CPS 296.1
LP and IP in Game theory
(Normal-form Games, Nash Equilibria and Stackelberg Games)

Joshua Letchford
MICKEY: All right, rock beats paper!  
(Mickey smacks Kramer's hand for losing)  
KRAMER: I thought paper covered rock.  
MICKEY: Nah, rock flies right through paper.  
KRAMER: What beats rock?  
MICKEY: (looks at hand) Nothing beats rock.

<table>
<thead>
<tr>
<th></th>
<th>0, 0</th>
<th>1, -1</th>
<th>1, -1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0, 0</td>
<td>1, -1</td>
<td>1, -1</td>
<td></td>
</tr>
<tr>
<td>-1, 1</td>
<td>0, 0</td>
<td>-1, 1</td>
<td></td>
</tr>
<tr>
<td>-1, 1</td>
<td>1, -1</td>
<td>0, 0</td>
<td></td>
</tr>
</tbody>
</table>
Dominance

• Player i’s strategy $s_i$ strictly dominates $s_i'$ if
  – for any $s_{-i}$, $u_i(s_i, s_{-i}) > u_i(s_i', s_{-i})$
• $s_i$ weakly dominates $s_i'$ if
  – for any $s_{-i}$, $u_i(s_i, s_{-i}) \geq u_i(s_i', s_{-i})$; and
  – for some $s_{-i}$, $u_i(s_i, s_{-i}) > u_i(s_i', s_{-i})$

\[-i = “the player(s) other than i”\]
Mixed strategies

- **Mixed strategy** for player i = probability distribution over player i’s (pure) strategies
- E.g., 1/3, 1/3, 1/3

- Example of dominance by a mixed strategy:

<table>
<thead>
<tr>
<th></th>
<th>3, 0</th>
<th>0, 0</th>
</tr>
</thead>
<tbody>
<tr>
<td>1/2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0, 0</td>
<td></td>
<td>3, 0</td>
</tr>
<tr>
<td>1/2</td>
<td></td>
<td>1, 0</td>
</tr>
</tbody>
</table>

Usage:
\( \sigma_i \) denotes a mixed strategy,
\( s_i \) denotes a pure strategy
Checking for dominance by mixed strategies

• Linear program for checking whether strategy \( s_i^* \) is strictly dominated by a mixed strategy:
  • maximize \( \varepsilon \)
  • such that:
    - for any \( s_{-i} \), \( \sum_{s_i} p_{s_i} u_i(s_i, s_{-i}) \geq u_i(s_i^*, s_{-i}) + \varepsilon \)
    - \( \sum_{s_i} p_{s_i} = 1 \)

• Linear program for checking whether strategy \( s_i^* \) is weakly dominated by a mixed strategy:
  • maximize \( \sum_{s_{-i}} [(\sum_{s_i} p_{s_i} u_i(s_i, s_{-i})) - u_i(s_i^*, s_{-i})] \)
  • such that:
    - for any \( s_{-i} \), \( \sum_{s_i} p_{s_i} u_i(s_i, s_{-i}) \geq u_i(s_i^*, s_{-i}) \)
    - \( \sum_{s_i} p_{s_i} = 1 \)
Best-response strategies

- Suppose you know your opponent’s mixed strategy
  - E.g., your opponent plays rock 50% of the time and scissors 50%
- What is the best strategy for you to play?
- Rock gives \(.5 \times 0 + .5 \times 1 = .5\)
- Paper gives \(.5 \times 1 + .5 \times (-1) = 0\)
- Scissors gives \(.5 \times (-1) + .5 \times 0 = -.5\)
- So the best response to this opponent strategy is to (always) play rock
- There is always some pure strategy that is a best response
  - Suppose you have a mixed strategy that is a best response; then every one of the pure strategies that that mixed strategy places positive probability on must also be a best response
### How to play matching pennies

#### Table

<table>
<thead>
<tr>
<th></th>
<th>L</th>
<th>R</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Us</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>L</td>
<td>1, -1</td>
<td>-1, 1</td>
</tr>
<tr>
<td>R</td>
<td>-1, 1</td>
<td>1, -1</td>
</tr>
</tbody>
</table>

#### Strategy

- Assume opponent knows our mixed strategy.
- If we play L 60%, R 40%...
- … opponent will play R…
- … we get $0.6(-1) + 0.4(1) = -0.2$
- What’s optimal for us? What about rock-paper-scissors?
General-sum games

- You could still play a minimax strategy in general-sum games
  - I.e., pretend that the opponent is only trying to hurt you

- But this is not rational:

<table>
<thead>
<tr>
<th></th>
<th>0, 0</th>
<th>3, 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>1, 0</td>
<td>2, 1</td>
<td></td>
</tr>
</tbody>
</table>

- If Column was trying to hurt Row, Column would play Left, so Row should play Down
- In reality, Column will play Right (strictly dominant), so Row should play Up
- Is there a better generalization of minimax strategies in zero-sum games to general-sum games?
Nash equilibrium

[Nash 50]

- A vector of strategies (one for each player) is called a strategy profile
- A strategy profile \((\sigma_1, \sigma_2, \ldots, \sigma_n)\) is a Nash equilibrium if each \(\sigma_i\) is a best response to \(\sigma_{-i}\)
  - That is, for any \(i\), for any \(\sigma_i'\), \(u_i(\sigma_i, \sigma_{-i}) \geq u_i(\sigma_i', \sigma_{-i})\)
- Note that this does not say anything about multiple agents changing their strategies at the same time
- In any (finite) game, at least one Nash equilibrium (possibly using mixed strategies) exists [Nash 50]
- (Note - singular: equilibrium, plural: equilibria)
The presentation game

<table>
<thead>
<tr>
<th>Audience</th>
<th>Presenter</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Pay attention (A)</strong></td>
<td><strong>Put effort into</strong></td>
<td><strong>Do not put</strong></td>
</tr>
<tr>
<td></td>
<td><strong>presentation (E)</strong></td>
<td>effort into</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>presentation (NE)</strong></td>
</tr>
<tr>
<td><strong>Do not pay</strong></td>
<td>4, 4</td>
<td>-16, -14</td>
</tr>
<tr>
<td><strong>attention (NA)</strong></td>
<td>0, -2</td>
<td>0, 0</td>
</tr>
</tbody>
</table>

- Pure-strategy Nash equilibria: (A, E), (NA, NE)
- Mixed-strategy Nash equilibrium:
  - $\((1/10 \text{ A}, 9/10 \text{ NA}), (4/5 \text{ E}, 1/5 \text{ NE})\)$
    - Utility 0 for audience, -14/10 for presenter
    - Can see that some equilibria are strictly better for both players than other equilibria
Some properties of Nash equilibria

• If you can eliminate a strategy using strict dominance or even iterated strict dominance, it will not occur (i.e., it will be played with probability 0) in every Nash equilibrium
  – Weakly dominated strategies may still be played in some Nash equilibrium

• In 2-player zero-sum games, a profile is a Nash equilibrium if and only if both players play minimax strategies
  – Hence, in such games, if \( (\sigma_1, \sigma_2) \) and \( (\sigma_1', \sigma_2') \) are Nash equilibria, then so are \( (\sigma_1, \sigma_2') \) and \( (\sigma_1', \sigma_2) \)
    • No equilibrium selection problem here!
Solving for a Nash equilibrium using MIP (2 players)

[Sandholm, Gilpin, Conitzer AAAI05]

- maximize *whatever you like* (e.g., social welfare)
- subject to
  - for both $i$, $\sum s_i p_{s_i} = 1$
  - for both $i$, for any $s_i$, $\sum_{s_{-i}} p_{s_{-i}} u_i(s_i, s_{-i}) = u_{s_i}$
  - for both $i$, for any $s_i$, $u_i \geq u_{s_i}$
  - for both $i$, for any $s_i$, $p_{s_i} \leq b_{s_i}$
  - for both $i$, for any $s_i$, $u_i - u_{s_i} \leq M(1 - b_{s_i})$

- $b_{s_i}$ is a binary variable indicating whether $s_i$ is in the support, $M$ is a large number
Stackelberg (commitment) games
(My research)

- Unique Nash equilibrium is (R,L)
  - This has a payoff of (2,1)
• What if the officer has the option to (credibly) announce where he will be patrolling?
• This would give him the power to “commit” to being at one of the buildings
  – This would be a pure-strategy Stackelberg game
Commitment...

<table>
<thead>
<tr>
<th></th>
<th>L</th>
<th>R</th>
</tr>
</thead>
<tbody>
<tr>
<td>L</td>
<td>(1,-1)</td>
<td>(3,1)</td>
</tr>
</tbody>
</table>

- If the officer can commit to always being at the left building, then the vandal's best response is to go to the right building.
  - This leads to an outcome of (3,1)
Committing to mixed strategies

<table>
<thead>
<tr>
<th></th>
<th>L</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>L</td>
<td>(1,-1)</td>
<td>(3,1)</td>
</tr>
<tr>
<td>R</td>
<td>(2,1)</td>
<td>(4,-1)</td>
</tr>
</tbody>
</table>

• What if we give the officer even more power: the ability to commit to a mixed strategy
  – This results in a mixed-strategy Stackelberg game
  – E.g., the officer commits to flip a weighted coin which decides where he patrols
Committing to mixed strategies is more powerful

<table>
<thead>
<tr>
<th></th>
<th>L</th>
<th>R</th>
</tr>
</thead>
<tbody>
<tr>
<td>L</td>
<td>(1, -1)</td>
<td>(3, 1)</td>
</tr>
<tr>
<td>R</td>
<td>(2, 1)</td>
<td>(4, -1)</td>
</tr>
</tbody>
</table>

• Suppose the officer commits to the following strategy: \{(.5+\epsilon)L, (.5- \epsilon)R\}
  – The vandal’s best response is R
  – As \epsilon goes to 0, this converges to a payoff of (3.5, 0)
Stackelberg games in general

• One of the agents (the leader) has some advantage that allows her to commit to a strategy (pure or mixed)

• The other agent (the follower) then chooses his best response to this
Visualization

<table>
<thead>
<tr>
<th></th>
<th>L</th>
<th>C</th>
<th>R</th>
</tr>
</thead>
<tbody>
<tr>
<td>U</td>
<td>0,1</td>
<td>1,0</td>
<td>0,0</td>
</tr>
<tr>
<td>M</td>
<td>4,0</td>
<td>0,1</td>
<td>0,0</td>
</tr>
<tr>
<td>D</td>
<td>0,0</td>
<td>1,0</td>
<td>1,1</td>
</tr>
</tbody>
</table>

(0,1,0) = M
(1,0,0) = U
(0,0,1) = D
Easy polynomial-time algorithm for two players

- For every column t separately, we solve separately for the best mixed row strategy (defined by \( p_s \)) that induces player 2 to play t
- maximize \( \sum_s p_s u_1(s, t) \)
- subject to
  - for any \( t' \), \( \sum_s p_s u_2(s, t) \geq \sum_s p_s u_2(s, t') \)
  - \( \sum_s p_s = 1 \)
- (May be infeasible)
- Pick the t that is best for player 1
(a particular kind of) Bayesian games

<table>
<thead>
<tr>
<th>leader utilities</th>
<th>follower utilities (type 1)</th>
<th>follower utilities (type 2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 4</td>
<td>1 0</td>
<td>1 0</td>
</tr>
<tr>
<td>1 3</td>
<td>0 1</td>
<td>1 3</td>
</tr>
</tbody>
</table>

probability .6 probability .4
Solving Bayesian games

• There’s a known MIP for this\(^1\)

• Details omitted due to the fact that its rather nasty.
  • The main trick of the MIP is encoding a exponential number of LP’s into a single MIP
  • Used in the ARMOR system deployed at LAX

(In)approximability

- (# types)-approximation: optimize for each type separately using the LP method. Pick the solution that gives the best expected utility against the entire type distribution.

- Can’t do any better in polynomial time, unless P=NP
  - Reduction from INDEPENDENT-SET

- For adversarially chosen types, cannot decide in polynomial time whether it is possible to guarantee positive utility, unless P=NP
  - Again, a MIP formulation can be given
Reduction from independent set

**leader utilities**

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a_i^1$</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>$a_i^2$</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>$a_i^3$</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

**follower utilities (type 1)**

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a_i^1$</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>$a_i^2$</td>
<td>0</td>
<td>10</td>
</tr>
<tr>
<td>$a_i^3$</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

**follower utilities (type 2)**

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a_i^1$</td>
<td>0</td>
<td>10</td>
</tr>
<tr>
<td>$a_i^2$</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>$a_i^3$</td>
<td>0</td>
<td>10</td>
</tr>
</tbody>
</table>

**follower utilities (type 3)**

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a_i^1$</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>$a_i^2$</td>
<td>0</td>
<td>10</td>
</tr>
<tr>
<td>$a_i^3$</td>
<td>3</td>
<td>1</td>
</tr>
</tbody>
</table>
Extensive-form games

• Often games have an inherent time structure
  – In these cases, it is often easier to represent these games in the extensive form

• The focus of my most recent paper (EC ‘10) was to determine in which extensive-form games the Stackelberg solution can be found efficiently
Stackelberg games in extensive form

Player 1

(0,3) (1,3) (0,1)

Player 2

(2,2) (3,0)

Subgame Perfect Nash Equilibrium

Mixed Strategy Nash Equilibrium

Pure strategy commitment

Mixed strategy commitment

(4,5,2)
Other aspects considered

- Pure or mixed strategy commitment
- Perfect vs imperfect information
- Chance nodes
- Restricted or costly commitment
  - Player 1 either incurs a cost for committing at some nodes/information sets or is unable to do so
- Tree vs DAG
  - The key difference in a DAG is the inability for player 1 to commit differently based on what path is taken to a node/information set
Overview of results (decision tree)
Case 1: pure strategy commitment

THEOREM. Can be solved in O(nm) time when:

- perfect information
- tree form
- no chance nodes
- no costs/restrictions
- pure strategy commitment
- any number of players

\( n \) is the number of internal nodes, \( m \) the number of leaf nodes
Case 1: algorithm

• Two main steps
  – An upward pass to determine what subset of each node’s descendant leaf nodes can be achieved
  – A downward pass to determine the correct commitment at each node
    • This is both on and off the path to the desired outcome
The upward pass

• At player 1 nodes
  – Take the union of all children’s achievable sets

• At player i ≠ 1 nodes
  – Determine the **pruning value** for each child
    • \( \max_{\text{other children}} \min u_i \)
    • This is how much we can punish player i for not going to this child
  – Prune each set, take the union of what remains
Case 1 example: upward pass

Player 2

pruning value = 0

Player 1

Left

((1,3),(0,1))

pruning value = 1

Player 1

((2,2),(3,0))

(1, 3) (0, 1) (2, 2) (3, 0)
The downward pass

- A recursive algorithm
  - At player 1 nodes
    - Simply commit on the path to the desired node and recurse on that child
  - At player \( i \neq 1 \) nodes
    - Recurse towards the desired outcome, as well as to the smallest outcome for every other child
Case 1 example: downward pass

Player 1

((1,3), (0,1))

Left

(1, 3)  (0, 1)

Player 2

((1,3), (0,1), (2,2))

Player 1

((2,2), (3,0))

(2, 2)  (3, 0)
Case 2: mixed (behavioral) strategy commitment

THEOREM. Can be solved in $O(nm^2)$ time when:

- perfect information
- tree form
- no chance nodes
- no costs/restrictions
- mixed strategy commitment
- two players

$n$ is the number of internal nodes, $m$ the number of leaf nodes
Case 2: algorithm (sketch)

• Two main steps
  – An upward pass to determine what mixtures of each node’s descendants can be achieved
  – A downward pass to determine the correct commitment to achieve the best mixed strategy
The upward pass

• This time we will need to store mixed strategies (meaning convex sets), rather than points
  – It turns out that since our eventual goal is to maximize player 1’s utility, that maintaining the ceiling of the convex sets is enough (line segments)
  – For computational reasons, we will not actually ever compute the ceiling, but instead maintain a slightly larger superset of the ceiling
The upward pass

- At player 1 nodes
  - Take the union of all children’s achievable sets
    - Represented as line segments
  - Also, for endpoints of line segments from two different children, can take convex combinations
    - This may result in another segment
    - These endpoints will either be leaf nodes or generated at player 2 nodes
The upward pass

• At player 2 nodes
  – For each child find the pruning value
  – Prune each line segment at this value (if either end point is smaller than this value)
  – Take the union of all children’s achievable sets
Case 2 example: upward pass

Player 1

(1, 3)  (0, 1)

Left

Player 2

pruning value = 0

(((1,3),(0,1)) ,
((2,2), (2.5,1)))

pruning value = 1

Player 1

(2.5,1)

((2,2),(3,0))
The downward pass

- A recursive algorithm
  - At player 1 nodes
    - Compute and commit to the necessary probabilities
    - Recurse on the children that receive positive probability
  - At player 2 nodes
    - Recurse towards the desired outcome, as well as to the smallest outcome on every other child

(note: player 2 does not ever need to randomize)
Case 2 example: downward pass

Player 2

(((1,3),(0,1)),
((2.5,1),(2,2)))

Player 1

(((1,3),(0,1)),
((2,2),(3,0)))

Left

(1, 3)

(0, 1)
Chance nodes

- Moves by a player with a fixed behavioral strategy that has no stake in the game
  - Usually referred to as moves by Nature.
  - Behavioral strategy is common knowledge
  - We don’t include Nature when we count the number of players
THEOREM. It is NP-hard to solve for the optimal strategy to commit to in a game with:

- chance nodes,
- two players
- tree form
- perfect information
- no costs/restrictions
- pure or mixed strategy commitment

• We prove this via reduction from Knapsack.
Knapsack

- Set of $N$ items
  - Each has a value $p_i$ and a weight $w_i$
- Find a subset of items that
  - Maximizes the sum of the $p_i$ of the items in the subset
  - s.t. the sum of the $w_i$ of the items in the subset is below a given limit $W$. 
Knapsack reduction

• One subtree for each item
  – Player 1 commitment determines if the item is included or not

• Chance node to make all items be considered

• Player 2 at the top makes a choice to enforce the weight limit
Knapsack reduction

Forces all items to be considered

Item 1’s subtree

Imposes the weight constraint
Open questions

• Are there good heuristics/approximation algorithms for any of the NP-hard cases?
• Are there other restrictions that allow for fast algorithms?
• Are the given algorithms tight or is there room for improvement?
Thank you for your attention
Pure-strategy extensive form representation of normal form
Mixed strategy extensive form representation of normal form

While conceptually useful, this is not useful computationally: the tree has infinite size.
Tie breaking

• As is commonly done, we assume that all players break ties in player 1’s favor.

• Consider a case where player 1 makes a mixed strategy commitment between two choices, (1,0), and (0,1).

• If player 2 has choice between the result of player 1’s commitment and (0,.5):
  – Player 1 can commit to a (.5+\epsilon) probability of playing (0,1) and a (.5-\epsilon) probability of playing (1,0).
  – Then, player 2 will prefer the outcome of player 1’s commitment.
Player 1

Player 2

(1,0) (=Left)

(0,1) (=Right)

Player 2

(1, -1)

(2, 1)

(3, 1)

(4, -1)
DAG example

Player 1

Player 2

Player 2

(2, 0)  (1,0)  (0, 2)  (0, 1)