A Fictitious Play Approach for Multiplayer Computer Games

Jacek Cyranka

Institute of Informatics
University of Warsaw

jcyranka@gmail.com
cyranka.net

Motivation Slide

1. **Eventual AGI** should not only be able to **compete and win with best Humans**, but also **collaborate with Humans** and **train Humans**;

2. Most of the results of AI in computer games concerned **Human-AI adversary scenario**;

3. We want to efficiently train **AI Agents** for **Multiplayer computer games** in **Collaborative/ Competition Scenario**;

Talk Outline

1. Introduction to Fictitious (Self-)Play.
2. Multi-player Setting.
3. Unity Dodgeball Environments.
4. Human AI-Experiment.
5. Future Challenges.

Fictitious Play & Self-Play

Historical Perspective

Monte-Carlo Methods

1949

Metropolis & Ulam

Fictitious Play Method

1951

G. W. Brown
"Iterative Solutions of Games by Fictitious Play"
J.Robinson
“An iterative method of solving a game”

1950

Dynamic programming & Markovian Decision Processes

1957

R. Bellman

Fictitious Play for Normal-form games

Normal Form Game

	(K,K)	(K,U)	(U,U)	(U,K)
L	3,1	3,1	1,3	1,3
R	2,1	0,0	0,0	2,1

[https://en.wikipedia.org/wiki/Normal-form_game]

Fictitious Play
Process of mixed strategies

$$\boxed{\Pi^i_{t+1}} \in \frac{t-1}{t}\boxed{\Pi^i_t} + \frac{1}{t}\boxed{b^i}\left(\boxed{\Pi^{-i}_t}\right)$$

Mixed strategies of i-th player at t+1 step

Mixed strategies of i-th player at t step

Best response

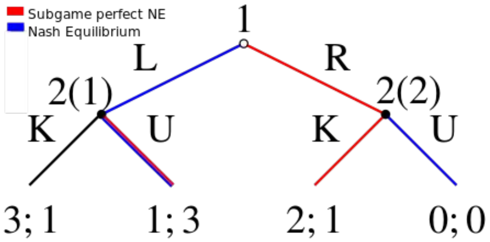
Strategies of other players

Some remarks:

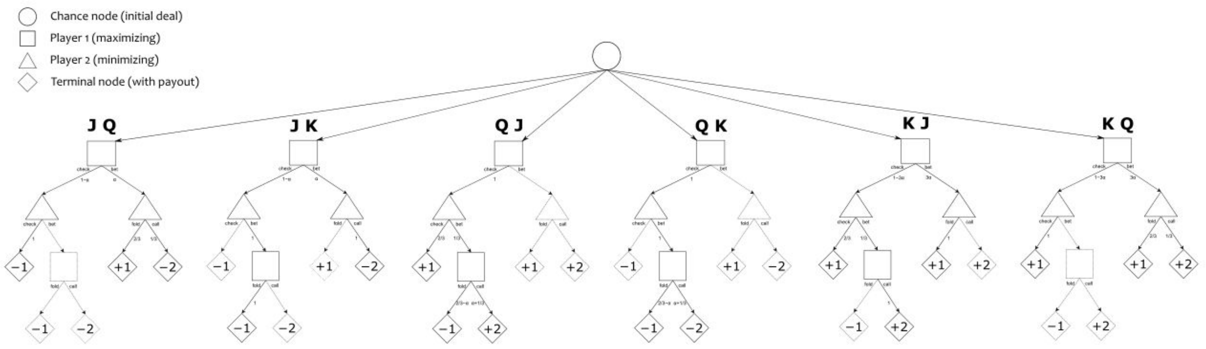
- Name *fictitious* comes from the original work in which the players were ‘imagining’ the opponent play,
- Converges to a *Nash-equilibrium* for two-player games under some assumptions,
- There is a weakened version (ε-best response & perturbation of the player strategies)

Extensive-form games

- More general class of games than normal-form;
- Represented by a rooted tree **(the game tree)** with **player payoffs** at nodes;
- **Chance (nature)** player encoding **probabilistic events** and **imperfect information**;
- Partitioning into equivalence classes **(information sets)**;
- There is an exponential reduction into the normal-form;



	(K,K)	(K,U)	(U,U)	(U,K)
L	3,1	3,1	1,3	1,3
R	2,1	0,0	0,0	2,1



[github.com/Danielhp95/gym-kuhn-poker]

Modern Approach with Supervised and Reinforcement Learning

Curse of Dimensionality in Fictitious Play

FSP is a machine learning framework that implements generalised weakened fictitious play in a sample-based fashion and in behavioural strategies

Vanilla Fictitious Play = Curse of Dimensionality !

Algorithm 1 Full-width extensive-form fictitious play

function FICTITIOUSPLAY(Γ)
  Initialize π_1 arbitrarily
  $j \leftarrow 1$
  while within computational budget **do**
    $\beta_{j+1} \leftarrow$ COMPUTEBRS(π_j)
    $\pi_{j+1} \leftarrow$ UPDATEAVGSTRATEGIES(π_j, β_{j+1})
    $j \leftarrow j + 1$
  end while
  return π_j
end function

Compute the best response for given strategy πj

Update avg strategies for given the new best-responses

[Heinrich, Johannes et al. *"Fictitious Self-Play in Extensive-Form Games." ICML* (2015).]

Fictitious Self-Play

Overcome the curse of dimensionality by applying

reinforcement learning for best response
supervised learning for strategies learning

Algorithm 2 General Fictitious Self-Play

function FICTITIOUSSELFPLAY(Γ, n, m)
  Initialize completely mixed π_1
  $\beta_2 \leftarrow \pi_1$
  $j \leftarrow 2$
  while within computational budget **do**
    $\eta_j \leftarrow$ MIXINGPARAMETER(j)
    $\mathcal{D} \leftarrow$ GENERATEDATA($\pi_{j-1}, \beta_j, n, m, \eta_j$)
    for each player $i \in \mathcal{N}$ **do**
      $\mathcal{M}^i_{RL} \leftarrow$ UPDATERLMEMORY($\mathcal{M}^i_{RL}, \mathcal{D}^i$)
      $\mathcal{M}^i_{SL} \leftarrow$ UPDATESLMEMORY($\mathcal{M}^i_{SL}, \mathcal{D}^i$)
      $\beta^i_{j+1} \leftarrow$ REINFORCEMENTLEARNING($\mathcal{M}^i_{RL}$)
      $\pi^i_j \leftarrow$ SUPERVISEDLEARNING($\mathcal{M}^i_{SL}$)
    end for
    $j \leftarrow j + 1$
  end while
  return π_{j-1}
end function

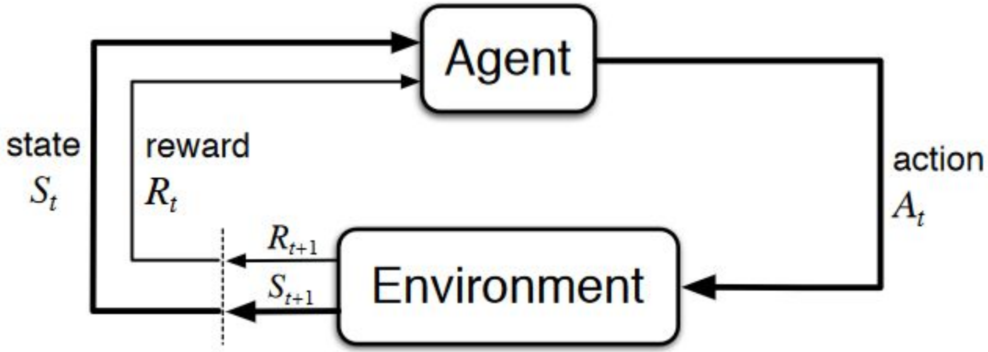
[Heinrich, Johannes et al. *"Fictitious Self-Play in Extensive-Form Games."* *ICML* (2015).]

Reinforcement Learning for best response

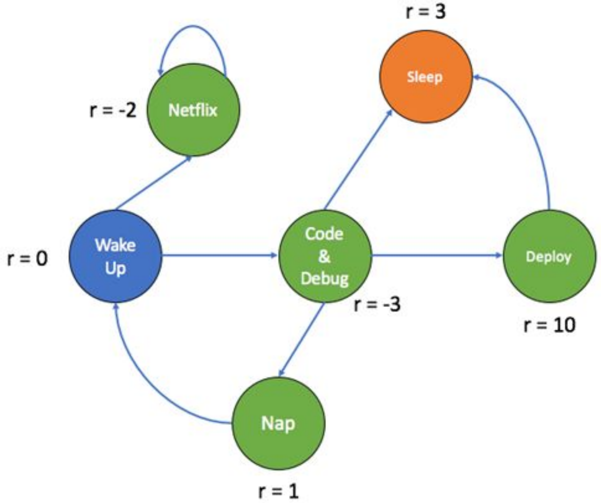
For each player *i* , the (fixed) strategy profile of their opponents π_{-i} defines a MDP.

Player *i* information states define **the states of the MDP**. **The MDP's dynamics** are given by the rules of the extensive-form game, the chance function and the opponents' fixed strategy profile.

The **opponents actions** are performed as **the environment** dynamics.



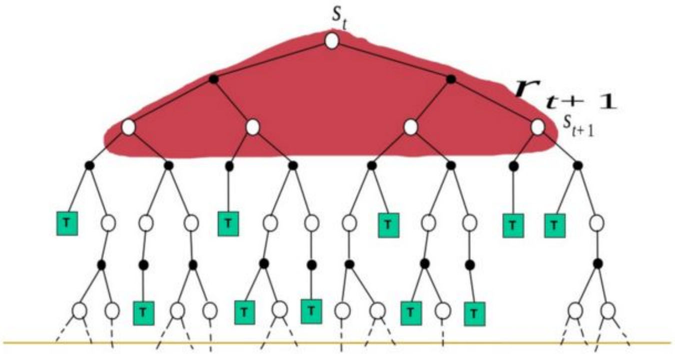
$$R_t = \sum_{t=0}^{\infty} \gamma^t r_t, \text{ where } \gamma \in (0,1) \text{ is called discount factor}$$



Dynamic Programming vs Temporal Difference

Dynamic Programming

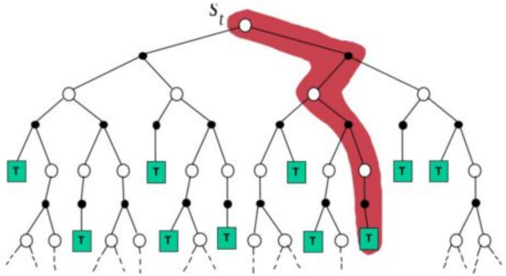
$$V(s_t) \leftarrow E_\pi\{r_{t+1} + \gamma V(s_t)\}$$



Monte Carlo Learning

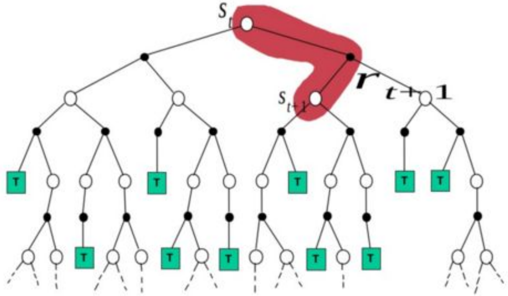
$$V(s_t) \leftarrow V(s_t) + \alpha[R_t - V(s_t)]$$

where R_t is the actual return following state s_t.



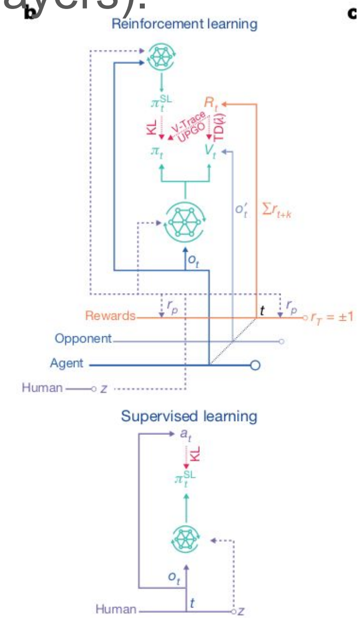
Temporal Difference Learning

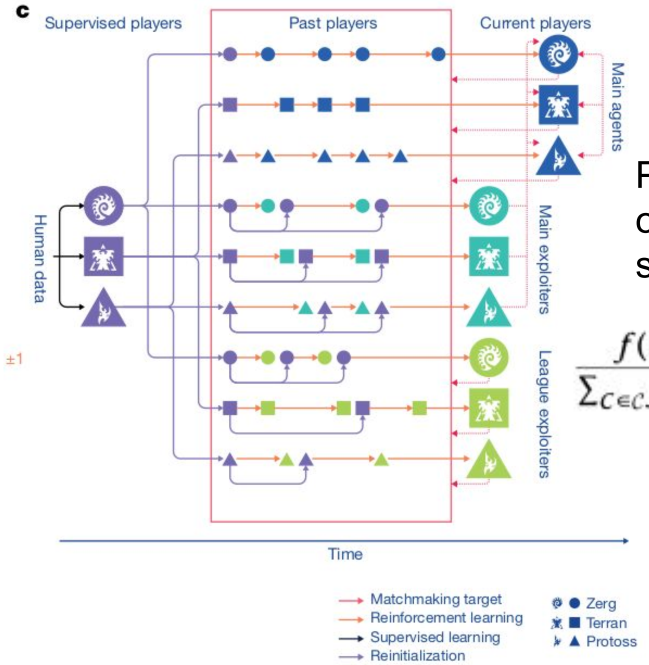
$$V(s_t) \leftarrow V(s_t) + \alpha[r_{t+1} + \gamma V(s_{t+1}) - V(s_t)]$$



Starcraft , RTS game solved by FSP

One of the major superhuman AI instances (above 99.8% of officially ranked human players).





Prioritized opponents sampling

$$\frac{f(\mathbb{P}[A \text{ beats } B])}{\sum_{C\in\mathcal{C}} f(\mathbb{P}[A \text{ beats } C])}$$

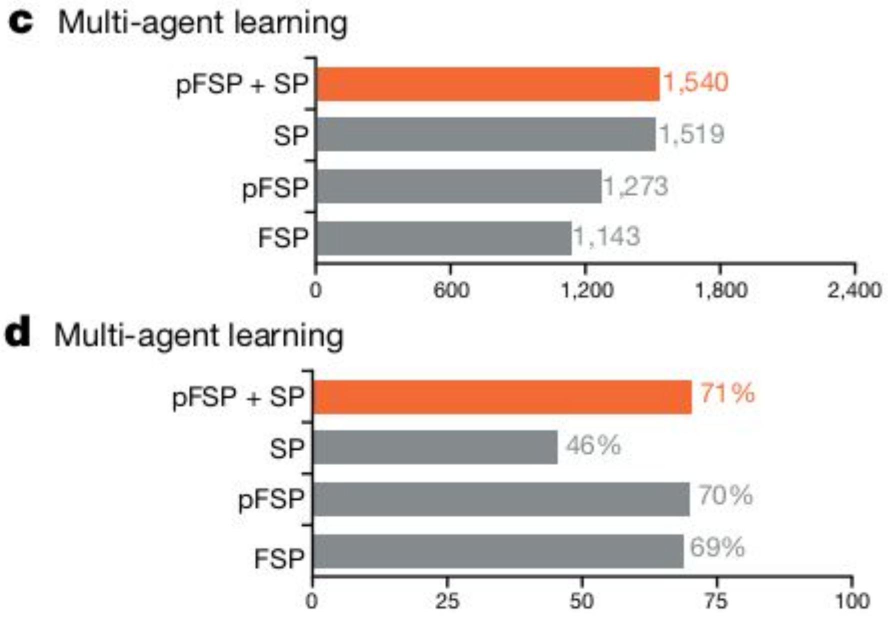
reinforcement learning for best response (sampled opponent checkpoint)

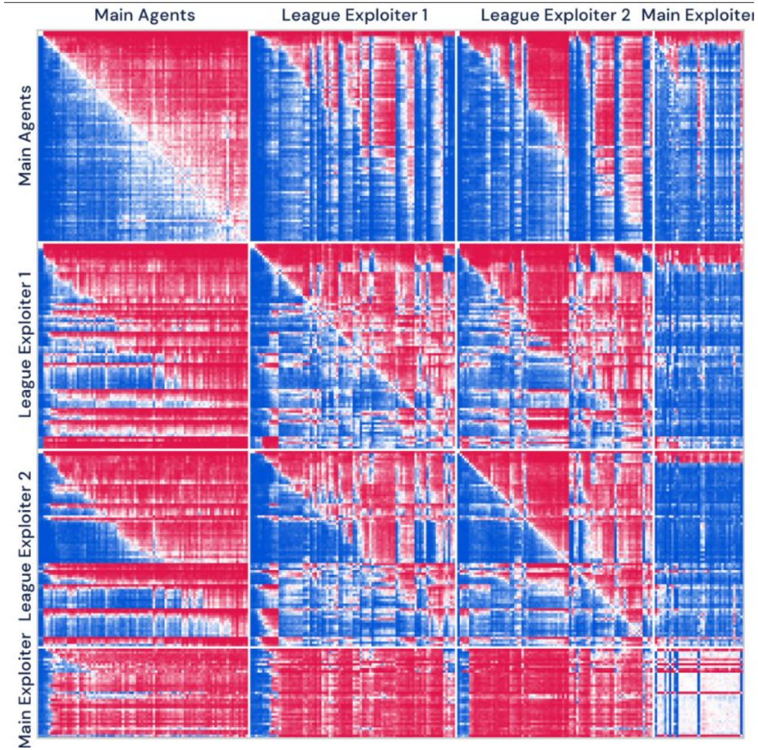
supervised learning for strategies learning

[Vinyals O, Babuschkin I, Czarnecki WM, et al. *Grandmaster level in StarCraft II using multi-agent reinforcement learning. Nature*. 2019;575(7782):350-354. doi:10.1038/s41586-019-1724-z]

FSP helps to achieve an overall best agent

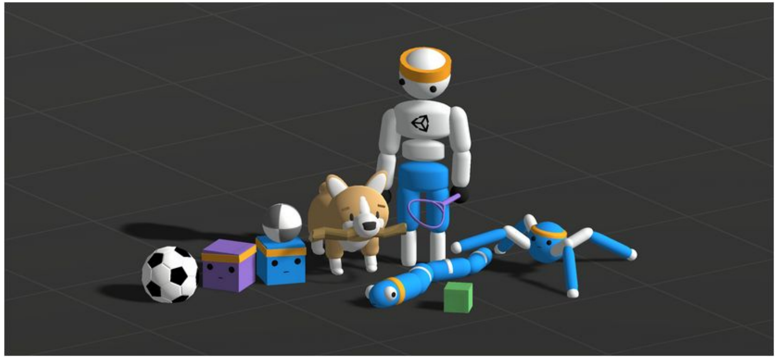
Main message: you should use **Fictitious Self Play** in **Combination** with **Self-Play**





Multiplayer setting
Unity Dodgeball Environments

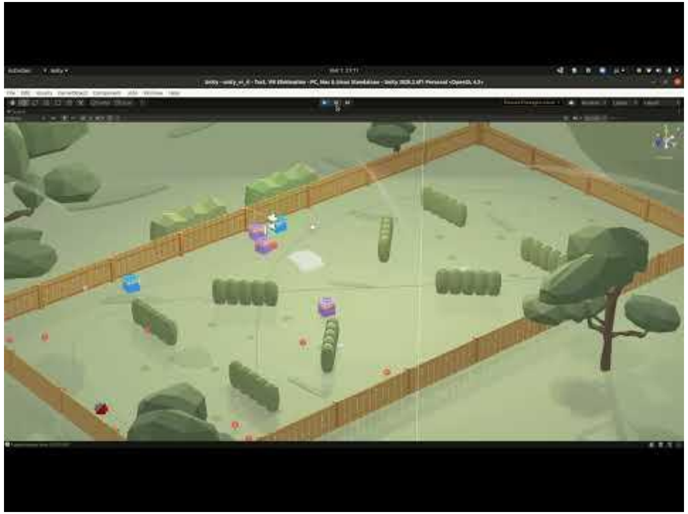
Unity ML-Agents Dodgeball



Unity ML-Agents Toolkit

- (July 12, 2021) ML-Agents plays Dodgeball
- (May 5, 2021) ML-Agents v2.0 release: Now supports training complex cooperative behaviors
- (December 28, 2020) Happy holidays from the Unity ML-Agents team!
- (November 20, 2020) How Eidos-Montréal created Grid Sensors to improve observations for training agents
- (November 11, 2020) 2020 AI@Unity interns shoutout
- (May 12, 2020) Announcing ML-Agents Unity Package v1.0!
- (February 28, 2020) Training intelligent adversaries using self-play with ML-Agents
- (November 11, 2019) Training your agents 7 times faster with ML-Agents
- (October 21, 2019) The AI@Unity interns help shape the world
- (April 15, 2019) Unity ML-Agents Toolkit v0.8: Faster training on real games
- (March 1, 2019) Unity ML-Agents Toolkit v0.7: A leap towards cross-platform inference

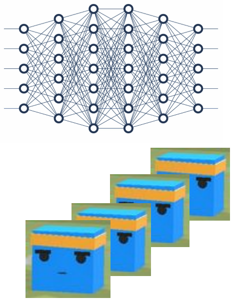
[https://github.com/Unity-Technologies/ml-agents]
[https://blog.unity.com/technology/ml-agents-plays-dodgeball]



Multi-Agent RL for Policy Improvement

SP

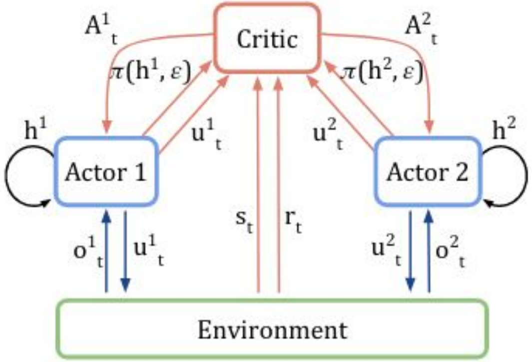
FSP (earlier checkpoints)



sample opponent and perform best response training using MA-RL

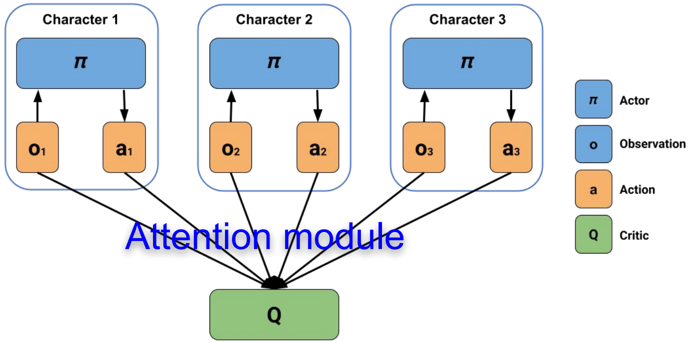
COMA

Decentralized critic architecture



(a)

MA-POCA



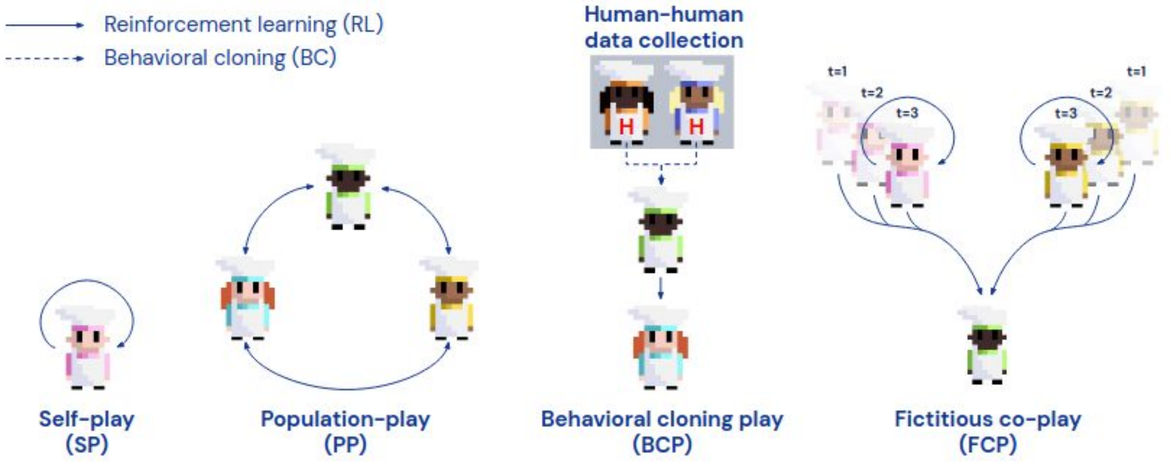
[Cohen, Andrew et al. *"On the Use and Misuse of Absorbing States in Multi-agent Reinforcement Learning." ArXiv* abs/2111.05992 (2021)]
[Foerster, Jakob N. et al. "Counterfactual Multi-Agent Policy Gradients." *ArXiv* abs/1705.08926 (2018)]

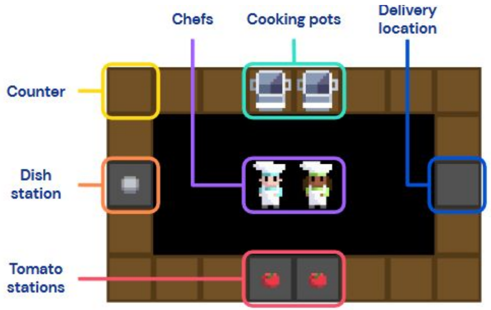
Fictitious **Co-Play**

Disadvantage of **symmetric** training of all collaborating agents: **they learn to play with team-mates at their level**

Algorithm 1: Fictitious Co-Play (FCP)

Input: Number of partners N, checkpoint frequency n_c
`// Stage 1:  train diverse partner populati`
partners = []
for $i = 1$ **to** N **do**
  Initialize agent i.
  $n = 0$ `// step count`
  while *not converged* **do**
    Update agent i in self-play.
    $n \mathrel{+}= 1$
    if $n \mod n_c = 0$ **then**
      Add frozen agent i checkpoint to partners.
`// Stage 2:  train FCP agent`
Filter partners with F.
Initialize FCP agent.
while *not converged* **do**
  Sample partner from partners.
  Update FCP in co-play with partner.

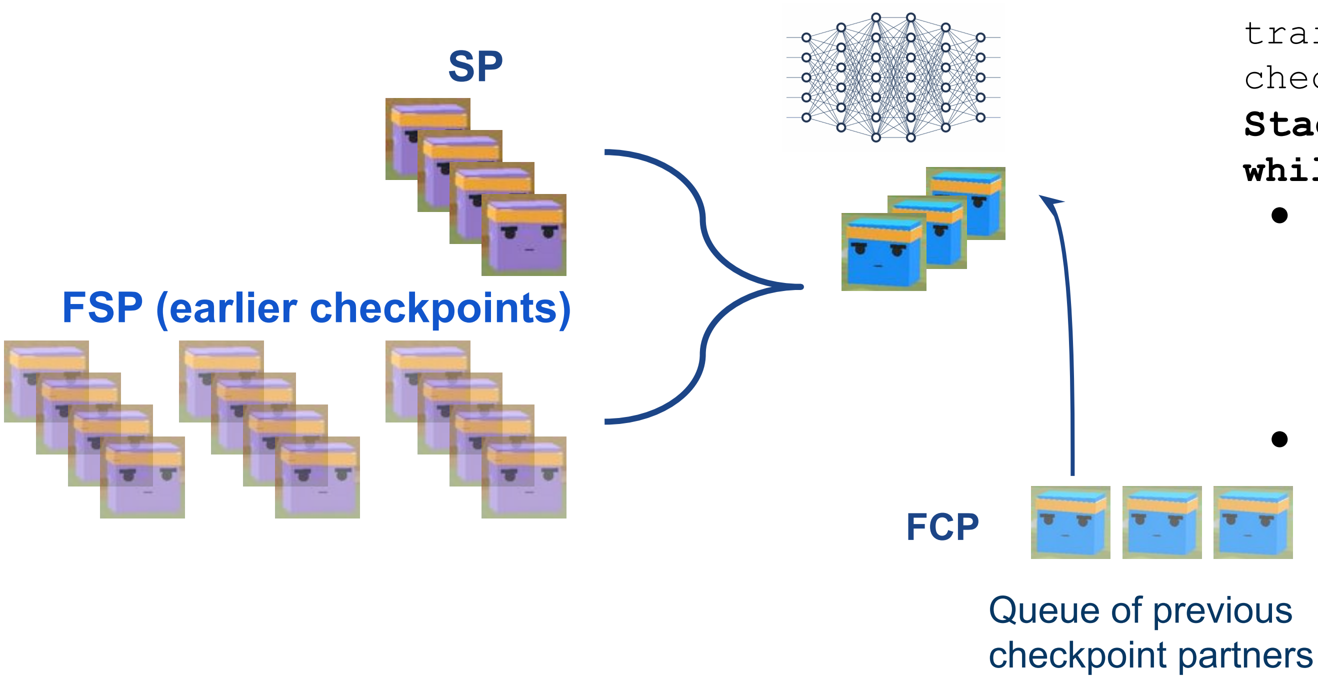




[Strouse, DJ et al. “Collaborating with Humans without Human Data.” *NeurIPS* (2021).]

Our Hybrid approach **Fictitious Co-Self Play**

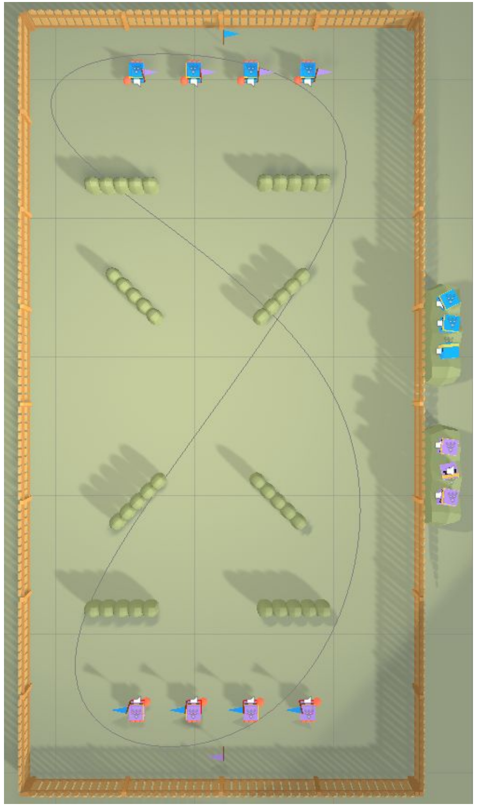
joint work with Jarek Kochanowicz & Witold Szejgis

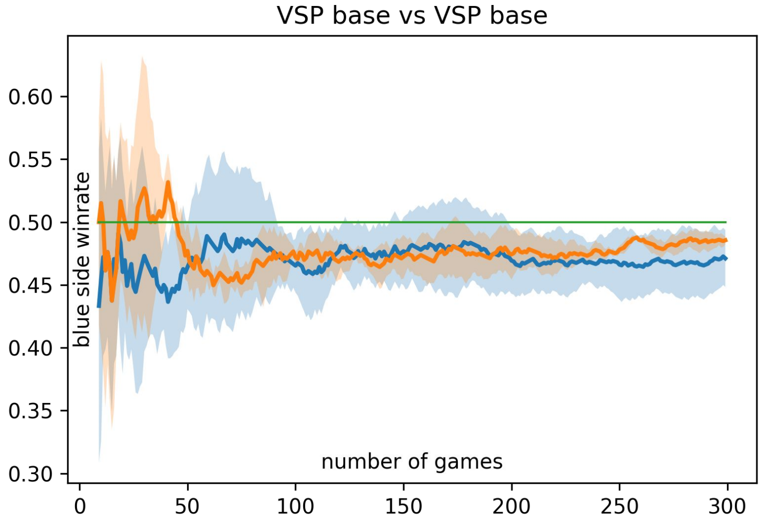


Hybrid Fictitious Co-Self Play Algorithm:

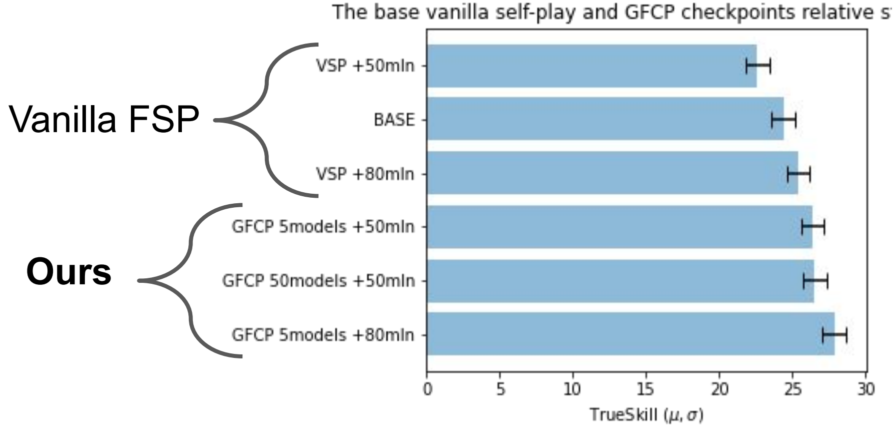
```
Stage I:
train a pool of frozen actor
checkpoints
Stage II:
while not converged:
  ● set an agent(s) in the
    active team to inference
    mode using one of the
    frozen checkpoints in
    Stage I;
  ● train the collaborating
    agents;
```

Subtle asymmetry of the game





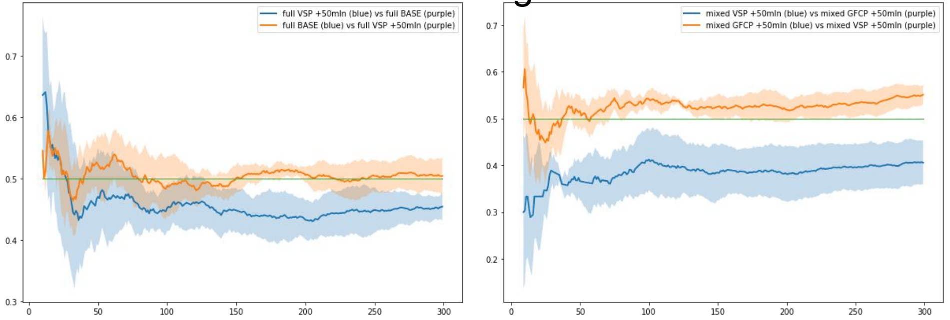
Our Approach vs Vanilla FSP



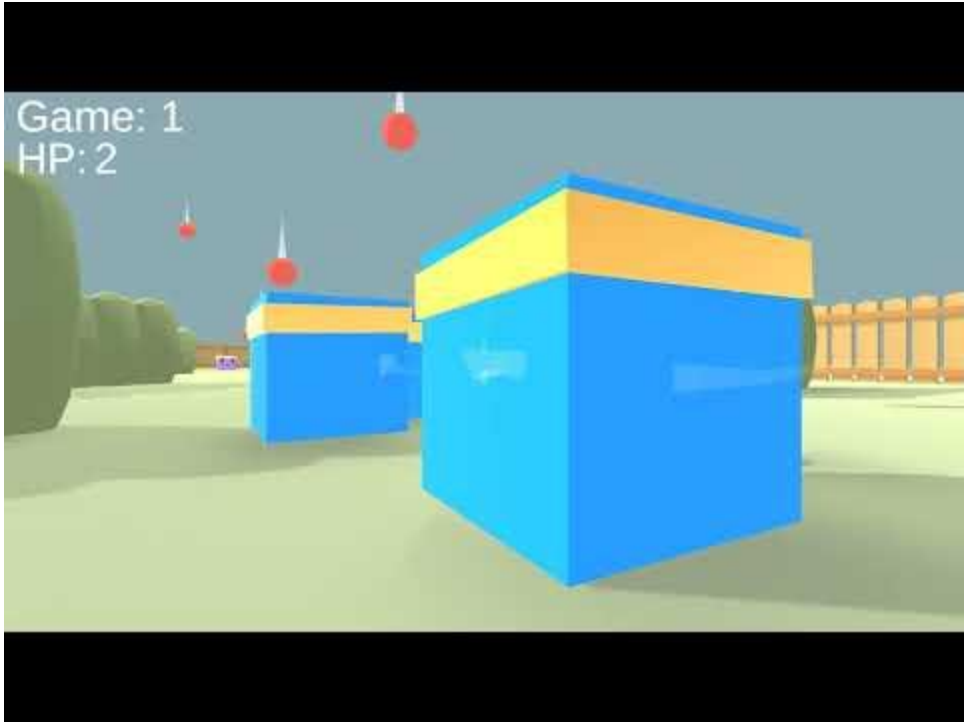
Trueskill team relative strength

team blue	team purple	win % blue	win % purple	std.dev.
+50mln add. training steps over the base, for mixed: 5 frozen checkpoints				
(1)full VSP	**full GFCP**	0.378	0.622	±0.022
full GFCP	full VSP	0.551	0.449	±0.018
(2)mixed VSP	**mixed GFCP**	0.406	0.594	±0.047
mixed GFCP	mixed VSP	0.551	0.449	±0.021
(3a)mixed VSP	full VSP	0.336	0.664	±0.021
full VSP	mixed VSP	0.662	0.338	±0.020
(3b)mixed GFCP	full GFCP	0.403	0.597	±0.015
full GFCP	mixed GFCP	0.541	0.459	±0.013

Blue side winrate
(three independent experiments)
~2000 games



Human as Agent , preliminary experiment in 3D FPP game, ~600 games in total played (22 players)



	AI partner agents	blue winrate	plr. deaths per game
Vanilla	VSP +50mln	$[0.552, 0.705]$	$[0.438, 0.631]$
Ours	GFCP +50mln	$[0.698, 0.83]$	$[0.307, 0.5]$

Table 3: 95% confidence intervals calculated using a sample of 22 human subjects, that played 597 games in total. The confidence intervals are for the two independent sets of samples, i.e. the games of human players matched with three VSP agents, and human players matched with GFCP agents.

statistic(per game)	human player	agent VSP	agent GFCP
deaths	0.469 ± 0.197	0.672	0.558
kills	1.122 ± 0.427	0.650	0.786
accuracy	0.351 ± 0.081	0.247	0.340

Table 4: Comparison of the global per game statistics of the human players against the AI agents (VSP & GFCP) calculated from the full set of games played on the blue team side.

If you are interested in participating in the experiment please sign-up

https://forms.gle/yM4YX9NTx4EhMo6a7

Future Goal - Team Curriculum Learning

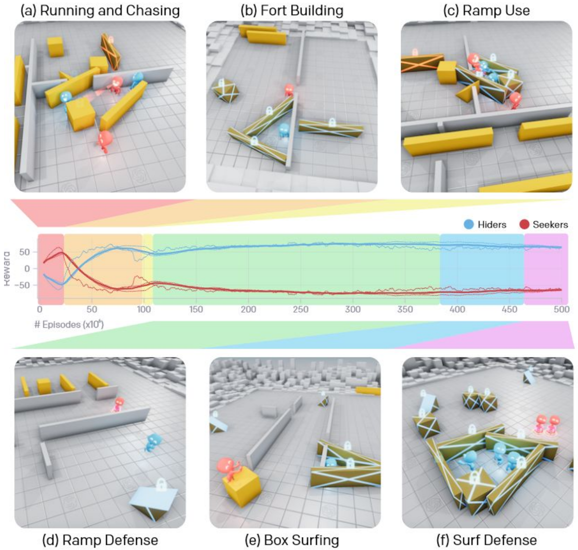
Learning to play Elimination

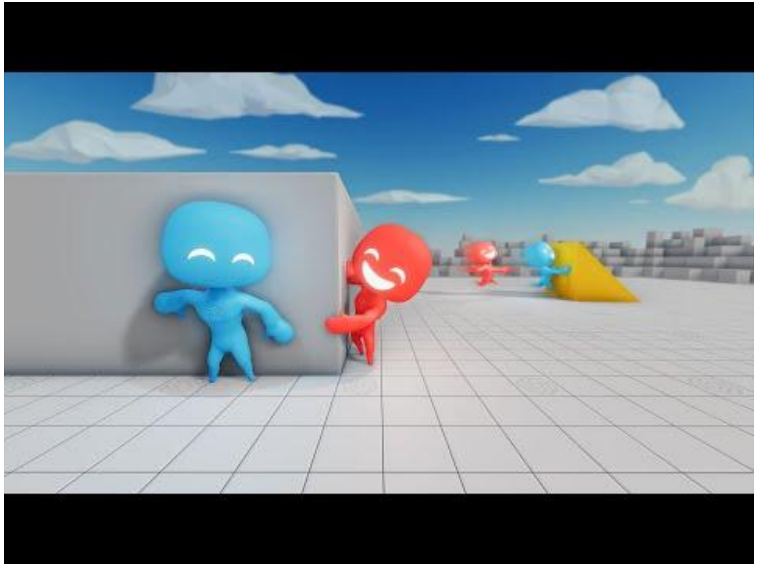
1. Early, learn to shoot but have poor aim and tend to shoot at random.
2. 40 million timesteps, the agents' aim improves,and they still wander randomly
3. 120 million timesteps of training, the agents become much more aggressive and confident and charging into enemy territory as a group.

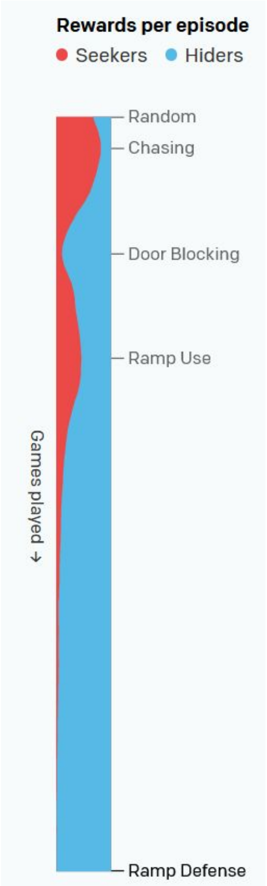
How to play Capture the Flag:

1. 14 million steps, the agents learn to shoot each other, without capturing the flag.
2. 30 million, the agents learn to pick up the enemy flag and return to base,
3. 80 million, the agents exhibit interesting strategies.

Accelerate team curriculum learning from OpenAI Hide&Seek







Thank You for Your Attention!