Strictly Competitive Games

Felix Munoz-Garcia

EconS 503 - Washington State University
Osborne, Chapter 11:
- Posted on Angel website.
- It is only 15 pages long, including examples, figures, experiments, etc.
Some strategic situations involve players with completely opposite interests/incentives.

We analyze those situations with Strictly Competitive Games.

They are a type of simultaneous-move games, as those described so far...

but with an additional assumption (next slide):
Strictly competitive games

**Definition**

A two-player, strictly competitive game is a two-player game with the property that, for every two strategy profiles $s$ and $s'$,

$$u_1(s) > u_1(s') \quad \text{and} \quad u_2(s) < u_2(s')$$

- **Intuition**: Hence, players have exactly opposite rankings over the outcomes resulting from the strategy profile $s$ and $s'$.
- Alternatively: if my payoff increases if we play $s = (s_1, s_2)$ rather than $s' = (s'_1, s'_2)$, then your payoff must decrease.
An implication is that, in a strictly competitive game,

- if $u_1(s) = u_1(s')$, then $u_2(s) = u_2(s')$. 
Alternatively, to check if a game is not strictly competitive, we want to find two strategy profiles (cells), $s$ and $s'$ for which players’ preferences are aligned, that is,

- $u_1(s) > u_1(s')$, then $u_2(s) > u_2(s')$
Example 1 - Matching pennies

### Player 2

<table>
<thead>
<tr>
<th></th>
<th>Heads</th>
<th>Tails</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Player 1</strong></td>
<td>Heads</td>
<td>1, -1</td>
</tr>
<tr>
<td></td>
<td>Tails</td>
<td>-1, 1</td>
</tr>
</tbody>
</table>

- One example of a strategy profile is $s = (H, H)$, and another is $s' = (T, H)$, where
  
  $$u_1(s) > u_1(s') \quad \text{and} \quad u_2(s) < u_2(s')$$

- Importantly, this is true for **any two** strategy profiles: if one player is improving his payoff, the other player is reducing his.
- In fact, many board games satisfy this condition: if we play in such a way that I end up winning, it must be that my opponent loses, and vice versa.
- **Examples:** tennis, chess, football, etc.
Practice:

For the following games, determine which of them satisfy the definition of strictly competitive games:

1. Matching Pennies (Anticoordination game),
2. Prisoner’s Dilemma,
3. Battle of the Sexes (Coordination game).
Check if this game satisfies the definition of strictly competitive games.

Recall that we must check that, for any two strategy profiles \( s \) and \( s' \),

\[
\begin{align*}
    u_1(s) &> u_1(s') \quad \text{and} \quad u_2(s) < u_2(s')
\end{align*}
\]
Matching Pennies

- Comparing each possible pair of outcomes
  
  1. \( u_1(H, H) > u_1(H, T) \), i.e., \( 1 > -1 \)
     \( u_2(H, H) < u_2(H, T) \), i.e., \( -1 > 1 \)
  
  2. \( u_1(H, T) < u_1(T, T) \), i.e., \( -1 < -1 \)
     \( u_2(H, T) > u_2(T, T) \), i.e., \( 1 > -1 \)
  
  3. \( u_1(H, H) = u_1(T, T) \), i.e., \( 1 = 1 \)
     \( u_2(H, H) = u_2(T, T) \), i.e., \( -1 = -1 \)
  
  4. \( u_1(H, T) = u_1(T, H) \), i.e., \( -1 = -1 \)
     \( u_2(H, T) = u_2(T, H) \), i.e., \( 1 = 1 \)
Check if this game satisfies the above definition of strictly competitive games.

[Hint: What happens when you compare (C, C) and (NC, NC)? Preference alignment].

Player 1

<table>
<thead>
<tr>
<th></th>
<th>Confess</th>
<th>Not Confess</th>
</tr>
</thead>
<tbody>
<tr>
<td>Confess</td>
<td>-5, -5</td>
<td>0, -15</td>
</tr>
<tr>
<td>Not Confess</td>
<td>-15, 0</td>
<td>-1, -1</td>
</tr>
</tbody>
</table>

Player 2
### Battle of the Sexes

<table>
<thead>
<tr>
<th></th>
<th>Football</th>
<th>Opera</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Husband</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Football</td>
<td>3, 1</td>
<td>0, 0</td>
</tr>
<tr>
<td>Opera</td>
<td>0, 0</td>
<td>1, 3</td>
</tr>
</tbody>
</table>

**Wife**

Check if this game is strictly competitive game:

- **[Hint: What happens when you compare \((F, O)\) and \((F, F)\)? Preference alignment].**
Zero-sum games

- An interesting class of strictly competitive games: zero-sum games.
- A zero-sum game is an strictly competitive game in which the payoffs of the two players adds up to zero. That is
  \[ u_1(s_1, s_2) + u_2(s_1, s_2) = 0 \]
  for any strategy pair \((s_1, s_2)\).
  Alternatively, \[ u_1(s_1, s_2) = -u_2(s_1, s_2) \].
- For a general strictly competitive game we were saying that:
  “if one strategy profile \((s_1, s_2)\) increases my payoff, then...
  ...such strategy must reduce your payoff”
- but in a zero-sum game we are imposing an stronger assumption:
  “the payoff that I gain, is exactly what you lose”
Zero-sum games

- The definition of a zero-sum game was satisfied by the matching pennies game....

\[
\begin{array}{c|cc}
\text{Player 2} & \text{Heads} & \text{Tails} \\
\hline
\text{Heads} & 1, -1 & -1, 1 \\
\text{Tails} & -1, 1 & 1, -1 \\
\end{array}
\]

since \( u_1(s_1, s_2) + u_2(s_1, s_2) = 1 + (-1) = 0 \), for any strategy profile that specifies one strategy for player 1 and one for player 2, \((s_1, s_2)\).
Some games are not zero-sum games, but they are constant-sum games. (They are of course an special type of strictly competitive game; verify).

The following *Tennis game* is a constant-sum game

\[
\begin{array}{c|cc}
\text{Player 2} & \text{Right} & \text{Left} \\
\hline
\text{Player 1} & 20, 80 & 70, 30 \\
\text{Right} & 90, 10 & 30, 70 \\
\end{array}
\]

since \( u_1(s_1, s_2) + u_2(s_2, s_1) = 100 \), for any strategy profile \((s_1, s_2)\) that specifies one strategy for player 1 and one for player 2.
Constant-sum games

Hence, for any strategy profile \((s_1, s_2)\)

- \(u_1(s_1, s_2) + u_2(s_1, s_2) = 0\), in zero-sum games
- \(u_1(s_1, s_2) + u_2(s_1, s_2) = \text{Constant}\), in constant-sum games
  - The Constant is exactly equal to zero in zero-sum games

Therefore
Compact representation of the Tennis game:

\[
\begin{array}{c|cc}
\text{Player 1} & \text{Right} & \text{Left} \\
\hline
\text{Right} & 20 & 70 \\
\text{Left} & 90 & 30 \\
\end{array}
\]

We don’t need to represent player 2’s payoff, since we know that in this constant-sum game

\[u_1(s_1, s_2) + u_2(s_1, s_2) = 100\]

for all strategy profiles \((s_1, s_2)\).

Hence, player 2’s payoffs are 80, 30, 10, and 70.
Constant-sum games

How to solve this class of games?

- We could use the NE solution concept (implying the need to rely on msNE for most of these games).
- An alternative, historically developed before John Nash introduced his "NE solution concept," is to use the so-called:
  - **Security strategies**
    (also referred as Max-Min strategies).
Security or Max-min strategy

- Compact representation of the Tennis game:

  Player 2

  Player 1
  \[\begin{array}{cc}
  \text{Right} & \text{Left} \\
  20 & 70 \\
  90 & 30
  \end{array}\]

- Note that Player 1 wants to maximize his own payoffs, and...
- Player 2 also wants to maximize his own payoffs, which implies minimizing Player 1’s payoffs, since we are in a constant-sum game.
Security or Max-min strategy

- Let us put ourselves in the worst case scenario:
  - First, for a given strategy $s_1$ that player 1 selects, choose the strategy of player 2’s that minimizes player 1’s payoffs.
    \[ w_1(s_1) = \min_{s_2} u_1(s_1, s_2) \]
    we refer to $w_1(s_1)$ as the worst payoff that player 1 could achieve by selecting strategy $s_1$.
  - Alternatively, we can interpret that, if player 1 select strategy $s_1$, he guarantees to obtain a payoff of at least $w_1(s_1)$.

- A Security strategy gives player 1 the best of the worst case scenarios:
  \[ \max_{s_1} w_1(s_1) = \max_{s_1} \min_{s_2} u_1(s_1, s_2) \]
  The strategy that solves this maximization problem is referred as the Security strategy, or Max-min strategy.
Security or Max-min strategy

\[
\max_{s_1} w_1(s_1) = \max_{s_1} \min_{s_2} u_1(s_1, s_2)
\]

- The payoff \( \max_{s_1} w_1(s_1) \) is usually referred as the Security-payoff level.

- Note what is happening here:
  - I maximize my payoff, given that I know that my opponent will minimize it (because he wants to maximize his own payoff).
We can generalize the above definition to mixed strategies, i.e., talking about $\sigma_i$ rather than $s_i$.

Player 1’s security payoff level is

$$\max_{\sigma_1} w_1(\sigma_1) = \max_{\sigma_1} \min_{s_2} u_1(\sigma_1, s_2)$$

And similarly for player 2:

$$\max_{\sigma_2} w_2(\sigma_2) = \max_{\sigma_2} \min_{s_1} u_2(\sigma_2, s_1)$$
Let us first apply Security strategies to the example of the Matching pennies game.

Afterwards, we will apply the same methodology to the Tennis game.
Security or Max-min strategy

Note that, in order to find the security (or max-min) strategy for player 1, we need to find

$$\max_{s_1} \min_{s_2} u_1(s_1, s_2)$$

We hence need to first find:

- $EU_1(p|H)$ conditional on player 2 choosing H.
- $EU_1(p|T)$ conditional on player 2 choosing T.

We can then find the min of these two expressions (i.e., their "lower envelope").

Finally, we can find the max of the min.

Confused? Ok, let’s do one example together.
1st step: Find the expected payoff of player 1

If player 1 chooses $H$ (In the first column), player 1’s EU becomes:

$$EU_1(p|H) = 1 \cdot p + (-1)(1 - p) = 2p - 1$$

If player 1 chooses $T$ (In the second column), player 1’s EU becomes:

$$EU_1(p|T) = (-1) \cdot p + 1(1 - p) = 1 - 2p$$
Remark

Note that $EU_1(p|H)$ represents the expected utility that player 1 obtains from randomizing between $H$ (with probability $p$) and $T$ (with probability $1 - p$), conditional on player 2 selecting Heads (in the first column).

Do not confuse it with $EU_1(H)$ that we used in msNE, which reflects player 1’s expected utility from selecting $H$ with certainty but facing a randomization from his opponent (e.g., player 2 randomizing between $H$ and $T$ with probability $q$ and $1 - q$ respectively.)
2nd step: Let’s graphically depict $EU_1(p|H)$ and $EU_1(p|T)$
3rd step: Identify the lower envelope, i.e.,

$$\min_{s_2} u_1(s_1, s_2)$$

$$EU_1(p) = \begin{cases} 2p - 1 & \text{for } H \\ 1 - 2p & \text{for } T \end{cases}$$

$$\min_{s_2} u_1(s_1, s_2)$$ is the lower envelope.
4th step: Identify the highest peak of the lower envelope: i.e.,

$$\max_{s_1} \min_{s_2} u_1(s_1, s_2)$$
Summarizing our results...

- We just found that the Security (or Max-Min) strategy for player 1 is:
  - To choose Heads with probability $p = \frac{1}{2}$.

- What about player 2?
  - Well, we have to follow the same procedure we used with player 1.
  - Practice on your own (see next two slides).
  - [Hint: you should find that player 2 also randomizes with probability $q = \frac{1}{2}$].
Similarly for player 2 (Practice!)

<table>
<thead>
<tr>
<th></th>
<th>Heads</th>
<th>Tails</th>
</tr>
</thead>
<tbody>
<tr>
<td>Heads</td>
<td>1, -1</td>
<td>-1, 1</td>
</tr>
<tr>
<td>Tails</td>
<td>-1, 1</td>
<td>1, -1</td>
</tr>
</tbody>
</table>

Expected payoff for player 2:

- If player 1 plays $H$ (first row):
  \[ EU_2(q|H) = \]

- If player 2 plays $T$ (second column):
  \[ EU_2(q|T) = \]
Graphical depiction for player 2

\[ EU_2(q|\cdot) \]

-1 0 1

\[ q \]

1
Let’s go back to the Tennis game:

- It is a constant-sum game, since the sum of players’ payoffs is equal to a constant (100), for all possible strategy profiles (i.e., for all possible cells in the matrix), but...
- It is not a zero-sum game, since the sum of players’ payoffs is not equal to zero for all strategy profiles.

| Player 1 | Player 2
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Right</td>
<td>Right</td>
</tr>
<tr>
<td>20, 80</td>
<td>70, 30</td>
</tr>
<tr>
<td>Left</td>
<td>90, 10</td>
</tr>
</tbody>
</table>

Let’s start with player 1:
Security (Max-min) Strategy - Tennis Game

**Player 1**

\[ \begin{array}{c|cc}
   & Right & Left \\
   \hline
   p & 20, 80 & 70, 30 \\
   1 - p & 90, 10 & 30, 70 \\
\end{array} \]

**Player 2**

- **Player 1**’s expected payoff:
  - If player 2 chooses Right:
    \[ EU_1(p|R) = 20p + 90(1 - p) = 90 - 70p \]
  - If player 2 chooses Left:
    \[ EU_1(p|L) = 70p + 30(1 - p) = 30 + 40p \]
Graphical depiction for player 1:

- Lower envelope: \( \min_{s_2} u_1(s_1, s_2) \)

\[ EU_1(p|L) = 30 + 40p \]
\[ EU_1(p|R) = 90 - 70p \]
**Trick:**

\[
\max_{s_1} \min_{s_2} u_1(s_1, s_2)
\]

coincides with the value of \( p \) for which

\[
EU_1(p|R) = EU_1(p|L).
\]

That is,

\[
70 - 70p = 30 + 40p \implies 60 = 110p \implies p = \frac{6}{11}
\]

Hence,

\[
EU_1(p|R) = 90 - 70 \cdot \frac{6}{11} = \frac{570}{11}
\]

This is the height of the highest peak in the lower envelope.
Security (Max-min) Strategy - Tennis Game

- Similarly, for player 2

<table>
<thead>
<tr>
<th>Player 1</th>
<th>Player 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Right</td>
<td>20, 80</td>
</tr>
<tr>
<td></td>
<td>70, 30</td>
</tr>
<tr>
<td>Left</td>
<td>90, 10</td>
</tr>
<tr>
<td></td>
<td>30, 70</td>
</tr>
</tbody>
</table>

- Player 2’s expected payoff:
  - If player 1 chooses Right (first row):
    \[ EU_2(q|R) = 80q + 30(1-q) = 30 + 50q \]
  - If player 2 chooses Left (second row):
    \[ EU_2(q|L) = 10q + 70(1-q) = 70 - 60q \]
Graphical depiction for player 2 (Practice!):

\[ EU_2(p|\cdot) \]
Security or Max-min strategy

- Security strategies were introduced at the beginning of the century before Nash came out with his equilibrium concept...
- For this reason, solving a game using security strategies does not necessarily give us the same equilibrium prediction as if we use Nash equilibrium.
- Although there is one exception!
  - Such exception is, of course, strictly competitive games.
Hence,

Let us see the relationship between the equilibrium predictions using Security strategies and that using NE.
Relationship between Security strategies and NE strategies:

- If a two-player game is strictly competitive and has a Nash equilibrium \( s^* = (s_1^*, s_2^*) \), then...
- \( s_1^* \) is a security strategy for player 1 and \( s_2^* \) is a security strategy for player 2.

That is, if \( s_1^* \) is a NE strategy for player 1 in a strictly competitive game, then \( s_1^* \) guarantees player 1 at least his security payoff level, regardless of what player 2 does.

In other words, by playing the NE strategy a player guarantees a payoff equal or higher than that he would obtain by playing the Security (or Maxmin) strategy.
2. Consider the following game:

\[
\begin{array}{c|cc}
\text{Player 1} & \text{Top} & \text{Bottom} \\
\hline
\text{Left} & 6, 0 & 3, 2 \\
\text{Right} & 0, 6 & 6, 0 \\
\end{array}
\]

- Find every player’s maxmin strategy.
- What is every player’s expected payoff from playing her maxmin strategy?
- Find every player’s Nash equilibrium strategy, both using pure strategies (psNE) and using mixed strategies (msNE).
- What is every player’s expected payoff from playing her Nash equilibrium strategy?
- Compare players’ payoff when they play maxmin and Nash equilibrium strategies (from parts (b) and (d), respectively). Which is higher?
What if a game is not strictly competitive?

Consider the following game:

<table>
<thead>
<tr>
<th></th>
<th>X</th>
<th>Y</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>A</strong></td>
<td>3, 5</td>
<td>-1, 1</td>
</tr>
<tr>
<td><strong>B</strong></td>
<td>2, 6</td>
<td>1, 2</td>
</tr>
</tbody>
</table>

Player 2

This is indeed an example of a game that **does not satisfy** the definition of strictly competitive games. In particular, we can find two strategy profiles, \((A, X)\) and \((A, Y)\) for which

\[
 u_1(A, X) > u_1(A, Y) \text{ for player 1,}
\]

but also

\[
 u_2(A, X) > u_2(A, Y) \text{ for player 2.}
\]
The game has a unique psNE: \((A, X)\).

But, is \(A\) the security strategy for player 1?

- We know that this is the case in strictly competitive games, but...
- this is not necessarily true in games that are not strictly competitive (such as this one).
- In order to check if \(A\) is a security strategy for player 1, let's find player 1's security strategies →
What if a game is not strictly competitive?

<table>
<thead>
<tr>
<th>Player 1</th>
<th>( p )</th>
<th>( 1-p )</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td></td>
<td></td>
</tr>
<tr>
<td>B</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Player 2</th>
<th>X</th>
<th>Y</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>3, 5</td>
<td>-1, 1</td>
</tr>
<tr>
<td>B</td>
<td>2, 6</td>
<td>1, 2</td>
</tr>
</tbody>
</table>

In order to check if \( A \) is the security strategy for player 1, we must find \( EU_1(p|X) \) and \( EU_1(p|Y) \).

\[
EU_1(p|X) = 3p + 2(1 - p) = 2 + p
\]

\[
EU_1(p|Y) = -1p + 1(1 - p) = 1 - 2p
\]
What if a game is not strictly competitive?

- Graphical representation for player 1:

\[
EU_1(p|X) = 2 + p
\]

\[
EU_1(p|Y) = 1 - 2p
\]

\[
\min_{s_2} u_1(s_1, s_2) \text{ is the lower envelope}
\]
What if a game is not strictly competitive?

- $EU_1(p|X)$ and $EU_1(p|Y)$ do not cross for any probability $p \in (0, 1)$.
- $EU_1(p|Y)$ is the minimum of $EU_1(p|X)$ and $EU_1(p|Y)$, i.e., the "Lower envelope."
- The lower envelope is maximized at $p = 0$.
- Hence, player 1 does not assign any probability to action $A$, but full probability to $B$
  - That is, $B$ is player 1’s security strategy (which differs from his Nash Equilibrium strategy, $A$).
What if a game is not strictly competitive?

- This confirms our previous result that:
  1. NE and Security strategies **coincide** for strictly competitive games, but...
  2. NE and Security strategies **do not generally coincide** for games that are not strictly competitive.

![Diagram showing the relationship between Nash Equilibrium and Security (Max-min) Strategies in the context of strictly competitive and non-strictly competitive games.](image-url)