Merging Logics of Rational Agency

✓ Entangling Knowledge/Beliefs and Preferences

✓ “Epistemizing” Logics of Action and Ability

✓ BDI (Belief + Desires + Intentions) Logics
Game theory is full of deep puzzles, and there is often disagreement about proposed solutions to them.
Game theory is full of deep puzzles, and there is often disagreement about proposed solutions to them. The puzzlement and disagreement are neither empirical nor mathematical but, rather, concern the meanings of fundamental concepts (‘solution’, ‘rational’, ‘complete information’) and the soundness of certain arguments...
Logic and Game Theory

Game theory is full of deep puzzles, and there is often disagreement about proposed solutions to them. The puzzlement and disagreement are neither empirical nor mathematical but, rather, concern the meanings of fundamental concepts (‘solution’, ‘rational’, ‘complete information’) and the soundness of certain arguments...Logic appears to be an appropriate tool for game theory both because these conceptual obscurities involve notions such as reasoning, knowledge and counter-factuality which are part of the stock-in-trade of logic, and because it is a prime function of logic to establish the validity or invalidity of disputed arguments.

(Modal) Logic in Games


Many topics...

- Logics of rational agency
- Logics of rational interaction
- Game Logics
- When are two games the same?
- Epistemic program in game theory
- Social Choice Theory and Logic
- (Formally) Verifying that a social procedure is correct
- Develop ("well-behaved") logical languages that can express game theoretic concepts, such as the Nash equilibrium
Games for Logic

A can force \( \{p\}, \{q, r\} \),

\( E \) can force \( \{p, q\}, \{p, r\} \).
A can force \{p\}, \{q, r\}, E can force \{p, q\}, \{p, r\}
A can force \{p\}, \{q, r\}, E can force \{p, q\}, \{p, r\}
A can force \{p\}, \{q, r\}, E can force \{p, q\}, \{p, r\}
Games for Logic

A can force \{p\}, \{q, r\}, E can force \{p, q\}, \{p, r\}
Games for Logic

A can force \{p\}, \{q, r\}, E can force \{p, q\}, \{p, r\}
Games for Logic

$p \land (q \lor r) \iff (p \land q) \lor (p \land r)$
A primer on game theoretic models (extensive/normal form games)
1. a group of *self-interested* agents (players) involved in some interdependent decision problem
1. a group of *self-interested* agents (players) involved in some interdependent decision problem
1. a group of *self-interested* agents (players) involved in some interdependent decision problem
Game Situations

1. a group of *self-interested* agents (players) involved in some interdependent decision problem
1. a group of *self-interested* agents (players) involved in some interdependent decision problem
### Game Situations

1. A group of *self-interested* agents (players) involved in some interdependent decision problem, and

<table>
<thead>
<tr>
<th></th>
<th>Ann</th>
<th>Bob</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>U</strong></td>
<td>1,1</td>
<td>0,0</td>
</tr>
<tr>
<td><strong>D</strong></td>
<td>0,0</td>
<td>1,1</td>
</tr>
</tbody>
</table>
Game Situations

1. a group of *self-interested* agents (players) involved in some interdependent decision problem, and
2. the players *recognize that they are engaged in a game situation*

<table>
<thead>
<tr>
<th></th>
<th>U</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ann</td>
<td>1,1</td>
<td>0,0</td>
</tr>
<tr>
<td>Bob</td>
<td>0,0</td>
<td>1,1</td>
</tr>
</tbody>
</table>
What should Ann (Bob) do?

Game Situations

<table>
<thead>
<tr>
<th>Ann</th>
<th>Bob</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>L</td>
</tr>
<tr>
<td>U</td>
<td>1,1</td>
</tr>
<tr>
<td>D</td>
<td>0,0</td>
</tr>
</tbody>
</table>

It depends on what she expects Bob to do, but this depends on what she thinks Bob expects her to do, and so on...
What should Ann (Bob) do?

It depends on what she expects Bob to do, but this depends on what she thinks Bob expects her to do, and so on...
A game is a description of strategic interaction that includes:

- Actions the players can take
- Description of the players' interests (i.e., preferences)
- Description of the "structure" of the decision problem

It does not specify the actions that the players do take.
A game is a description of strategic interaction that includes

- actions the players *can* take
- description of the players’ interests (i.e., preferences),
- description of the “structure” of the decision problem
A game is a description of strategic interaction that includes

- actions the players can take
- description of the players’ interests (i.e., preferences),
- description of the “structure” of the decision problem

*It does not specify the actions that the players do take.*
Solution Concepts

A solution concept is a systematic description of the outcomes that may emerge in a family of games.

This is the starting point for most of game theory and includes many variants: Nash equilibrium, backwards inductions, or iterated dominance of various kinds.

These are usually thought of as the embodiment of “rational behavior” in some way and used to analyze game situations.
A strategic games is a tuple $\langle N, \{A_i\}_{i \in N}, \{\succeq_i\}_{i \in N}\rangle$ where

- $N$ is a finite set of players

- $A_i$ is a nonempty set of actions for each $i \in N$

- $\succeq_i$ is a preference relation on $A = \prod_{i \in N} A_i$ (Often $\succeq_i$ are represented by utility functions $u_i : A \to \mathbb{R}$)
Strategic Games

A strategic games is a tuple \( \langle N, \{A_i\}_{i \in N}, \{\succeq_i\}_{i \in N} \rangle \) where

- \( N \) is a finite set of **players**
- for each \( i \in N \), \( A_i \) is a nonempty set of **actions**
A strategic games is a tuple $\langle N, \{A_i\}_{i \in N}, \{\succeq_i\}_{i \in N} \rangle$ where

- $N$ is a finite set of players
- for each $i \in N$, $A_i$ is a nonempty set of actions
- for each $i \in N$, $\succeq_i$ is a preference relation on $A = \prod_{i \in N} A_i$ (Often $\succeq_i$ are represented by utility functions $u_i : A \rightarrow \mathbb{R}$)
Strategic Games: Comments on Preferences

- Preferences may be over a set of consequences $C$. Assume $g : A \rightarrow C$ and $\{\succeq^*_i \mid i \in N\}$ a set of preferences on $C$. Then for $a, b \in A$,
  
  $$a \succeq_i b \text{ iff } g(a) \succeq^*_i g(b)$$

- Consequences may be affected by exogenous random variable whose realization is not known before choosing actions. Let $\Omega$ be a set of states, then define $g : A \times \Omega \rightarrow C$. Where $g(a|\cdot)$ is interpreted as a lottery.

- Often $\succeq_i$ are represented by utility functions $u_i : A \rightarrow \mathbb{R}$
Strategic Games: Example

- \( N = \{ \text{Row}, \text{Column} \} \)
- \( A_{\text{Row}} = \{ u, d \}, \ A_{\text{Column}} = \{ r, l \} \)
- \( (u, r) \preceq_{\text{Row}} (d, l) \preceq_{\text{Row}} (u, l) \sim_{\text{Row}} (d, r) \)
- \( (u, r) \preceq_{\text{Column}} (d, l) \preceq_{\text{Column}} (u, l) \sim_{\text{Column}} (d, r) \)
Strategic Games: Example

<table>
<thead>
<tr>
<th></th>
<th>Column</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>r</td>
</tr>
<tr>
<td><strong>u</strong></td>
<td>(2,2)</td>
</tr>
<tr>
<td><strong>d</strong></td>
<td>(0,0)</td>
</tr>
</tbody>
</table>

- $N = \{\text{Row}, \text{Column}\}$
- $A_{\text{Row}} = \{u, d\}$, $A_{\text{Column}} = \{r, l\}$
- $u_{\text{Row}} : A_{\text{Row}} \times A_{\text{Column}} \rightarrow \{0, 1, 2\}$, $u_{\text{Column}} : A_{\text{Row}} \times A_{\text{Column}} \rightarrow \{0, 1, 2\}$ with $u_{\text{Row}}(u, r) = u_{\text{Column}}(u, r) = 2$, $u_{\text{Row}}(d, l) = u_{\text{Column}}(d, l) = 2$, and $u_x(u, l) = u_x(d, r) = 0$ for $x \in N$. 

$\rightarrow$
Nash Equilibrium

Let $\langle N, \{A_i\}_{i \in N}, \{\succeq_i\}_{i \in N} \rangle$ be a strategic game.

For $a_{-i} \in A_{-i}$, let

$$B_i(a_{-i}) = \{ a_i \in A_i \mid (a_{-i}, a_i) \succeq_i (a_{-i}, a'_i) \ \forall \ a'_i \in A_i \}$$

$B_i$ is the best-response function.
Nash Equilibrium

Let $\langle N, \{A_i\}_{i \in N}, \{\succeq_i\}_{i \in N}\rangle$ be a strategic game.

For $a_{-i} \in A_{-i}$, let

$$B_i(a_{-i}) = \{a_i \in A_i \mid (a_{-i}, a_i) \succeq_i (a_{-i}, a'_i) \ \forall \ a'_i \in A_i\}$$

$B_i$ is the best-response function.

$a^* \in A$ is a Nash equilibrium iff $a^*_i \in B_i(a^*_{-i})$ for all $i \in N$. 

Strategic Games Example: Bach or Stravinsky?

<table>
<thead>
<tr>
<th></th>
<th>$b_c$</th>
<th>$s_c$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$b_r$</td>
<td>2,1</td>
<td>0,0</td>
</tr>
<tr>
<td>$s_r$</td>
<td>0,0</td>
<td>1,2</td>
</tr>
</tbody>
</table>
Strategic Games Example: Bach or Stravinsky?

<table>
<thead>
<tr>
<th></th>
<th>$b_c$</th>
<th>$s_c$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$b_r$</td>
<td>2,1</td>
<td>0,0</td>
</tr>
<tr>
<td>$s_r$</td>
<td>0,0</td>
<td>1,2</td>
</tr>
</tbody>
</table>

$N = \{r, c\}$, $A_r = \{b_r, s_r\}$, $A_c = \{b_c, s_c\}$
Strategic Games Example: Bach or Stravinsky?

\[
\begin{array}{c|cc}
 & b_c & s_c \\
\hline
b_r & 2,1 & 0,0 \\
s_r & 0,0 & 1,2 \\
\end{array}
\]

\[N = \{r, c\} \quad A_r = \{b_r, s_r\}, A_c = \{b_c, s_c\}\]

\[B_r(b_c) = \{b_r\} \quad B_r(s_c) = \{s_r\}\]
Strategic Games Example: Bach or Stravinsky?

<table>
<thead>
<tr>
<th></th>
<th>$b_c$</th>
<th>$s_c$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$b_r$</td>
<td>2,1</td>
<td>0,0</td>
</tr>
<tr>
<td>$s_r$</td>
<td>0,0</td>
<td>1,2</td>
</tr>
</tbody>
</table>

$N = \{r, c\}$  $A_r = \{b_r, s_r\}$, $A_c = \{b_c, s_c\}$

$B_r(b_c) = \{b_r\}$  $B_r(s_c) = \{s_r\}$

$B_c(b_r) = \{b_c\}$  $B_c(s_r) = \{s_c\}$
Strategic Games Example: Bach or Stravinsky?

\[
\begin{array}{c|cc}
   & b_c & s_c \\
\hline
b_r & 2,1 & 0,0 \\
s_r & 0,0 & 1,2 \\
\end{array}
\]

\[N = \{r, c\} \quad A_r = \{b_r, s_r\}, A_c = \{b_c, s_c\}\]

\[B_r(b_c) = \{b_r\} \quad B_r(s_c) = \{s_r\}\]

\[B_c(b_r) = \{b_c\} \quad B_c(s_r) = \{s_c\}\]

\((b_r, b_c)\) is a Nash Equilibrium \quad \(s_r, s_c\) is a Nash Equilibrium
Another Example: Pure Coordination

<table>
<thead>
<tr>
<th></th>
<th>Ann</th>
<th>Bob</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>U</strong></td>
<td>1,1</td>
<td>0,0</td>
</tr>
<tr>
<td><strong>D</strong></td>
<td>0,0</td>
<td>1,1</td>
</tr>
</tbody>
</table>
Another Example: Hi-Low

<table>
<thead>
<tr>
<th></th>
<th>Bob</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Ann</td>
<td>L</td>
<td>R</td>
</tr>
<tr>
<td>U</td>
<td>3,3</td>
<td>0,0</td>
</tr>
<tr>
<td>D</td>
<td>0,0</td>
<td>1,1</td>
</tr>
</tbody>
</table>
Reasoning about (strategic) games
Reasoning *about* (strategic) games

There is Kripke structure “built in” a strategic game.

\[ W = \{ \sigma \mid \sigma \text{ is a strategy profile: } \sigma \in \prod_{i \in N} S_i \} \]

<table>
<thead>
<tr>
<th></th>
<th>a</th>
<th>b</th>
<th>c</th>
</tr>
</thead>
<tbody>
<tr>
<td>d</td>
<td>(2,3)</td>
<td>(2,2)</td>
<td>(1,1)</td>
</tr>
<tr>
<td>e</td>
<td>(0,2)</td>
<td>(4,0)</td>
<td>(1,0)</td>
</tr>
<tr>
<td>f</td>
<td>(0,1)</td>
<td>(1,4)</td>
<td>(2,0)</td>
</tr>
</tbody>
</table>
Reasoning about (strategic) games

\[ \sigma \sim_i \sigma' \iff \sigma_i = \sigma'_i: \text{ this epistemic relation represents player } i\text{'s “view of the game” at the } \textit{ex interim} \text{ stage where } i\text{'s choice is fixed but the choices of the other players’ are unknown} \]

<table>
<thead>
<tr>
<th></th>
<th>a</th>
<th>b</th>
<th>c</th>
</tr>
</thead>
<tbody>
<tr>
<td>d</td>
<td>(2,3)</td>
<td>(2,2)</td>
<td>(1,1)</td>
</tr>
<tr>
<td>e</td>
<td>(0,2)</td>
<td>(4,0)</td>
<td>(1,0)</td>
</tr>
<tr>
<td>f</td>
<td>(0,1)</td>
<td>(1,4)</td>
<td>(2,0)</td>
</tr>
</tbody>
</table>
Reasoning about (strategic) games

\( \sigma \approx_i \sigma' \) iff \( \sigma_-^i = \sigma_-^i \): this relation of “action freedom” gives the alternative choices for player \( i \) when the other players’ choices are fixed.

<table>
<thead>
<tr>
<th></th>
<th>( a )</th>
<th>( b )</th>
<th>( c )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( d )</td>
<td>(2,3)</td>
<td>(2,2)</td>
<td>(1,1)</td>
</tr>
<tr>
<td>( e )</td>
<td>(0,2)</td>
<td>(4,0)</td>
<td>(1,0)</td>
</tr>
<tr>
<td>( f )</td>
<td>(0,1)</td>
<td>(1,4)</td>
<td>(2,0)</td>
</tr>
</tbody>
</table>
Reasoning about (strategic) games

\[ \sigma \succeq_i \sigma' \text{ iff player } i \text{ prefers the outcome } \sigma \text{ at least as much as outcome } \sigma' \]

<table>
<thead>
<tr>
<th></th>
<th>a</th>
<th>b</th>
<th>c</th>
</tr>
</thead>
<tbody>
<tr>
<td>d</td>
<td>(2,3)</td>
<td>(2,2)</td>
<td>(1,1)</td>
</tr>
<tr>
<td>e</td>
<td>(0,2)</td>
<td>(4,0)</td>
<td>(1,0)</td>
</tr>
<tr>
<td>f</td>
<td>(0,1)</td>
<td>(1,4)</td>
<td>(2,0)</td>
</tr>
</tbody>
</table>
Reasoning about strategic games

\[ \mathcal{M} = \langle \mathcal{W}, \{\sim_i\}_{i \in N}, \{\approx_i\}_{i \in N}, \{\succeq_i\}_{i \in N} \rangle \]

- \( \sigma \models [\sim_i] \varphi \) iff for all \( \sigma' \), if \( \sigma \sim_i \sigma' \) then \( \sigma' \models \varphi \).
- \( \sigma \models [\approx_i] \varphi \) iff for all \( \sigma' \), if \( \sigma \approx_i \sigma' \) then \( \sigma' \models \varphi \).
- \( \sigma \models \langle \succeq_i \rangle \varphi \) iff there exists \( \sigma' \) such that \( \sigma' \succeq_i \sigma \) and \( \sigma' \models \varphi \).
- \( \sigma \models \langle \succ_i \rangle \varphi \) iff there is a \( \sigma' \) with \( \sigma' \succeq_i \sigma \), \( \sigma \not\preceq_i \sigma' \), and \( \sigma' \models \varphi \).
Reasoning about strategic games

\[ \mathcal{M} = \langle \mathcal{W}, \{\neg i\}_{i \in \mathcal{N}}, \{\approx i\}_{i \in \mathcal{N}}, \{\succeq i\}_{i \in \mathcal{N}} \rangle \]

- \( \sigma \models [\neg i] \varphi \) iff for all \( \sigma' \), if \( \sigma \sim_i \sigma' \) then \( \sigma' \models \varphi \).
- \( \sigma \models [\approx i] \varphi \) iff for all \( \sigma' \), if \( \sigma \approx_i \sigma' \) then \( \sigma' \models \varphi \).
- \( \sigma \models \langle \succeq i \rangle \varphi \) iff there exists \( \sigma' \) such that \( \sigma' \succeq_i \sigma \) and \( \sigma' \models \varphi \).
- \( \sigma \models \langle \succ i \rangle \varphi \) iff there is a \( \sigma' \) with \( \sigma' \succeq_i \sigma \), \( \sigma \not\succeq_i \sigma' \), and \( \sigma' \models \varphi \).

What is the complete logic of finite games?
Reasoning about strategic games

\[ M = \langle W, \{\sim_i\}_{i \in N}, \{\approx_i\}_{i \in N}, \{\succeq_i\}_{i \in N} \rangle \]

- \( \sigma \models [\sim_i] \varphi \) iff for all \( \sigma' \), if \( \sigma \sim_i \sigma' \) then \( \sigma' \models \varphi \).
- \( \sigma \models [\approx_i] \varphi \) iff for all \( \sigma' \), if \( \sigma \approx_i \sigma' \) then \( \sigma' \models \varphi \).
- \( \sigma \models \langle \succeq_i \rangle \varphi \) iff there exists \( \sigma' \) such that \( \sigma' \succeq_i \sigma \) and \( \sigma' \models \varphi \).
- \( \sigma \models \langle \succ_i \rangle \varphi \) iff there is a \( \sigma' \) with \( \sigma' \succeq_i \sigma \), \( \sigma \not\succeq_i \sigma' \), and \( \sigma' \models \varphi \).

the equivalence \([\sim_i][\approx_i] \varphi \leftrightarrow [\approx_i][\sim_i] \varphi \) is valid on full games
Reasoning about strategic games

\[ M = \langle W, \{\sim_i\}_{i \in N}, \{\approx_i\}_{i \in N}, \{\succeq_i\}_{i \in N} \rangle \]

- \( \sigma \models [\sim_i] \varphi \) iff for all \( \sigma' \), if \( \sigma \sim_i \sigma' \) then \( \sigma' \models \varphi \).
- \( \sigma \models [\approx_i] \varphi \) iff for all \( \sigma' \), if \( \sigma \approx_i \sigma' \) then \( \sigma' \models \varphi \).
- \( \sigma \models \langle \succeq_i \rangle \varphi \) iff there exists \( \sigma' \) such that \( \sigma' \succeq_i \sigma \) and \( \sigma' \models \varphi \).
- \( \sigma \models \langle \succ_i \rangle \varphi \) iff there is a \( \sigma' \) with \( \sigma' \succeq_i \sigma \), \( \sigma \not\succeq_i \sigma' \), and \( \sigma' \models \varphi \).

Can we modally define the Nash Equilibrium?
Reasoning about strategic games

\[ \mathcal{M} = \langle \mathcal{W}, \{\sim_i\}_{i \in N}, \{\approx_i\}_{i \in N}, \{\succeq_i\}_{i \in N} \rangle \]

- \( \sigma \models [\sim_i] \varphi \) iff for all \( \sigma' \), if \( \sigma \sim_i \sigma' \) then \( \sigma' \models \varphi \).
- \( \sigma \models [\approx_i] \varphi \) iff for all \( \sigma' \), if \( \sigma \approx_i \sigma' \) then \( \sigma' \models \varphi \).
- \( \sigma \models \langle \succeq_i \rangle \varphi \) iff there exists \( \sigma' \) such that \( \sigma' \succeq_i \sigma \) and \( \sigma' \models \varphi \).
- \( \sigma \models \langle \succ_i \rangle \varphi \) iff there is a \( \sigma' \) with \( \sigma' \succeq_i \sigma \), \( \sigma \not\succeq_i \sigma' \), and \( \sigma' \models \varphi \).

Can we modally define the Nash Equilibrium? \( \text{Nash} := \bigwedge_{i \in \mathcal{N}} \text{Br}_i \)
Reasoning about strategic games

\[ M = \langle W, \{\sim_i\}_{i \in N}, \{\approx_i\}_{i \in N}, \{\succeq_i\}_{i \in N} \rangle \]

- \( \sigma \models [\sim_i] \varphi \) iff for all \( \sigma' \), if \( \sigma \sim_i \sigma' \) then \( \sigma' \models \varphi \).
- \( \sigma \models [\approx_i] \varphi \) iff for all \( \sigma' \), if \( \sigma \approx_i \sigma' \) then \( \sigma' \models \varphi \).
- \( \sigma \models \langle \succeq_i \rangle \varphi \) iff there exists \( \sigma' \) such that \( \sigma' \succeq_i \sigma \) and \( \sigma' \models \varphi \).
- \( \sigma \models \langle \succ_i \rangle \varphi \) iff there is a \( \sigma' \) with \( \sigma' \succeq_i \sigma \), \( \sigma \not\succeq_i \sigma' \), and \( \sigma' \models \varphi \).

Can we modally define the best response for \( i \)?
Reasoning about strategic games

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$a$</td>
<td>$b$</td>
</tr>
<tr>
<td>$d$</td>
<td>(2,3)</td>
<td>(2,2)</td>
</tr>
<tr>
<td>$e$</td>
<td>(0,2)</td>
<td>(4,0)</td>
</tr>
<tr>
<td>$f$</td>
<td>(0,1)</td>
<td>(1,4)</td>
</tr>
</tbody>
</table>
Reasoning about strategic games

\[ \sigma | = \neg \langle \preceq_i \rangle \top : \text{there is no outcome at least as good as } \sigma \]
Reasoning about strategic games

$$\sigma \models \neg \langle \preceq_i \rangle \top$$: there is no outcome at least as good as $\sigma$
Reasoning about strategic games

\[(d, a) \not\models \neg \langle \geq i \rangle \top: \text{there is no outcome at least as good as } \sigma\]
Reasoning about strategic games

there is no outcome which i can choose that is at least as good
Reasoning about strategic games

\[ \sigma \models \langle \approx_i \cap \succ_i \rangle \varphi \text{ iff there is } \sigma' \text{ such that } \sigma(\approx_i \cap \succ_i)\sigma' \text{ and } \sigma' \models \varphi \]
Reasoning about strategic games

the best response for player $i$ is defined as $\neg \langle \approx_i \cap \succ_i \rangle^T$
Reasoning about extensive games
Reasoning about extensive games

A

B

(1, 0) (2, 3) (1, 5) (3, 1) (4, 4)
Invented by Zermelo, Backwards Induction is an iterative algorithm for “solving” and extensive game.
Reasoning about extensive games

(A, 0)

(2, 3)

(1, 5)

(3, 1)

(4, 4)
Reasoning about extensive games

A

B
(1, 0)
(2, 3)
(1, 5)

B

(4, 4)
(3, 1)
(4, 4)
Reasoning about extensive games
Reasoning about extensive games
Reasoning about extensive games

\begin{figure}
\centering
\begin{tikzpicture}
  \node (A) at (0,0) {A}
  \node (B) at (-1,-1) {B}
  \node (1_0) at (-2,-2) {(1,0)}
  \node (2_3) at (-2,-1) {(2,3)}
  \node (1_5) at (0,-2) {(1,5)}
  \node (4_4) at (1,-2) {(4,4)}
  \node (3_1) at (0,-3) {(3,1)}
  \draw (A) -- (1_0);
  \draw (A) -- (2_3);
  \draw (B) -- (1_5);
  \draw (B) -- (4_4);
\end{tikzpicture}
\end{figure}
Reasoning about extensive games

A

(2, 3)    (1, 5)

(1, 0) (2, 3) (1, 5) (4, 4)

(3, 1) (4, 4)
Reasoning about extensive games

A

(2, 3)  (1, 5)

(1, 0)  (2, 3)  (1, 5)  (4, 4)

(3, 1)  (4, 4)
Reasoning about extensive games
Reasoning about extensive games

[Diagram of an extensive game with nodes labeled A and B, and payoffs (1, 0), (2, 3), (1, 5), (3, 1), (4, 4).]
Reasoning about extensive games

```
\[
\begin{array}{c}
\text{B} \\
(1, 0) \quad (2, 3) \\
\text{A} \\
(1, 5) \quad (4, 4)
\end{array}
\]
```
Characterizing Backwards Induction

For each extensive game form, the strategy profile $\sigma$ is a backward induction solution iff $\sigma$ is played at the root of a tree satisfying the following modal axiom for all propositions $p$ and players $i$:

$$(\text{turn}_i \land \langle \sigma^* \rangle (\text{end} \land p)) \rightarrow [\text{move}_i] \langle \sigma^* \rangle (\text{end} \land \langle \geq_i \rangle p)$$

$move_i = \bigcup_a \text{is an } i\text{-move } a$, $\text{turn}_i$ is a propositional variable saying that it is $i$’s turn to move, and $\text{end}$ is a propositional variable true at only end nodes.

Characterizing Backwards Induction

\[ x \xrightarrow{\sigma} y \xleftarrow{\sigma} z \geq u \xrightarrow{\sigma} v \]
Modal Languages for Games

- \([a \cup b](c \cup d)p\): “for each choice between \(a\) or \(b\) there is a choice between \(c\) or \(d\) ending in a \(p\)-state.”
Modal Languages for Games

- \([a \cup b] \langle c \cup d \rangle p\): “for each choice between \(a\) or \(b\) there is a choice between \(c\) or \(d\) ending in a \(p\)-state.

- Strategies as programs:
  \(\left[\left(\text{turn}_E ?; \sigma\right) \cup \left(\text{turn}_A ?; \tau\right)\right]^* (\text{end} \rightarrow p)\)
Modal Languages for Games

- \([a \cup b]⟨c \cup d⟩p\): “for each choice between \(a\) or \(b\) there is a choice between \(c\) or \(d\) ending in a \(p\)-state.

- Strategies as programs:
  \[\left(\left((\text{turn}_E)?; \sigma\right) \cup (\text{turn}_A?; \tau)\right)^(\text{end} \rightarrow p)\]

- \(□_A ϕ\): “\(A\) has a strategy to ensure that \(ϕ\) is true”
Modal Languages for Games

- \([a \cup b]⟨c \cup d⟩p\): “for each choice between a or b there is a choice between c or d ending in a p-state.”

- Strategies as programs:
  \[\left[\left(\text{turn}_E ?; \sigma\right) \cup \left(\text{turn}_A ?; \tau\right)\right]^*(\text{end} \rightarrow p)\]

- □_A\varphi: “A has a strategy to ensure that \varphi is true”

- Knowledge/beliefs:
Modal Languages for Games

- \([a \cup b]⟨c \cup d⟩p\): “for each choice between \(a\) or \(b\) there is a choice between \(c\) or \(d\) ending in a \(p\)-state.”

- Strategies as programs:
  \[\left[\left(\left(\text{turn}_E\right); σ\right) \cup \left(\text{turn}_A\right); τ\right]\right]^*(\text{end} \rightarrow p)\]

- □ₐ\(ψ\): “\(A\) has a strategy to ensure that \(ψ\) is true”

- Knowledge/beliefs:
  \(K_E(⟨a⟩p \lor ⟨b⟩p), \neg K_E⟨a⟩p \land \neg K_E⟨b⟩p\)
Modal Languages for Games

- \([a \cup b]\langle c \cup d\rangle p\): “for each choice between \(a\) or \(b\) there is a choice between \(c\) or \(d\) ending in a \(p\)-state.”

- Strategies as programs:
  \[([(\text{turn}_E)?; \sigma) \cup (\text{turn}_A ?; \tau)]^*(\text{end} \rightarrow p)\]

- \(\square_A \varphi\): “\(A\) has a strategy to ensure that \(\varphi\) is true”

- Knowledge/beliefs:
  \(K_E(\langle a \rangle p \lor \langle b \rangle p), \neg K_E(\langle a \rangle p \land \neg K_E(\langle b \rangle p)\)
  \(K_A \square_E \varphi\) vs. \(\square_E K_A \varphi\)
Modal Languages for Games

- \([a \cup b] \langle c \cup d \rangle p\): “for each choice between \(a\) or \(b\) there is a choice between \(c\) or \(d\) ending in a \(p\)-state.”

- Strategies as programs:
  \[\left[\left(\left(\text{turn}_E ?; \sigma \right) \cup \left(\text{turn}_A ?; \tau \right)\right)^*\right](\text{end} \rightarrow p)\]

- \(\Box_A \varphi\): “\(A\) has a strategy to ensure that \(\varphi\) is true”

- Knowledge/beliefs:
  \(K_E(\langle a \rangle p \lor \langle b \rangle p), \neg K_E \langle a \rangle p \land \neg K_E \langle b \rangle p\)
  \(K_A \Box_E \varphi\) vs. \(\Box_E K_A \varphi\)

- preferences, ...
Reasoning with games
Let \( P \) be a set of atomic programs and \( At \) a set of atomic propositions.

Formulas of **PDL** have the following syntactic form:

\[ \varphi ::= p | \bot | \neg \varphi | \varphi \lor \psi | [\alpha] \varphi \]

\[ \alpha ::= a | \alpha \cup \beta | \alpha; \beta | \alpha^* | \varphi? \]

where \( p \in At \) and \( a \in P \).

\([\alpha] \varphi \) is intended to mean “after executing the program \( \alpha \), \( \varphi \) is true”
Background: Propositional Dynamic Logic

Semantics: $\mathcal{M} = \langle W, \{R_a \mid a \in P\}, V \rangle$ where for each $a \in P$, $R_a \subseteq W \times W$ and $V : \text{At} \rightarrow \wp(W)$

- $R_\alpha \cup \beta := R_\alpha \cup R_\beta$
- $R_\alpha ; \beta := R_\alpha \circ R_\beta$
- $R_\alpha^* := \bigcup_{n \geq 0} R_\alpha^n$
- $R_\varphi? = \{(w, w) \mid \mathcal{M}, w \models \varphi\}$

$\mathcal{M}, w \models [\alpha] \varphi$ iff for each $v$, if $wR_\alpha v$ then $\mathcal{M}, v \models \varphi$
Background: Propositional Dynamic Logic

1. Axioms of propositional logic

2. $[\alpha](\varphi \rightarrow \psi) \rightarrow ([\alpha]\varphi \rightarrow [\alpha]\psi)$

3. $[\alpha \cup \beta]\varphi \leftrightarrow [\alpha]\varphi \land [\beta]\varphi$

4. $[\alpha; \beta]\varphi \leftrightarrow [\alpha][\beta]\varphi$

5. $[\psi?]\varphi \leftrightarrow (\psi \rightarrow \varphi)$

6. $\varphi \land [\alpha][\alpha^*]\varphi \leftrightarrow [\alpha^*]\varphi$

7. $\varphi \land [\alpha^*](\varphi \rightarrow [\alpha]\varphi) \rightarrow [\alpha^*]\varphi$

8. Modus Ponens and Necessitation (for each program $\alpha$)
Background: Propositional Dynamic Logic

1. Axioms of propositional logic

2. $[\alpha](\varphi \rightarrow \psi) \rightarrow ([\alpha]\varphi \rightarrow [\alpha]\psi)$

3. $[\alpha \cup \beta]\varphi \leftrightarrow [\alpha]\varphi \land [\beta]\varphi$

4. $[\alpha; \beta]\varphi \leftrightarrow [\alpha][\beta]\varphi$

5. $[\psi?]\varphi \leftrightarrow (\psi \rightarrow \varphi)$

6. $\varphi \land [\alpha][\alpha^*]\varphi \leftrightarrow [\alpha^*]\varphi$ (Fixed-Point Axiom)

7. $\varphi \land [\alpha^*](\varphi \rightarrow [\alpha]\varphi) \rightarrow [\alpha^*]\varphi$ (Induction Axiom)

8. Modus Ponens and Necessitation (for each program $\alpha$)
From **PDL** to Game Logic

From **PDL** to Game Logic


**Main Idea:**
In **PDL**: \( w \models \langle \pi \rangle \varphi \): there is a run of the program \( \pi \) starting in state \( w \) that ends in a state where \( \varphi \) is true.

The programs in **PDL** can be thought of as *single player games*. 
From **PDL** to Game Logic


**Main Idea:**

In **PDL**: $w \models \langle \pi \rangle \varphi$: there is a run of the program $\pi$ starting in state $w$ that ends in a state where $\varphi$ is true.

The programs in **PDL** can be thought of as *single player games*.

Game Logic generalized **PDL** by considering two players:

In **GL**: $w \models \langle \gamma \rangle \varphi$: Angel has a *strategy* in the game $\gamma$ to ensure that the game ends in a state where $\varphi$ is true.
From **PDL** to Game Logic

**Consequences of two players:**

\[ \langle \gamma \rangle \phi \] Angel has a strategy in \( \gamma \) to ensure \( \phi \) is true

\[ [\gamma] \phi \] Demon has a strategy in \( \gamma \) to ensure \( \phi \) is true

Either Angel or Demon can win:

\[ \langle \gamma \rangle \phi \lor [\gamma] \neg \phi \] Either Angel or Demon wins

But not both:

\[ \neg (\langle \gamma \rangle \phi \land [\gamma] \neg \phi) \] Not both

Thus, \[ [\gamma] \phi \leftrightarrow \neg \langle \gamma \rangle \neg \phi \] is a valid principle

However, \[ [\gamma] \phi \land [\gamma] \psi \rightarrow [\gamma] (\phi \land \psi) \] is not a valid principle
From **PDL** to Game Logic

**Consequences of two players:**

\[\langle \gamma \rangle \varphi: \text{Angel has a strategy in } \gamma \text{ to ensure } \varphi \text{ is true}\]

\[\lbrack \gamma \rbrack \varphi: \text{Demon has a strategy in } \gamma \text{ to ensure } \varphi \text{ is true}\]
From PDL to Game Logic

**Consequences of two players:**

$\langle \gamma \rangle \varphi$: Angel has a strategy in $\gamma$ to ensure $\varphi$ is true

$[\gamma] \varphi$: Demon has a strategy in $\gamma$ to ensure $\varphi$ is true

Either Angel or Demon can win: $\langle \gamma \rangle \varphi \lor [\gamma] \neg \varphi$
From PDL to Game Logic

Consequences of two players:

\langle \gamma \rangle \varphi: \text{Angel has a strategy in } \gamma \text{ to ensure } \varphi \text{ is true}

[\gamma] \varphi: \text{Demon has a strategy in } \gamma \text{ to ensure } \varphi \text{ is true}

Either Angel or Demon can win: \langle \gamma \rangle \varphi \lor [\gamma] \neg \varphi

But not both: \neg (\langle \gamma \rangle \varphi \land [\gamma] \neg \varphi)
From **PDL** to **Game Logic**

**Consequences of two players:**

\(\langle \gamma \rangle \varphi\): Angel has a strategy in \(\gamma\) to ensure \(\varphi\) is true

\([\gamma] \varphi\): Demon has a strategy in \(\gamma\) to ensure \(\varphi\) is true

Either Angel or Demon can win: \(\langle \gamma \rangle \varphi \lor [\gamma] \neg \varphi\)

But not both: \(\neg (\langle \gamma \rangle \varphi \land [\gamma] \neg \varphi)\)

Thus, \([\gamma] \varphi \leftrightarrow \neg \langle \gamma \rangle \neg \varphi\) is a valid principle
From **PDL** to Game Logic

**Consequences of two players:**

$⟨\gamma⟩\varphi$: Angel has a strategy in $γ$ to ensure $\varphi$ is true

$[γ]\varphi$: Demon has a strategy in $γ$ to ensure $\varphi$ is true

Either Angel or Demon can win: $⟨\gamma⟩\varphi \lor [γ]¬\varphi$

But not both: $¬(⟨\gamma⟩\varphi \land [γ]¬\varphi)$

Thus, $[γ]\varphi \leftrightarrow ¬⟨\gamma⟩¬\varphi$ is a valid principle

However, $[γ]\varphi \land [γ]\psi \rightarrow [γ](\varphi \land \psi)$ is **not** a valid principle
From **PDL** to Game Logic

Reinterpret operations and invent new ones:

- \(?\varphi\): Check whether \(\varphi\) currently holds
- \(\gamma_1; \gamma_2\): First play \(\gamma_1\) then \(\gamma_2\)
- \(\gamma_1 \cup \gamma_2\): Angel choose between \(\gamma_1\) and \(\gamma_2\)
- \(\gamma^*\): Angel can choose how often to play \(\gamma\) (possibly not at all); each time she has played \(\gamma\), she can decide whether to play it again or not.
- \(\gamma^d\): Switch roles, then play \(\gamma\)
- \(\gamma_1 \cap \gamma_2 := (\gamma_1^d \cup \gamma_2^d)^d\): Demon chooses between \(\gamma_1\) and \(\gamma_2\)
- \(\gamma^x := ((\gamma^d)^*)^d\): Demon can choose how often to play \(\gamma\) (possibly not at all); each time he has played \(\gamma\), he can decide whether to play it again or not.
From **PDL** to Game Logic

Reinterpret operations and invent new ones:

- $\text{?}\varphi$: Check whether $\varphi$ currently holds
- $\gamma_1; \gamma_2$: First play $\gamma_1$ then $\gamma_2$
- $\gamma_1 \cup \gamma_2$: Angel choose between $\gamma_1$ and $\gamma_2$
- $\gamma^*$: Angel can choose how often to play $\gamma$ (possibly not at all); each time she has played $\gamma$, she can decide whether to play it again or not.
- $\gamma^d$: Switch roles, then play $\gamma$
- $\gamma_1 \cap \gamma_2 := (\gamma^d_1 \cup \gamma^d_2)^d$: Demon chooses between $\gamma_1$ and $\gamma_2$
- $\gamma^x := ((\gamma^d)^*)^d$: Demon can choose how often to play $\gamma$ (possibly not at all); each time he has played $\gamma$, he can decide whether to play it again or not.
From **PDL** to Game Logic

Reinterpret operations and invent new ones:

- $\Box \varphi$: Check whether $\varphi$ currently holds
- $\gamma_1 ; \gamma_2$: First play $\gamma_1$ then $\gamma_2$
- $\gamma_1 \cup \gamma_2$: Angel choose between $\gamma_1$ and $\gamma_2$
- $\gamma^*$: Angel can choose how often to play $\gamma$ (possibly not at all); each time she has played $\gamma$, she can decide whether to play it again or not.
- $\gamma^d$: Switch roles, then play $\gamma$
- $\gamma_1 \cap \gamma_2 := (\gamma_1^d \cup \gamma_2^d)^d$: Demon chooses between $\gamma_1$ and $\gamma_2$
- $\gamma^x := ((\gamma^d)^*)^d$: Demon can choose how often to play $\gamma$ (possibly not at all); each time he has played $\gamma$, he can decide whether to play it again or not.
From **PDL** to Game Logic

Reinterpret operations and invent new ones:

- $\square \varphi$: Check whether $\varphi$ currently holds
- $\gamma_1; \gamma_2$: First play $\gamma_1$ then $\gamma_2$
- $\gamma_1 \cup \gamma_2$: Angel choose between $\gamma_1$ and $\gamma_2$
- $\gamma^*$: Angel can choose how often to play $\gamma$ (possibly not at all); each time she has played $\gamma$, she can decide whether to play it again or not.
- $\gamma^d$: Switch roles, then play $\gamma$
- $\gamma_1 \cap \gamma_2 := (\gamma_1^d \cup \gamma_2^d)^d$: Demon chooses between $\gamma_1$ and $\gamma_2$
- $\gamma^x := ((\gamma^d)^*)^d$: Demon can choose how often to play $\gamma$ (possibly not at all); each time he has played $\gamma$, he can decide whether to play it again or not.
From **PDL** to Game Logic

**Reinterpret operations and invent new ones:**

- $\psi$: Check whether $\varphi$ currently holds
- $\gamma_1; \gamma_2$: First play $\gamma_1$ then $\gamma_2$
- $\gamma_1 \cup \gamma_2$: Angel choose between $\gamma_1$ and $\gamma_2$
- $\gamma^*$: Angel can choose how often to play $\gamma$ (possibly not at all); each time she has played $\gamma$, she can decide whether to play it again or not.
- $\gamma^d$: Switch roles, then play $\gamma$
- $\gamma_1 \cap \gamma_2 := (\gamma_1^d \cup \gamma_2^d)^d$: Demon chooses between $\gamma_1$ and $\gamma_2$
- $\gamma^x := ((\gamma^d)^*)^d$: Demon can choose how often to play $\gamma$ (possibly not at all); each time he has played $\gamma$, he can decide whether to play it again or not.
From **PDL** to Game Logic

**Reinterpret operations and invent new ones:**

- $\mathcal{L}$: Check whether $\varphi$ currently holds
- $\gamma_1; \gamma_2$: First play $\gamma_1$ then $\gamma_2$
- $\gamma_1 \cup \gamma_2$: Angel choose between $\gamma_1$ and $\gamma_2$
- $\gamma^*$: Angel can choose how often to play $\gamma$ (possibly not at all); each time she has played $\gamma$, she can decide whether to play it again or not.
- $\gamma^d$: Switch roles, then play $\gamma$
- $\gamma_1 \cap \gamma_2 := (\gamma_1^d \cup \gamma_2^d)^d$: Demon chooses between $\gamma_1$ and $\gamma_2$
- $\gamma^x := ((\gamma^d)^*)^d$: Demon can choose how often to play $\gamma$ (possibly not at all); each time he has played $\gamma$, he can decide whether to play it again or not.
From **PDL** to Game Logic

**Reinterpret operations and invent new ones:**

- \(?\varphi\): Check whether \(\varphi\) currently holds
- \(\gamma_1; \gamma_2\): First play \(\gamma_1\) then \(\gamma_2\)
- \(\gamma_1 \cup \gamma_2\): Angel choose between \(\gamma_1\) and \(\gamma_2\)
- \(\gamma^*\): Angel can choose how often to play \(\gamma\) (possibly not at all); each time she has played \(\gamma\), she can decide whether to play it again or not.
- \(\gamma^d\): Switch roles, then play \(\gamma\)
- \(\gamma_1 \cap \gamma_2 := (\gamma_1^d \cup \gamma_2^d)^d\): Demon chooses between \(\gamma_1\) and \(\gamma_2\)
- \(\gamma^x := ((\gamma^d)^*)^d\): Demon can choose how often to play \(\gamma\) (possibly not at all); each time he has played \(\gamma\), he can decide whether to play it again or not.
From **PDL** to Game Logic

**Reinterpret operations and invent new ones:**

- $\Box \varphi$: Check whether $\varphi$ currently holds
- $\gamma_1; \gamma_2$: First play $\gamma_1$ then $\gamma_2$
- $\gamma_1 \cup \gamma_2$: Angel choose between $\gamma_1$ and $\gamma_2$
- $\gamma^*$: Angel can choose how often to play $\gamma$ (possibly not at all); each time she has played $\gamma$, she can decide whether to play it again or not.
- $\gamma^d$: Switch roles, then play $\gamma$
- $\gamma_1 \cap \gamma_2 := (\gamma_1^d \cup \gamma_2^d)^d$: Demon chooses between $\gamma_1$ and $\gamma_2$
- $\gamma^x := ((\gamma^d)^\ast)^d$: Demon can choose how often to play $\gamma$ (possibly not at all); each time he has played $\gamma$, he can decide whether to play it again or not.
Let $\Gamma_0$ be a set of atomic games and $\text{At}$ a set of atomic propositions. Then formulas of Game Logic are defined inductively as follows:

$$
\gamma := g | \varphi? | \gamma;\gamma | \gamma\cup\gamma | \gamma^* | \gamma^d \\
\varphi := \bot | p | \neg\varphi | \varphi \lor \varphi | \langle\gamma\rangle\varphi | [\gamma]\varphi
$$

where $p \in \text{At}$, $g \in \Gamma_0$. 

---

Logic and Artificial Intelligence 31/36
A neighborhood game model is a tuple $\mathcal{M} = \langle W, \{E_g \mid g \in \Gamma_0\}, V \rangle$ where

$W$ is a nonempty set of states

For each $g \in \Gamma_0$, $E_g : W \to \wp(\wp(W))$ is a monotonic neighborhood function.

$X \in E_g(w)$ means in state $s$, Angel has a strategy to force the game to end in some state in $X$ (we may write $wE_gX$)

$V : At \to \wp(W)$ is a valuation function.
Propositional letters and boolean connectives are as usual.

\( \mathcal{M}, w \models \langle \gamma \rangle \varphi \iff \varphi^\mathcal{M} \in E_\gamma(w) \)
Game Logic

Propositional letters and boolean connectives are as usual.

\[ M, w \models \langle \gamma \rangle \varphi \iff (\varphi)^M \in E_\gamma(w) \]

Suppose \( E_\gamma(Y) := \{ s \mid Y \in E_g(s) \} \)

- \( E_{\gamma_1;\gamma_2}(Y) := E_{\gamma_1}(E_{\gamma_2}(Y)) \)
- \( E_{\gamma_1 \cup \gamma_2}(Y) := E_{\gamma_1}(Y) \cup E_{\gamma_2}(Y) \)
- \( E_{\varphi?}(Y) := (\varphi)^M \cap Y \)
- \( E_{\varphi^d}(Y) := E_\gamma(\overline{Y}) \)
- \( E_{\varphi^*}(Y) := \mu X. Y \cup E_\gamma(X) \)
Game Logic: Axioms

1. All propositional tautologies
2. $\langle \alpha; \beta \rangle \varphi \leftrightarrow \langle \alpha \rangle \langle \beta \rangle \varphi$ Composition
3. $\langle \alpha \cup \beta \rangle \varphi \leftrightarrow \langle \alpha \rangle \varphi \lor \langle \beta \rangle \varphi$ Union
4. $\langle \psi ? \rangle \varphi \leftrightarrow (\psi \land \varphi)$ Test
5. $\langle \alpha^d \rangle \varphi \leftrightarrow \neg \langle \alpha \rangle \neg \varphi$ Dual
6. $(\varphi \lor \langle \alpha \rangle \langle \alpha^* \rangle \varphi) \rightarrow \langle \alpha^* \rangle \varphi$ Mix

and the rules,

$\begin{align*}
\varphi & \rightarrow \psi \\
\varphi \rightarrow \psi & \rightarrow \psi \\
\langle \alpha \rangle \varphi & \rightarrow \langle \alpha \rangle \psi \\
(\varphi \lor \langle \alpha \rangle \psi) & \rightarrow \psi \\
\langle \alpha^* \rangle \varphi & \rightarrow \psi
\end{align*}$
Game Logic

- Game Logic is more expressive than PDL
Game Logic

- Game Logic is more expressive than PDL

\[ \langle (g^d)^* \rangle \bot \]
Game Logic

- Game Logic is more expressive than PDL

\[ \langle (g^d)^* \rangle \perp \]

- All GL games are determined.
Game Logic

**Theorem** Dual-free game logic is sound and complete with respect to the class of all game models.
Game Logic

**Theorem** Dual-free game logic is sound and complete with respect to the class of all game models.

**Theorem** Iteration-free game logic is sound and complete with respect to the class of all game models.
Game Logic

Theorem Dual-free game logic is sound and complete with respect to the class of all game models.

Theorem Iteration-free game logic is sound and complete with respect to the class of all game models.

Open Question Is (full) game logic complete with respect to the class of all game models?
