W)$; that is $B^I(W) \subseteq B(W) \subseteq B^O(W)$.

**Lemma**

Next, for any $B^I(W)$ and $B^O(W)$, (i) $W \subseteq W'$ implies $B^I(W) \subseteq B^I(W')$, and (ii) $W \subseteq W'$ implies $B^O(W) \subseteq B^O(W')$. 
Step 4: Initial Guesses and Convergence

Proposition

Suppose $B^O(\cdot)$ is an outer monotone approximation of $B(\cdot)$. Then the maximal fixed point of $B^O$ contains $V$. More precisely, if $W \supseteq B^O(W) \supseteq V$, then $B^O(W) \supseteq B^O(B^O(W)) \supseteq \cdots \supseteq V$.

Lemma

$W \supseteq B^O(W) \supseteq V$. 
Proposition

Suppose $B^I(\cdot)$ is an inner monotone approximation of $B(\cdot)$. Then the maximal fixed point of $B^I$ is contained in $V$. More precisely, if $W \subseteq B^I(W) \subseteq V$, then $B^I(W) \subseteq B^I(B^I(W)) \subseteq \cdots \subseteq V$.

Lemma

$W \subseteq B^I(W) \subseteq V$. 
Fixed Point

These results together with the monotonicity of the $B$ operator, implies the following theorem.

**Theorem**

Let $V$ be the equilibrium value set. Then (i) if $W_0 \supseteq V$ then $B^O(W_0) \supseteq B^O(B^O(W_0)) \supseteq \cdots \supseteq V$, and (ii) if $W_0 \subset B^I(W_0)$ then $B^I(W_0) \subset B^I(B^I(W_0)) \subset \cdots \subset V$. Furthermore, any fixed point of $B^I$ is contained in the maximal fixed point of $B$, which in turn is contained in the maximal fixed point of $B^O$. 
Monotone Inner Hyperplane Approximation

Input: Points \( Z = \{z_1, \ldots, z_M\} \) such that \( W = \text{co}(Z) \).

Step 1 Find extremal points of \( B(W) \):

For each search subgradient \( h_\ell \in H, \ell = 1, \ldots, L \).

(1) For each \( a \in A \), solve the linear program

\[
\begin{align*}
    c_\ell(a) & = \max_w h_\ell \cdot [(1 - \delta)\Pi(a) + \delta w] \\
    \text{(i)} & \quad w \in W \\
    \text{(ii)} & \quad (1 - \delta)\Pi^i(a) + \delta w_i \geq (1 - \delta)\Pi^*_i(a_{-i}) + \delta w_i, \quad i = 1, \ldots, N
\end{align*}
\]

Let \( w_\ell(a) \) be a \( w \) value which solves (1).
Monotone Inner Hyperplane Approximation cont’d

(2) Find best action profile $a \in A$ and continuation value:

$$
a^*_\ell = \arg \max \{ c_\ell(a) | a \in A \}
$$

$$
z^{+}_\ell = (1 - \delta)\Pi(a^*_\ell) + \delta w_\ell(a^*_\ell)
$$

Step 2 Collect set of vertices $Z^+ = \{ z^{+}_\ell | \ell = 1, ..., L \}$, and define $W^+ = co(Z^+)$. 
The Outer Approximation, Hyperplane Algorithm

Outer approximation: Same as inner approximation except record normals and continuation values $z^+_\ell$
Outer vs. Inner Approximations

- Any equilibrium is in the inner approximation
  - Can construct an equilibrium strategy from $V$.
  - There exist multiple such strategies
The Outer Approximation, Hyperplane Algorithm

- No point outside of outer approximation can be an equilibrium
  - Can demonstrate certain equilibrium payoffs and actions are not possible
  - E.g., can prove that joint profit maximization is not possible
Error Bounds

- Difference between inner and outer approximations is approximation error.
- Computations actually constitute a proof that something is in or out of equilibrium payoff set - not just an approximation.
- Difference is small in many examples.
ErrorBounds
Convergence: Repeated Prisoner’s Dilemma
Hyperplanes: Repeated Prisoner’s Dilemma
Example 2: Repeated Cournot Duopoly

- Firm $i$ sales: $q_i$
- Firm $i$ unit cost: $c_i = 0.6$
- Demand: $p = \max\{6 - q_1 - q_2, 0\}$
- Profit: $\Pi_i(q_1, q_2) = q_i(p - c_i)$
- Nash Eqm. Payoff of Stage Game: (3.24, 3.24)
- Shared Monopoly Payoff: (3.64, 3.64)
Repeated Cournot

![Graph showing repeated Cournot game with payoffs and strategies marked.](image-url)
Example 2: Repeated Cournot Duopoly

- Set of eqm payoffs quite large.
- Shared monopoly profits (+ and ⋆) are achievable (for \( \delta = 0.8 \))
- When costs are positive, threats far worse than reversion to Nash.
Strategies: Repeated Cournot
**Strategies: Repeated Cournot**

Actions, promises, and threats on the boundary of $V$, $c = 0.6$

<table>
<thead>
<tr>
<th>$\ell$</th>
<th>$(v_1(\ell), v_2(\ell))$</th>
<th>$(w_1(\ell), w_2(\ell))$</th>
<th>$(q_1, q_2)$</th>
<th>$\Pi(q_1, q_2)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>3.97 3.30</td>
<td>3.75 3.52</td>
<td>1.7 0.9</td>
<td>4.8 2.4</td>
</tr>
<tr>
<td>8</td>
<td>3.71 3.57</td>
<td>3.72 3.55</td>
<td>1.3 1.3</td>
<td>3.6 3.6</td>
</tr>
<tr>
<td>10</td>
<td>3.64 3.64</td>
<td>3.64 3.64</td>
<td>1.3 1.3</td>
<td>3.6 3.6</td>
</tr>
<tr>
<td>27</td>
<td>0.29 6.76</td>
<td>0.36 6.65</td>
<td>0.0 3.0</td>
<td>0.0 7.1</td>
</tr>
<tr>
<td>46</td>
<td>0.00 0.00</td>
<td>0.77 0.77</td>
<td>5.1 5.1</td>
<td>-3.0 -3.0</td>
</tr>
<tr>
<td>60</td>
<td>4.75 0.00</td>
<td>6.71 0.32</td>
<td>5.1 2.1</td>
<td>-3.0 -1.3</td>
</tr>
</tbody>
</table>
Example 2: Repeated Cournot Duopoly

- Unlike APS’s imperfect monitoring example, eqm. paths are not bang-bang.
- Continuation of worst eqm is not worst. Movement towards cooperation?
- Shared Monopoly: Markov and stationary.
- Low profits today for Firm $i$ are supported by higher continuation values.
Next Meeting

- Dynamic Games
- Using algorithm to find endogenous state spaces.
- Extensions to planner+continuum of agents.
- Examples from applications in IO, Macro.