Research Project Proposal: Abstractions in Extensive-Form Games

Jacopo Pio Gargano
jacopopio.gargano@mail.polimi.it
CSE Track
Recreational games

Chess
Recreational games

Chess

1996

MiniMax with alpha-beta pruning search
Recreational games

Go
Recreational games

Go

2015

Monte Carlo tree search
Deep neural networks
Reinforcement Learning - self-play
Physical security

ARMOR - LAX airport

PROTECT - New York City port
Physical security

ARMOR - LAX airport

PROTECT - New York City port

Attacker VS Defender
Limited resources
Find optimal resource allocation
Physical security

Wildlife poaching

Forest protection
Physical security

Wildlife poaching

Forest protection

Reserves are huge
Predict poachers positions
Find optimal resource allocation
Car racing

Defender

Attacker
Military

Attacker

Defender
Military

Attacker

Defender

Dogfighting
Game theory

- Theoretical framework for strategic interaction
- Mathematical models and algorithms (Algorithmic Game Theory)
- Conflict and cooperation
- Intelligent rational decision-makers
- Decisions influencing agents' welfare
Game theory

A game is a process consisting in:

• a set of players

• an initial situation

• rules that players must follow

• all possible final situations - outcomes

• the preferences of all the players - utilities
Game tree representation
Game tree representation

Players

Terminal nodes

Decision nodes

Actions

Utilities
Game tree representation
Game tree representation

Players: Player 1, Player 2
Terminal nodes: 4
Decision nodes: 3
Actions: 6, 19, 32
Utilities: 4, 4
Game tree representation

Players
Player 1
Player 2

Terminal nodes
1.1
1.2
1.3

Decision nodes

Actions

Utilities

4, 4
6, 1
19, 13
2, 14
Game tree representation
Game tree representation

Players: Player 1, Player 2
Terminal nodes: 1.1, 2.1, 1.2, 1.3, 2.1
Decision nodes: 1.1, 2.1
Actions:
Utilities: 4, 4, 6, 19, 32, 14, 2
Game tree representation

Players

Terminal nodes

Decision nodes

Actions

Utilities
Game tree representation

1.1
L₁  R₁

2.1

1.2
L₂  R₂

1.3
L₃  R₃

Player 1
Player 2
Game tree representation

- Players
- Terminal nodes
- Decision nodes
- Actions
- Utilities

<table>
<thead>
<tr>
<th>Player 1</th>
<th>Player 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td></td>
</tr>
<tr>
<td>2.1</td>
<td></td>
</tr>
<tr>
<td>l₁</td>
<td>r₁</td>
</tr>
<tr>
<td>1.2</td>
<td>1.3</td>
</tr>
</tbody>
</table>

Utilities:
- Fragment 4, 4, 6, 19, 32, 14, 2
Game tree representation

Players

L_1

R_1

L_2

R_2

L_3

R_3

Player 1

Player 2

Terminal nodes

Decision nodes

Actions

Utilities

L_1

R_1

l_1

r_1

1.1

2.1

1.2

1.3

4,4

6

19

32

14

2

1
Game tree representation

Players

Terminal nodes

Decision nodes

Actions

Utilities

Player 1

Player 2

Player 1

Player 2

L1

R1

4, 4

L2

R2

L3

R3

6, 1

9, 3

2, 1

4, 2

L1

R1

l1

r1

1.1

2.1

1.2

1.3

4, 4

6

9

19

32

14

2
Formal model

\((N, A, V, T, \nu, \rho, \chi, U)\)
Formal model

\((N, A, V, T, \nu, \rho, \chi, U)\)

Set of players \(\{1, 2\}\)
(Nature can be a player)
Formal model

\[(N, A, V, T, \nu, \rho, \chi, U)\]

Set of players \(\{1, 2\}\)
(Nature can be a player)

Set of actions \(\{L_1, R_1, l_1, r_1, L_2, R_2, \ldots\}\)
Formal model

\((N, A, V, T, \nu, \rho, \chi, U)\)

- Set of players \(\{1, 2\}\) (Nature can be a player)
- Set of actions \(\{L_1, R_1, l_1, r_1, L_2, R_2, \ldots\}\)
- Set of decision nodes \(\{1.1, 1.2, 1.3, 2.1\}\)
Formal model

\[(N, A, V, T, \nu, \rho, \chi, U)\]

Set of players \(\{1, 2\}\)
(Nature can be a player)

Set of actions \(\{L_1, R_1, l_1, r_1, L_2, R_2, \ldots\}\)

Set of decision nodes \(\{1.1, 1.2, 1.3, 2.1\}\)

Set of terminal nodes \(\{t_1, t_2, t_3, t_4, t_5\}\)
Formal model

\[(N, A, V, T, \nu, \rho, \chi, U)\]

- Set of players \{1, 2\}
  (Nature can be a player)
- Set of actions \{L_1, R_1, l_1, r_1, L_2, R_2, \ldots\}
- Set of decision nodes \{1.1, 1.2, 1.3, 2.1\}
- Set of terminal nodes \{t_1, t_2, t_3, t_4, t_5\}

Player function: \(V \rightarrow N\)
Formal model

\[(N, A, V, T, \nu, \rho, \chi, U)\]

- Set of players \(\{1, 2\}\) (Nature can be a player)
- Set of actions \(\{L_1, R_1, l_1, r_1, L_2, R_2, \ldots\}\)
- Set of decision nodes \(\{1.1, 1.2, 1.3, 2.1\}\)
- Set of terminal nodes \(\{t_1, t_2, t_3, t_4, t_5\}\)
- Next node function: \(V \times A \rightarrow V \cup T\)
- Player function: \(V \rightarrow N\)
Formal model

\((N, A, V, T, \nu, \rho, \chi, U)\)

- Set of players \(\{1, 2\}\) (Nature can be a player)
- Set of actions \(\{L_1, R_1, l_1, r_1, L_2, R_2, \ldots\}\)
- Set of decision nodes \(\{1.1, 1.2, 1.3, 2.1\}\)
- Set of terminal nodes \(\{t_1, t_2, t_3, t_4, t_5\}\)

- Action function: \(V \rightarrow 2^A\)
- Next node function: \(V \times A \rightarrow V \cup T\)
- Player function: \(V \rightarrow N\)
Formal model

\[(N, A, V, T, \nu, \rho, \chi, U)\]

- Set of players \(\{1, 2\}\)
  (Nature can be a player)

- Set of actions \(\{L_1, R_1, l_1, r_1, L_2, R_2, \ldots\}\)

- Set of decision nodes \(\{1.1, 1.2, 1.3, 2.1\}\)

- Set of terminal nodes \(\{t_1, t_2, t_3, t_4, t_5\}\)

- Player function: \(V \rightarrow N\)

- Action function: \(V \rightarrow 2^A\)

- Next node function: \(V \times A \rightarrow V \cup T\)

- Set of utility functions: \(\{U_1, U_2\}\)
Perfect vs imperfect information
Perfect vs imperfect information
Perfect vs imperfect information

Perfect information

Imperfect information

Information sets
Strategies

A function associating to each information set a probability distribution over the available actions
Strategies

A function associating to each information set a probability distribution over the available actions.
Strategies

A function associating to each information set a probability distribution over the available actions
Nash equilibrium

A joint combination of strategies, stable w.r.t. unilateral deviations of a single player
Nash equilibrium

A joint combination of strategies, stable w.r.t. unilateral deviations of a single player.
Nash equilibrium

A joint combination of strategies, stable w.r.t. unilateral deviations of a single player
Nash equilibrium

A joint combination of strategies, stable w.r.t. unilateral deviations of a single player
Nash equilibrium

- Mixed strategy NE
- Every n-player finite game has at least one Nash Equilibrium profile in mixed strategies
- epsilon-Nash Equilibrium
State-of-the-art game solving

Linear programming

(Shi, Littman, 2000)
(Billings et al., 2003)
(Gilpin, Sandholm, 2007)
State-of-the-art game solving

No-regret learning

Zinkevich et al., 2008
State-of-the-art huge-game solving: *preplay*
State-of-the-art huge-game solving: *preplay*

>10^{100} \quad \overset{\text{game abstraction}}{\longrightarrow} \quad \sim 10^{12}

original game \quad \text{abstract game}
State-of-the-art huge-game solving: preplay

original game

>10^{100}

game abstraction

abstract game

~10^{12}

abstract game strategies

game solving
State-of-the-art huge-game solving: *preplay*

- Original game strategies (blueprint)
- Game abstraction
- Abstract game strategies
- Game solving
State-of-the-art huge-game solving: *play*
State-of-the-art huge-game solving: play

> $10^{100}$

original game strategies (blueprint)
State-of-the-art huge-game solving: \textit{play}

- Original game strategies (blueprint)
- Finer grain game abstraction with limited lookahead
- Auxiliary game

\[ >10^{100} \]

\[ \sim 10^3 \]
State-of-the-art huge-game solving: play

finer grain game abstraction with limited lookahead

original game strategies (blueprint)

auxiliary game

~10^3

game solving

auxiliary game strategies
State-of-the-art huge-game solving: play

- Original game strategies (blueprint)
- Finer grain game abstraction with limited lookahead
- Strategy mapping
- Auxiliary game strategies
- Auxiliary game
- Game solving

$>10^{100}$

$\sim 10^3$
State-of-the-art huge-game solving: play
State-of-the-art huge-game solving: *play*

- Original game strategies (blueprint)
- Finer grain game abstraction with limited lookahead
- Auxiliary game

$>10^{100}$
State-of-the-art huge-game solving: *play*

Original game strategies (blueprint)

Finer grain game abstraction with limited lookahead

Auxiliary game

Game solving

Auxiliary game strategies

State-of-the-art huge-game solving: $>10^{100}$ to $\sim 10^3$
State-of-the-art huge-game solving: play

original game strategies (blueprint)

finer grain game abstraction with limited lookahead

auxiliary game

strategy mapping

~10^3

auxiliary game strategies

game solving

>10^{100}
State-of-the-art huge-game solving: *play*

$>10^{100}$

original game strategies (blueprint)
State-of-the-art huge-game solving: play

finer grain game abstraction

original game strategies (blueprint)

auxiliary game

~10^3
State-of-the-art huge-game solving: play

original game strategies (blueprint)

finer grain game abstraction

auxiliary game

game solving

abstract game strategies

>10^{100}

~10^3
State-of-the-art huge-game solving: play

- Original game strategies (blueprint)
- Finer grain game abstraction
- Auxiliary game strategies
- Game solving

$>10^{100}$
Game solving

- Optimization problem definition

- Counter Factual Regret minimization plus (CFR+) (Zinkevich et al., *Counter Factual Regret minimization in games with incomplete information*, NeurIPS, 2008)

- Monte Carlo CFR+ (Lanctot et al., *Monte Carlo sampling for regret minimization in extensive games*, NeurIPS, 2009)
Abstractions

• Smaller version of the game capturing the most essential properties of the real domain

• Abstracted game solution provides a useful approximation of the optimal strategy
Abstractions

• Smaller version of the game capturing the most essential properties of the real domain

• Abstracted game solution provides a useful approximation of the optimal strategy

• Lossless information abstractions (Gilpin, Sandoholm, 2007)

• Lossy information abstractions (Gilpin, Sandholm, 2007)
Abstractions

• Information abstractions

• Linear programming and bucketing

• Expectation-based and potential-aware abstractions (Gilpin, Sandholm, 2007)

• Action abstractions

• Actions discretization and game refinement (Brown, Sandholm, 2015)

• Simulation-based abstractions (Tuyls et al. 2018), (Viqueira et al., 2019)
Beyond abstractions

Libratus, 2017

Game Abstraction + MCCFR self-play
Nested subgame solving
Self-improvement
Beyond abstractions

Libratus, 2017
Game Abstraction + MCCFR self-play
Nested subgame solving
Self-improvement

Pluribus, 2019
Extended to 6 players
Simulation-based abstractions

- Bottom-up approach based on data (artificial learning)
- Game traces (observed vs generated)
- Query an oracle for noisy payoff given a strategy
- Model-free
The problem

• Real-world games and strategic scenarios are too large to be represented

• No clear domain-independent abstraction approach was presented to solve these games

• Poker as the main reference application
Our goal

Develop a bottom-up model-free abstraction approach, supported by theoretical guarantees, able to find mixed strategy Nash equilibria in any extensive-form game in a simulation-based fashion.
Idea

<table>
<thead>
<tr>
<th>Original game</th>
<th>Abstract game</th>
</tr>
</thead>
<tbody>
<tr>
<td>Game</td>
<td></td>
</tr>
<tr>
<td>Strategy</td>
<td></td>
</tr>
</tbody>
</table>

- Original game
- Abstract game
- Knowledge base

Game:
- >10/100
- original game
<table>
<thead>
<tr>
<th></th>
<th>Original game</th>
<th>Abstract game</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Game</strong></td>
<td>![Original game diagram]</td>
<td>![Abstract game diagram]</td>
</tr>
<tr>
<td><strong>Strategy</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>traces:</td>
<td>knowledge base:</td>
</tr>
</tbody>
</table>
Idea

<table>
<thead>
<tr>
<th>Game</th>
<th>Original game</th>
<th>Abstract game</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>traces</td>
<td></td>
</tr>
<tr>
<td></td>
<td>knowledge base</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Strategy</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>abstract game</td>
</tr>
<tr>
<td></td>
<td></td>
<td>~10^{12}</td>
</tr>
</tbody>
</table>

domain-independent game abstraction
Idea

Game

Original game

knowledge base

traces

Abstract game

domain-independent game abstraction

~10^{12}

Strategy

game solving (CFR-based)

abstract game

abstract game strategies
Idea

**Game**
- Original game
  - traces
  - knowledge base
- Abstract game
  - domain-independent game abstraction
  - ~10^{12}

**Strategy**
- Original game strategies (blueprint)
- Abstract game strategies
- Game solving (CFR-based)

- Strategy remapping
**Idea**

- **Original game**
  - Game
    - Knowledge base
    - Traces
  - Strategy
    - Blueprint evaluation in original game
    - Original game strategies (blueprint)
    - Strategy remapping

- **Abstract game**
  - Domain-independent game abstraction
  - Abstract game
  - Game solving (CFR-based)
  - Abstract game strategies

- Approximation: \(10^{12}\)
Idea

Blueprint exploitability evaluation
Traces gathering to refine abstraction

Original game

Game

query traces

knowledge base

Abstract game

Strategy

original game strategies (blueprint)

blueprint evaluation in original game
**Idea**

<table>
<thead>
<tr>
<th>Blueprint exploitability evaluation</th>
<th>Original game</th>
<th>Abstract game</th>
</tr>
</thead>
<tbody>
<tr>
<td>Traces gathering to refine abstraction</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Game**

- query traces
- traces

**Strategy**

- blueprint evaluation in original game

**Original game strategies (blueprint)**

- knowledge base

- Traces gathering to refine abstraction
Idea

Original game

Game

Strategy

Abstract game

Blueprint exploitability evaluation
Traces gathering to refine abstraction

query traces

traces

domain-independent game abstraction

knowledge base

abstract game

~10^{12}
Idea

Original game

Abstract game

Game

- Blueprint exploitability evaluation
- Traces gathering to refine abstraction

- query traces
- traces
- domain-independent game abstraction

- knowledge base

Abstract game

- \( \sim 10^{12} \)

- game solving (CFR-based)

Strategy

abstract game strategies
Idea

Blueprint exploitability evaluation
Traces gathering to refine abstraction

Original game

- Game
  - query traces
  - traces

- Strategy
  - original game strategies (blueprint)
  - strategy remapping

Abstract game

- Abstract game
  - domain-independent game abstraction
  - game solving (CFR-based)

- Knowledge base
  - ~10^{12}

Traces gathering to refine abstraction
Idea

Blueprint exploitability evaluation
Traces gathering to refine abstraction

Original game

Game

strategy remapping

Strategies

query traces

traces

knowledge base

domain-independent game abstraction

Abstract game

~10^{12}

Game solving (CFR-based)

strategy remapping

Abstract game strategies

original game strategies (blueprint)

Traces gathering to refine abstraction

Original game

Abstract game

Idea
In a nutshell...

- Real-world strategic conditions are *too large* to be represented and analyzed
- Need for a *domain-independent* way to solve large games
- Exploit *data availability* (traces) and *artificial learning techniques*
In a nutshell...

• Real-world strategic conditions are too large to be represented and analyzed

• Need for a domain-independent way to solve large games

• Exploit data availability (traces) and artificial learning techniques

• How to abstract the game starting from traces?

• How to choose future traces?
Applications

Contract Bridge
Applications

Contract Bridge

Car Racing
Applications

Contract Bridge

Cybersecurity

Car Racing