Overview

- Motivation and definitions
- Representation: Properties
- Algorithms

Motivation

- Multi-party games: large number of players.
- Traditional representation: matrix or normal form
  - every player "plays" with all others.
  - payoff matrix for each player grows exponentially with number of players!
- New representation: Graphical games
  - exploits "game structure"
  - limited interaction: each player only "plays" with a "small" subset of all other players.
  - More compact representation

[See accompanying PowerPoint presentation]
Some "Strategic Properties" of Graphical Games

- Problem still non-trivial: the eq. strategy of a player "affects" that of every other player (if G fully connected)
- Let X, Y subset of player. If X, Y disconnected in G, X, Y form independent games
- For every player i, if we "set" (the strategies for) the neighbors of i in G, we get 2 independent subgames:
  1. i by himself; 2. all non-neighbors of i.

The (Conditional) eq. of each subgame are also independent.

- More generally, let
  \[ S = \text{set of players that "separates" the remaining set of players into 2 non-empty subsets} \]
  \[ X, Y \]

  If we "set" the players in S, the resulting subgame (and conditional eq.) for players in X is independent of that for players in Y.

Ex: G - a tree

\[ S = \{ i \} \]

(Dynamic Programming) Alg. exploits these properties.
Consider assigning a NE for root of tree

What do we need?

- "Set" $V = v$; Consider $\bar{U} = \bar{u}$, and ask
  - Is $V = v$ a best response to $\bar{U} = \bar{u}$?
  - $\forall i$, Does there exist an eq. "upstream" in which $U_i$ plays $u_i$ when $V$ is "set" to $v$?
    $$T_{Vu_i}(v, u_i)$$

- If "yes" to all questions, $\exists$ a NE in which $V = v$ and $\bar{U} = \bar{u}$
  Such a $\bar{u}$ is called a witness (to $v$)

Otherwise, keep trying other values for $v$ and $\bar{u}$ until we find one!
[NE existence $\Rightarrow$ there is at least one such setting $(v, \bar{u})$]

- For such $(v, \bar{u})$, let $V = v$ and $\bar{U} = \bar{u}$ in NE.
- Recursively, apply same "procedure" for each parent $U_i$

How do we get $T_{Vu_i}(v, u_i)$?
Applies dynamic programming.
[See accompanying PowerPoint presentation]
Approximation Algorithm

Basic idea:

- **Discretize mixed-strategy space**
  - (uniformly along each "dimension"
    - \( \Rightarrow \) uniform grid)
  
  \( \forall \varepsilon \in (0, 2, 2, 1) \)

  \( \varepsilon \)-grid \( \Rightarrow \) each player has \( \Gamma \frac{1}{2} \) mixed strategies to consider.

- Use approximate eq. condition:
  - replace "best-response" by "\( \varepsilon \)-best-response"
  - recall, \( \hat{\pi} \) is \( \varepsilon \)-NE if no player can gain more than \( \varepsilon \) by unilaterally deviating from \( \hat{\pi} \)

  \( \forall i, \max_a M_i(\hat{\pi}[i:a]) - M_i(\hat{\pi}) \leq \varepsilon \)

So,

- Table size: \( \Gamma \frac{1}{2} \)
- Computation time (per player): \( O(\Gamma \frac{1}{2} K) \)

[See accompanying PowerPoint presentation for an example]

Now, how should we set \( \varepsilon \) s.t.

if \( \hat{\pi} \) is NE, \( \hat{\pi} \) in \( \varepsilon \)-grid, closest(inL) to \( \hat{\pi} \),

then \( \hat{\pi} \) is \( \varepsilon \)-NE?
Approximation Algorithm (Analysis)

Let $\bar{p}, \bar{q}$ be joint mixed strategies; $K = \text{max}\text{ neighborhood size}$

**Lemma 1:** If $\forall i, p_i - q_i < \frac{\varepsilon}{2}$, then

$$|M_i(\bar{p}) - M_i(\bar{q})| \leq \frac{((1+\varepsilon)K - 1)}{2}$$

$$\leq K\varepsilon \uparrow$$

(for $\varepsilon < \frac{2}{K}$)

**Pf:** [See accompanying note]

**Lemma 2:** If $\bar{p}$ is NE, $\bar{q}$ in $\varepsilon$-grid and closest to $\bar{p}$, and $\varepsilon < \frac{2}{K}$, then $\bar{q}$ is $(2K\varepsilon)$-NE.

**Pf:** $\forall i, M_i(\bar{q}) \geq M_i(\bar{p}) - K\varepsilon$ (By Lemma 1)

$$= \max_a M_i(\bar{p}[i:a]) - K\varepsilon$$ (By NE defn)

$$\geq \max_a M_i(\bar{q}[i:a]) - K\varepsilon - K\varepsilon$$ (By Lemma 1)

$$= -2K\varepsilon$$

Let $\varepsilon = \frac{\varepsilon}{2K}$. So $\varepsilon < \frac{2K\varepsilon}{E} + 1$

⇒ Table size $\leq \left(\frac{2K\varepsilon}{E} + 1\right)^2$ ⇒ rep.size, poly in $\frac{1}{E}, K, n$

⇒ Computation per player $\leq \left(\frac{2K\varepsilon}{E} + 1\right)^K$ ⇒ running time poly in $\frac{1}{E}, n, 2^{\log K}$

**Result:**
- **ApproxTreeNash** computes an $\varepsilon$-NE
- Every NE has a representative $\varepsilon$-NE in tables.
- Table representation size, poly. in model size.
- If $K \text{ s.t. } K\log K = O(\log n)$, computation time also poly. in model size.

[What about multi-action games with $m > 2$?]
Exact Algorithm: Tree case, 2-actions [All equilibria]

Basic idea:

- Easy to compute/represent exactly the tables sent by leaves:
  - union of "axis-parallel" "line segments"
- Use (represented) exact tables received from parents to recursively compute/represent exact tables sent to child.
  - Invariance: exact tables are finite union of "axis-parallel" "line segments"

Tables sent "down" by leaves

Consider expected payoff of leaf $U_i$

$$M_{U_i}(u_i,v) = u_i \left[ M_{U_i}(1,v) - M_{U_i}(0,v) \right] + M_{U_i}(0,v)$$

$$\Delta_{U_i}(v)$$

$\forall v \in [0,1],$

$\Delta(v) > 0 \Rightarrow U_i = 1$ is best response to $V = v$

$\Delta(v) < 0 \Rightarrow U_i = 0$ 

$\Delta(v) = 0 \Rightarrow U_i = u', \forall u' \in [0,1]$ 

["$U_i$ is indifferent to $V = v"]$

How can we find "indifference" value $v'$?

[if it exists...]
Exact Alg. (Continued)

Finding "indifference" value \( v' \)

\[
\Delta(v) = v \left[ M_{u_i}(1,1) - M_{u_i}(1,0) - (M_{u_i}(0,1) - M_{u_i}(0,0)) \right] \\
+ M_{u_i}(1,0) + M_{u_i}(0,0)
\]

\( \Delta(v') = 0 \) iff either \( b = 0 = c \) or \( v' = \frac{-c}{b}, \ b \neq 0 \).

(only care about \( v \in [0,1] \) !)

Let \( u_i : [0,1] \rightarrow \mathbb{R} \) be an indicator function \( u_i(v) = I(\Delta(v) > 0) \) \( \forall (v, u_i) \in [0,1]^2 \),

\[ T_{u_i}(v, u_i) = 1 \] iff

- \( v \in [0, v'] \) and \( u_i \in [u_i(v), u_i(v')] \), or
- \( v \in [v', 1] \) and \( u_i \in [0, 1] \), or
- \( v \in [v', 1] \) and \( u_i \in [u_i(v), u_i(v')] \)

\( v \)-list representation of \( T_{u_i} \)

\[
\begin{align*}
V & \quad V' = \{ v_0, v_1, v_2, v_3 \} \\
I_0 & \quad [v_0] \\
I_1 & \quad [v_1, v_2] \\
I_2 & \quad [v_2, v_3] \\
I_3 & \quad [v_3] \\
\end{align*}
\]

In general, a sequence of points in \([0,1]\]

\( a = v_0 \leq v_1 \leq \ldots \leq v_m = 1 \)

and \( \forall k = 0, \ldots, m, I_k = [v_k, v_{k+1}] \) is the union of the intervals in \([0,1]\)

\[ I_1 \cup I_2 \]

[Remark: At least, \( t = 1 \)].
Consider $u_1, u_2, 0, u_k, v$.

**Merge** $v$-lists from all parents.

- Consider an individual $v \in [v_0, v_{t+1}]$.

$$\bar{u} \in I, x \times I_k, I_i \in \{I_j, j=1, \ldots, t\}$$

How do we find values for $\omega$ s.t. $V$ is indifferent $\forall v \in [v_0, v_{t+1}]$?

**Same idea:** $M_v(\omega, \bar{u}) = v [M_v(1, \omega, \bar{u}) - M_v(0, \omega, \bar{u})] + M_v(0, \omega, \bar{u})$

$$\Delta_v(\omega, \bar{u}) = M_v(1, \omega, \bar{u}) - M_v(0, \omega, \bar{u})$$

$$= \omega [M_v(1, 1, \bar{u}) - M_v(1, 0, \bar{u}) - (M_v(0, 1, \bar{u}) - M_v(0, 0, \bar{u}))] + M_v(1, 0, \bar{u}) - M_v(0, 0, \bar{u})$$

So we want

$$\omega \in W = \{\omega \in [0, 1] : \exists \bar{u} \in I, x \times I_k \text{ s.t. } \Delta(\omega, \bar{u}) = 0\}$$

[See accompanying paper by Kearns et al., 2001]

- Only need to check extremal points of $I, x \times I_k$!
Exact Alg.
Summary:

- Can show size of tables grow exponentially with number of players [See Kearns et al., 2001]

- Exact alg. computes a representation of all exact NE in a tree graphical game in time exponential in model size.

- Possible to generate NE from the resulting tables.
Exact Algorithm: Tree case, 2-action, single NE

- [See accompanying paper by Littman et al. 2002]

- Alg. computes single exact NE in 2-action, tree graphical games in time poly in model size.

- **Basic idea:**
  - Pick only one "path" in table $T(w,r)$ s.t.
    \[ \forall w, \exists r \text{ s.t. } T(w,r) = 1. \]
    [ignore others]
  - Which "path" should select?
    The one with minimum number of "turns"