Artificial Intelligence Opponent for Contested Space (AIOCS)

Dr. (Richard) Scott Erwin, DR-IV
Space Vehicles Directorate
24 Feb 2021

FISO telecon
## Preamble: Game Complexity

<table>
<thead>
<tr>
<th>Game</th>
<th>Board</th>
<th>Tiles/Space</th>
<th>Units (unique)</th>
<th>Abilities/Unit</th>
<th>Units (total)</th>
<th>Moves/Turn</th>
<th>Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tic-tac-Toe</td>
<td>9 (3 x 3)</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>&lt; 4 per side</td>
<td>1</td>
<td>Complete</td>
</tr>
<tr>
<td>Chess</td>
<td>64 (8 x 8)</td>
<td>1</td>
<td>6</td>
<td>16 per side</td>
<td>1</td>
<td>1</td>
<td>Complete</td>
</tr>
<tr>
<td>Go</td>
<td>361 (19 x 19)</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>&lt; 180</td>
<td>1</td>
<td>Complete</td>
</tr>
<tr>
<td>StarCraft 2</td>
<td>16.7M (4k x 4k)</td>
<td>~8</td>
<td>45</td>
<td>2 - 5</td>
<td>&lt; 200 per side</td>
<td>Real-time</td>
<td>Incomplete</td>
</tr>
<tr>
<td>Dota 2</td>
<td>225M (15k x 15k)</td>
<td>?</td>
<td>111</td>
<td>~50</td>
<td>5 per side</td>
<td>Real-time</td>
<td>Incomplete?</td>
</tr>
</tbody>
</table>

*Feb 10, 1996: First Win by Computer against top Human*

*Nov 21, 2005: Last Win by Human against Top Computer*
Go (2016)

- Google DeepMind AlphaGo computer beats 9th Dan Go Champion Lee Sedol in 4 of 5 games March 2016

- Game 4 (Sedol win): Sedol makes what AlphaGo considers a probability zero move; after, AlphaGo cannot recover and its position unravels

- Game 5 (AlphaGo wins): Sedol believes he is winning much of the game; only in endgame realizes he cannot win. Illustrates human vs. non-human playing differences

- Technology: Reinforcement learning – AlphaGo used past games to learn from + played against itself, used resulting win/loss information to build up knowledge base.

- AlphaGo later beat by AlphaStar, which used no past games – learned purely by self-play
“AlphaStar is an intriguing and unorthodox player – one with the reflexes and speed of the best pros but strategies and a style that are entirely its own. The way AlphaStar was trained, with agents competing against each other in a league, has resulted in gameplay that’s unimaginably unusual; it really makes you question how much of StarCraft’s diverse possibilities pro players have really explored.”

10^{26} actions per agent per time step possible
all races; restricted actions/minute to human levels, human camera view of map

Now better than 99.8% of human players
Poker (30 Aug 2019)

- Brown & Sandhom (Facebook AI, CMU) developed Pluribus engine
- Learned by play against 5 copies of itself
- Defeated poker professional Darren Elias, who holds the record for most World Poker Tour titles, and Chris "Jesus" Ferguson, winner of six World Series of Poker events.

"It was incredibly fascinating getting to play against the poker bot and seeing some of the strategies it chose" said Gagliano. "There were several plays that humans simply are not making at all,

Though poker is an incredibly complicated game, Pluribus made efficient use of computation...Pluribus computed its blueprint strategy in eight days using only 12,400 core hours and used just 28 cores during live play.
What’s next (July 2020): Diplomacy

*Diplomacy* is an American strategic board game created by Allan B. Calhamer in 1954 and released commercially in the United States in 1959.[1] Its main distinctions from most board wargames are its negotiation phases (players spend much of their time forming and betraying alliances with other players and forming beneficial strategies) and the absence of dice and other game elements that produce random effects.

(Wikipedia)

"Diplomacy has seven players and focuses on building alliances, negotiation, and teamwork in the face of uncertainty about other agents. As a result, agents have to constantly reason about who to cooperate with and how to coordinate actions," said Tom Eccles, a research engineer at DeepMind.

"Diplomacy is NOT a game about cooperation – it is a game about BETRAYAL. Please stop this exercise before it is too la——”

--Ian Leong, 2020 AFRL Scholar
Status as of October 2020

**Difficulty of Various Games for Computers**

- **Easy**:
  - StarCraft
  - Chess
  - Go

- **Solved Computers Can Play Perfectly**:
  - Shogi (2018)
  - Connect Four (2017)

- **Computers Can Beat Top Humans**:
  - 2011
  - 2016

- **Computers Still Lose to Top Humans** (but progress made could change this)

- **Hard**:
  - Diplomacy

**Complexity (arbitrary units)**

- **DIPLOMACY**
- **STARCAST 2**
- **GO**
- **CHESS**

**Time**

- 2005
- 2010
- 2015
- 2020
- 2025

Appears that as game complexity explodes, delta time to superhuman performance decreases;

Hypothesis: (single) humans get worse as complexity increases

Evidence that human-machine teams (“centaurs”) can beat machines alone

Unclear regarding teams of humans against (teams of?) machines (Dota 2)
AI in Tactical Air Domain: AlphaDogfight

- Contracted R&D Effort @ DARPA
  - Simulator-based guns only dogfight
- Three rounds of AI vs. AI (victor: Heron Systems)
- Last Round: Heron AI vs. “Banger”
  - Banger: AF pilot, NAFB Weapons School grad
  - Full Information Game for AI (had simulator state?)
  - Heron AI splashes Banger 5 out of 5 runs
- Next: AFRL/RQ & DARPA Air Combat Evolution (ACE)
  - AI will be piloting live aircraft (trainer)

“The AI exhibited “superhuman aiming ability” during the simulation, Mock said.”

What is the limiting factor in fighter aircraft agility? Human pilot’s physical limits (g-loads, etc.)

AI Pilots will be flying aerodynamically superior vehicles to anything a human can survive in
Foreign Interest
Our Game: Space Control

• **Develop State-of-Art AI player**
  • For a dynamic, zero-sum contested space game

• **Use Cases**
  • Training Satellite Operators and Operations Center Personnel
  • Understanding what strategy & tactics mean for the new space warfighting domain
  • To validate human-developed tactics, or to test machine-generated tactics, in the space domain

• **Challenges & Opportunities**
  • **C:** New operational domain – no historical data on what works, what doesn’t (no played games)
  • **C:** No “space aces”, game grandmasters, or human experience to draw on
  • **O:** Dominant physics (vice human activity dominated) & good physics models (simulations are realistic)
Summer 2020: M vs. N Game: Space Capture The Flag

**Game Objective:** Pass one of your mobiles through opponents base & return that mobile to your base

**Dynamics:**
- 2-Body Keplarian Orbital Dynamics
- Inverse-Square Gravity ODE’s -> numerical integration in general
- Determines motion of all objects between stages (actions)
- Can simplify for specific orbits (e.g., don’t need to integrate for circular)

**Action Space:**
- Maneuver to one of X pre-specified orbits (sub-GEO to GEO to super-GEO circular)
- Maneuver to intercept adversary agent
- Maneuver to “capture flag” (intercept)
- Maneuver to “return to base” (intercept)

**Constraints:**
- Fuel: Each Mobile Agent has a finite total (cumulative) \( \Delta V \)
  - Once limit is reached, agent can no longer take actions
- Max Per-Thrust Limit (can’t use entire tank in one maneuver)
- Finite Time Duration to win

**Mobile Collisions/Interactions:**
- “Aircraft-like” rules (“from behind” wins, head on collision = both lose)

---

Approved for public release; distribution is unlimited. Public Affairs release approval #AFRL-2021-0496
Reducing Complexity: Discrete State & Action Space Version

**State Space**
- Define finite number of circular orbits (racetracks)
- Example above: 7 orbits: GEO, GEO ± 1000, GEO ± 2000, GEO ± 3000 km
- Super: slower than GEO (drift back); sub: faster than GEO (drift forward)
- Mobiles must be in one of these orbits to take an action
- Avoids need for numerical integration

**Action Space: Slow (Transfers & Intercepts)**
- Can initiate Hohmann x-fer to any other defined racetrack when action allowed
- Can delay Hohmann by an amount up to time of next allowed action
- If opponent mobile will be within interaction distance on arrival – interception
- Else can use to just move to different orbit to allow drifting w/ respect to other orbits/bases/mobiles

**Action Space: Fast (Intercepts only)**
- Can initiate Lambert targeting to any other defined racetrack when action allowed
- Can delay transfer by an amount up to time of next allowed action
- If opponent mobile will be within interaction distance on arrival – interception
- Not allowed for simple orbit x-fers (must use Slow)
Initial Pre-Scripted Opponent (#mobiles = 1 each)
Initial Pre-Scripted Opponent (#mobiles = 1 each)

**First Action**

Agent executes a **DRIFT** play with $\pm \Delta h$

- will start pulling ahead/behind of own station
- will start approaching opponent station & mobiles
Initial Pre-Scripted Opponent (#mobiles = 1 each)

(after some propagation)
Initial Pre-Scripted Opponent (#mobiles = 1 each)

(more propagation)
Initial Pre-Scripted Opponent (#mobiles = 1 each)

**Attack**

Agent gets w/in specified distance of opponent mobile from behind:
- trigger HIT interception to destroy it
- executes a DRIFT play with $\pm \Delta h$ to continue towards opponent base
Initial Pre-Scripted Opponent (#mobiles = 1 each)

“Capture & Return”

Agent gets w/in specified distance of opponent base station:

- trigger HIT interception to enter opponent base zone
- Then trigger DRIFT play with $\pm \Delta h$ to move back to own base
Initial Pre-Scripted Opponent (#mobiles = 1 each)
Initial Pre-Scripted Opponent (#mobiles = 1 each)

Winning

Agent gets w/in specified distance of own base

- trigger HIT interception to enter own zone & win game
Summer 2020 Research: Implementation/Agent/Training

- Implemented game in Python (AIGym compliant)
- Used AC3 DRL agent structure (discrete action/state space)
- Required a significant amount of curriculum training
- Managed to get DRL agents to learn to win against scripted opponent on up to 3 vs. 3 game
- Some caveats:
  - Did not check for collisions/interactions between action times (“teleporting”)
  - Visualization/post-game analysis poor
  - No human player interface (no time)
- Proof of Concept Demonstrated!
Summer 2020 Research: Ensemble DRL Agents

**Ensemble DRL Agent Development & Assessment**
Mr. Sahitya Senapathy, H.S. Senior (→ U. Penn Fall ’20)

- Use $N$ independently trained agents working together to generate the action
  - May have different strengths & weaknesses
  - May have explored different portions of state-action space
- Use a function to select the action to be used from the set of $N$ actions recommended at each step
  - e.g., voting
- Results:
  - Agents that were all trained on exactly the same conditions worked well when blended
  - Agents trained over different conditions ended up with incompatible action recommendations

---

**Agent Reward**

<table>
<thead>
<tr>
<th>Agent</th>
<th>Reward</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ensemble</td>
<td>0.932 / 1</td>
</tr>
<tr>
<td>Agent 1.a</td>
<td>0.976 / 1</td>
</tr>
<tr>
<td>Agent 1.b</td>
<td>0.976 / 1</td>
</tr>
<tr>
<td>Agent 2.c</td>
<td>0.962 / 1</td>
</tr>
<tr>
<td>Agent 2.d</td>
<td>0.958 / 1</td>
</tr>
<tr>
<td>Agent 2.e</td>
<td>0.968 / 1</td>
</tr>
</tbody>
</table>

---

**Agent Reward**

<table>
<thead>
<tr>
<th>Agent</th>
<th>Reward</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ensemble</td>
<td>0.982 / 1</td>
</tr>
<tr>
<td>Agent 1.a</td>
<td>0.976 / 1</td>
</tr>
<tr>
<td>Agent 1.b</td>
<td>0.976 / 1</td>
</tr>
<tr>
<td>Agent 2.c</td>
<td>0.962 / 1</td>
</tr>
<tr>
<td>Agent 2.d</td>
<td>0.958 / 1</td>
</tr>
<tr>
<td>Agent 2.e</td>
<td>0.968 / 1</td>
</tr>
</tbody>
</table>

---

**Output**

---

**Input**
Summer 2020 Research: Agent Analysis & Knowledge Extraction

Human/AI Player Performance Evaluation, Statistical Player Strengths/Weakness Analysis, Explainable AI Tool Evaluation

Ms. Srija Makkapati, Sophomore, Computer Science, Princeton University

• DRL knowledge encoding indecipherable to humans (weights & linkages) in general
  • How to evaluate the ability of players quantitatively?
  • How to know when DRL agent’s skill becomes “superhuman”?

• Looking at a number of input-output characterization methods
  • Elo rating of players (human and computer) to establish game skill based on win-loss rate
  • Match statistics & sub-metrics (e.g., offense, defense, speed)
  • Looking at network perturbation tools (e.g., LIME) to provide further info about DRL players abilities
HRL Results (Neuroevolution Training)

Blue – AI
Red – Scripted

Movie: AlvScripted - 3v3.mp4
HRL Results (Neuroevolution Training)

Blue – AI
Red – AI

Movie: AlvAl – 3v3.mp4
HRL Results (Neuroevolution Training)

Blu – Al
Red – Scripted

Movie: AlvScripted -100v100.mp4
Path Forward/Next Steps

Add Sensing/Tracking Elements (Partial Information)

Add Communication Network (Distributed Decision Making)
Adding the Human: Interface, Visualization, Projection
Thank You

Mr. Loren Anderson (U. MN)
Dr. Deepak Khosla (HRL)
Mr. Mark Skouson (Centauri)
Mr. Edfil Basan (Vermeer)
Mr. Sean Soleyman (HRL)
Mr. Apu Bhopale (AFRL/RV)
Dr. Joel Mozer (USSF/ST)
Dr. Thomas Cooley (AFRL/RV)
Ms. Srija Makkapati (Princeton)
Dr. Michele Gaudreault (USSF/ST)
Mr. Brian Streem (Vermeer)
Mr. Alan Perry (HRL)
Mr. Paul Zetocha (AFRL/RV)
Dr. Jeff Hudack (AFRL/RI)
Ms. Michelle Simon (AFRL/RV)
Dr. Ted Senator (DARPA)
Dr. Joseph Gleason (U. NM)
Dr. Kerianne Hobbs (AFRL/RV)
Mr. Shahin Firouzbakht (Vermeer)
Mr. Sahitya Senapathy (U. Penn)
Dr. Emily Bohner (AFRL/RV)
Dr. David Copp (U. CA Irvine)
Dr. John Ianni (AFRL/RH)
Ms. Michelle Louie (U. NM)

(apologies to anyone I forgot)