Master
Reinforcement Learning 2022
Lecture 7:
Multi-Agent

Aske Plaat

liacs Leiden Institute of Advanced Computer Science

Different Approaches

- Model-free
    - Value-based [2,3]
    - Policy-based [4]
- Model-based
    - Learned [5]
    - Perfect; Two-Agent [6]
- Multi-agent [7]
- Hierarchical Reinforcement Learning (Sub-goals) [8]
- Meta Learning [9]

Motivation

Motivation

- Real-world decision making: model interaction

- Poker

- StarCraft

- Football

Overview

- 1: Competitive
- 2: Cooperative
- 3: Mixed

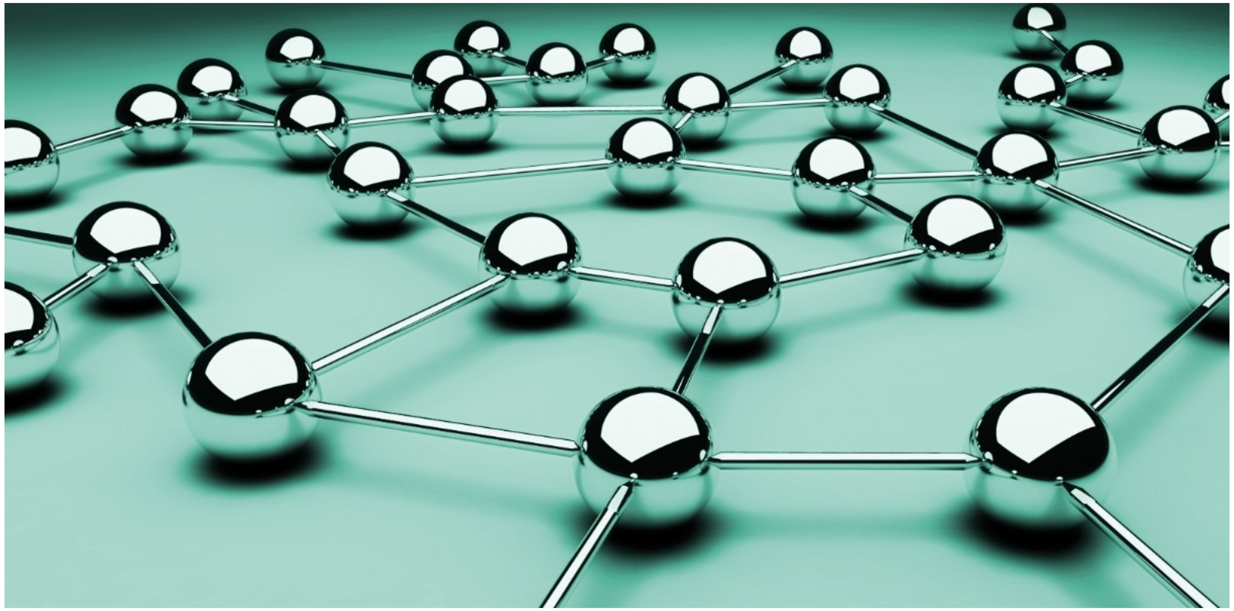
- 1: CFR
- 2: Centr/Decentr, Opponent
- 3: Evo, Swarm, Population Based Teams

- 1: Poker
- 2: Hide and Seek
- 3: Capture The Flag

Social Behavior

- Modeling competition; egoism
- Modeling cooperation; altruism
- Emergent social behavior

Related Fields



- Multi-agent Systems
- Swarm computing, Evolutionary Algorithms
- Complex Networks, Real World Networks

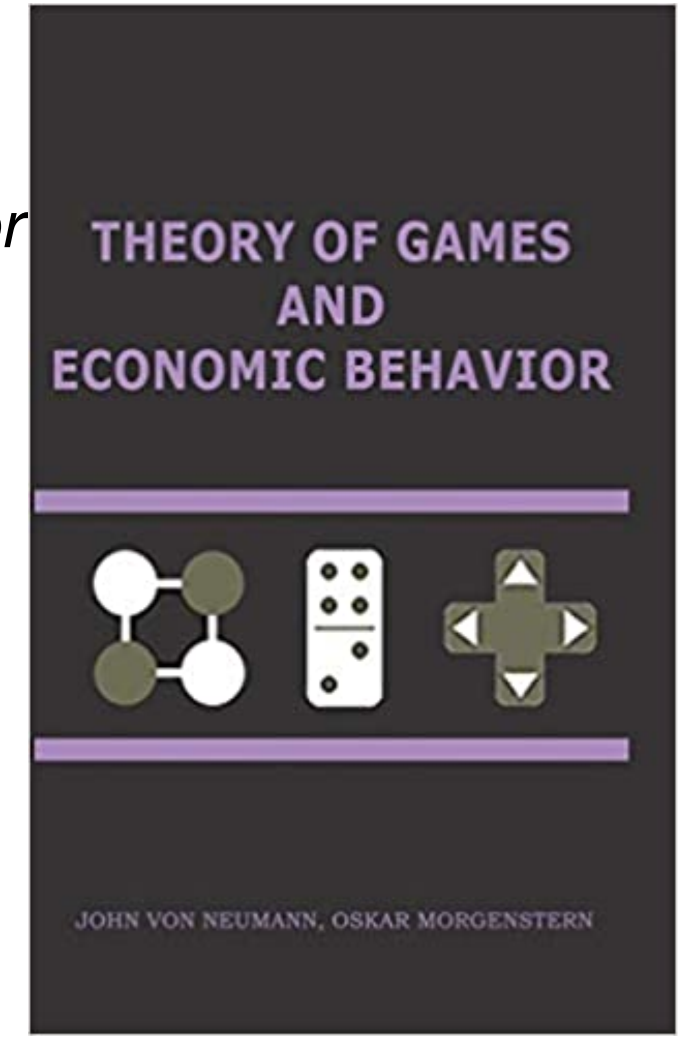
Multi-agent problems

Game Theory

- Von Neumann & Morgenstern, 1944
  Theory of Games and Economic Behavior

- MDP

- Partial information: POMDP



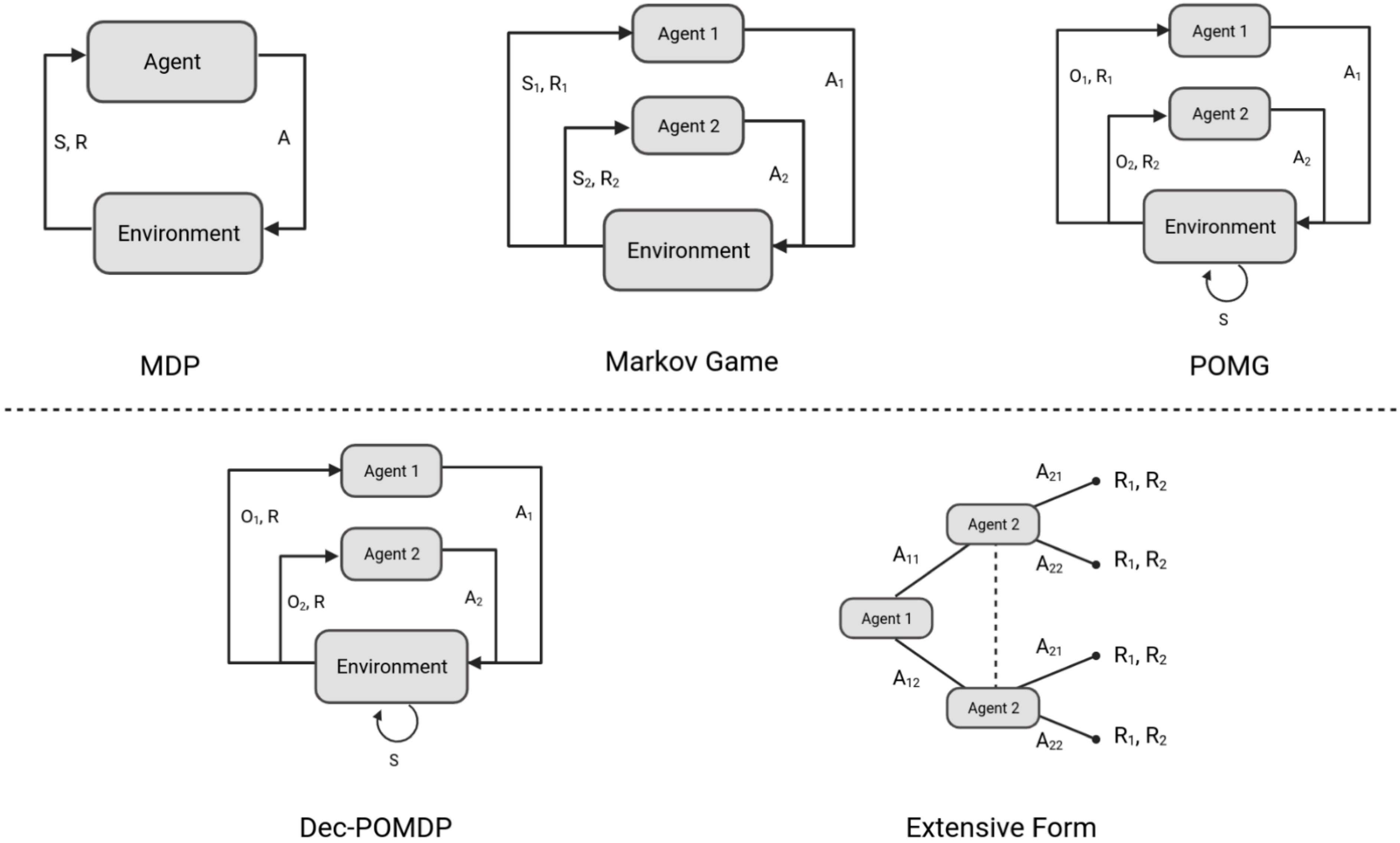
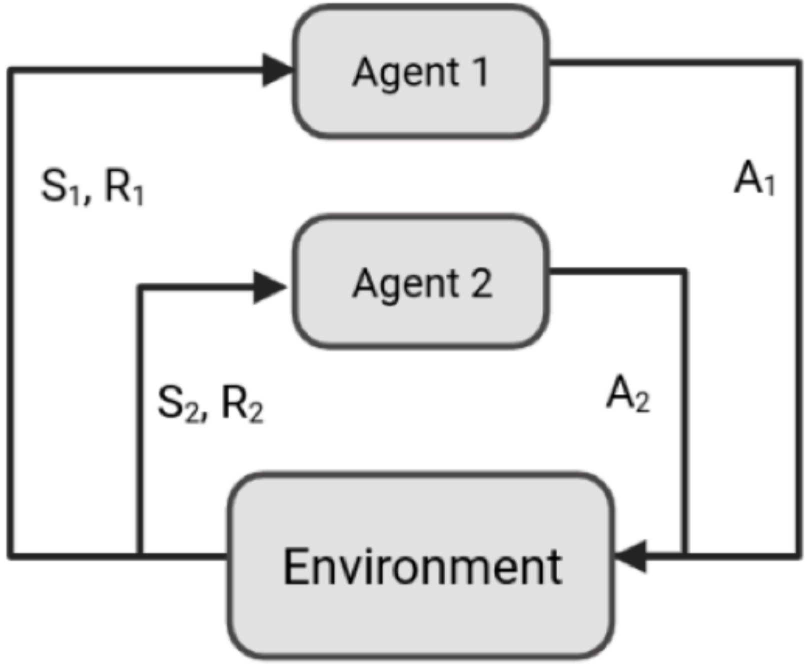


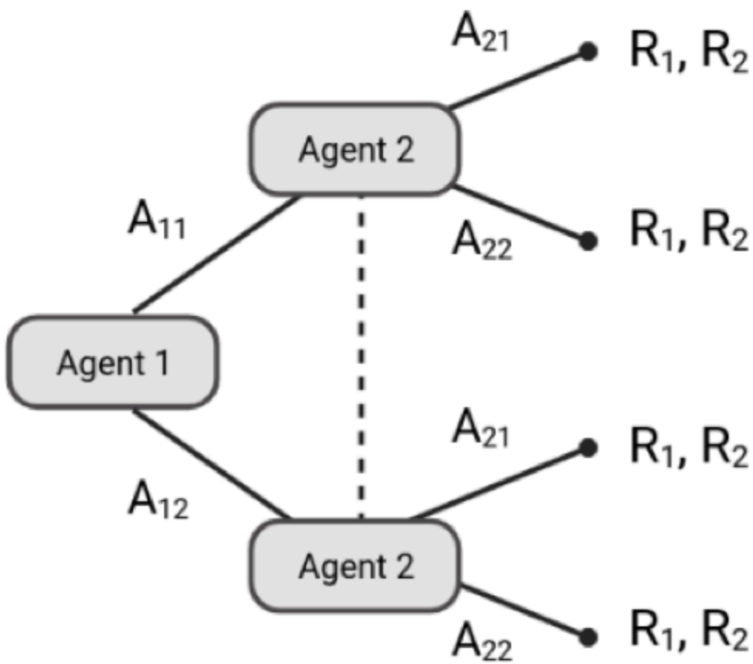
Fig. 2 Visual depiction of the main problem representations in multiagent reinforcement learning The MDP is the primary framework used in the single-agent setting. An agent is in some state S, performs action A, and receives a reward R from the environment. In partially observable environments, the agent cannot view the true state S and receives an observation O instead. For simplicity, all figures display the interaction between two agents $i = 1, 2$ but can be extended to more agents.

Stochastic Games



- Stochastic Games
- Markov Games

Extensive-form Games



- Imperfect information games
- Possible outcomes: information set

Competition

Competition

- Zero sum; win/loss
- John Nash:
  The Nash equilibrium is point $\pi^\star$ from which in a non-collaborative setting none of the agents has any incentive to deviate.
- It is the optimal competitive strategy; each agent chooses best actions for themselves assuming others do the same

Nash equilibrium

- “Multi-agent minimax”
- The Nash-policy for an agent is its best-response strategy
- It is guaranteed to do no worse than tie against any opponent strategy
- For games of imperfect information the Nash equilibrium is an expected outcome

Nash equilibrium

		Firm B		
		Hold down output	Increase output	
Firm A	Hold down output	A gets $1,000 B gets $1,000	A gets $200 B gets $1,500	
	Increase output	A gets $1,500 B gets $200	A gets $400 B gets $400	**Nash**

Counterfactual Regret Minimization

- Multi-agent, partial information, competition
- Algorithm: Counterfactual regret minimization
- Minimize the regret of not having taken the right action, playing many “what-ifs” (counterfactuals)
- CFR is probabilistic multi-agent version of competitive minimax
- Works quite well in Poker
- Complicated code, see paper

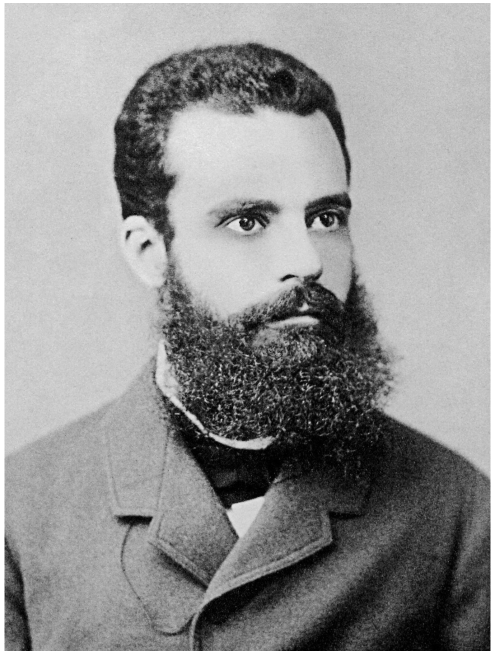
Poker

Pluribus


Genome engineering on a large scale pp. 849 & 922
Monkeyflower research begins to bloom p. 854
Protecting elections from social media manipulation p. 858
Science
$15
30 AUGUST 2019
sciencemag.org
AAAS
76%
PLYR 2
PLYR 6
PLYR 1
PLYR 5
FOLD
PLYR 3
PLYR 4
CALLING OUR
BLUFF
AI masters multiplayer poker
pp. 864 & 885

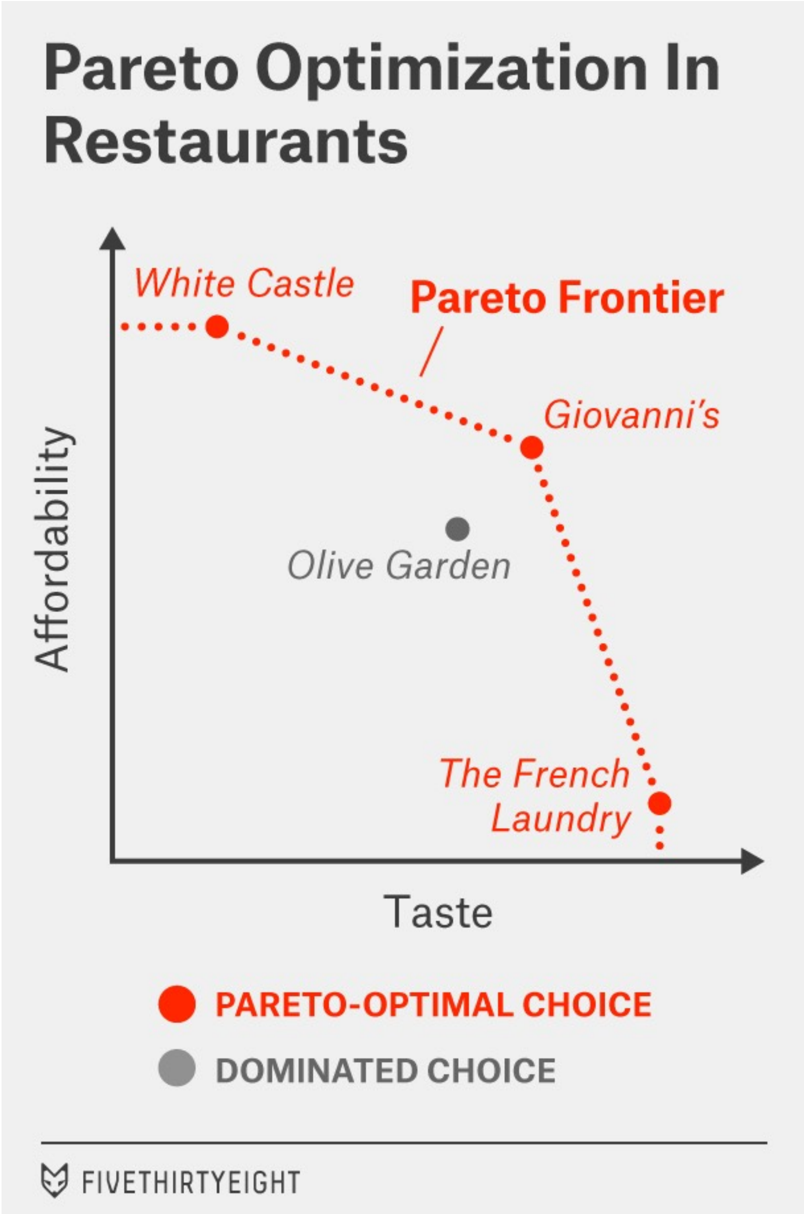
Cooperation

Cooperation



- Non zero sum; win/win
- Vilfredo Pareto
  Pareto front is, in a cooperative setting, the combination of choices where no agent can be better off without at least making one other agent worse off
- It is the optimal cooperative strategy, the best outcome without hurting others.

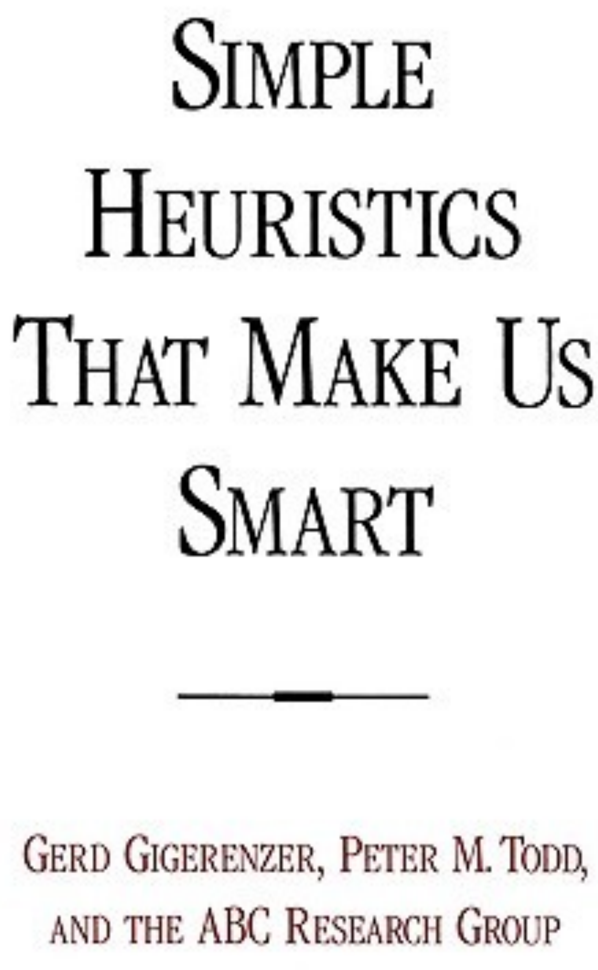
Pareto front

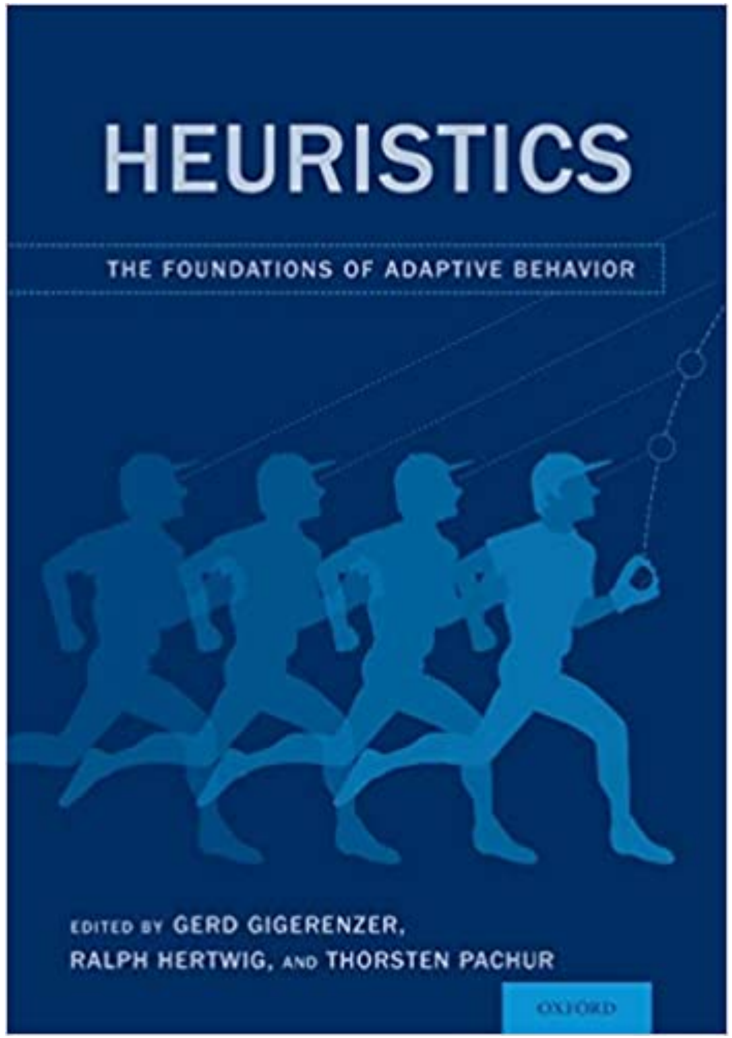


Cooperative Behavior

- Dealing with nonstationarity and partial observability can be done (ignored) by separate training, no communication
- Realism can be improved with Centralized Training/Decentralized Execution -> Centralized controller, or interaction graphs
- Active field of research; overview
  - Value based: VDN, QMIX
  - Policy based: COMA, MADDPG
  - Opponent modeling: DRON, LOLA
  - Communication: Diplomacy game
  - Psychology: Heuristics

Heuristics

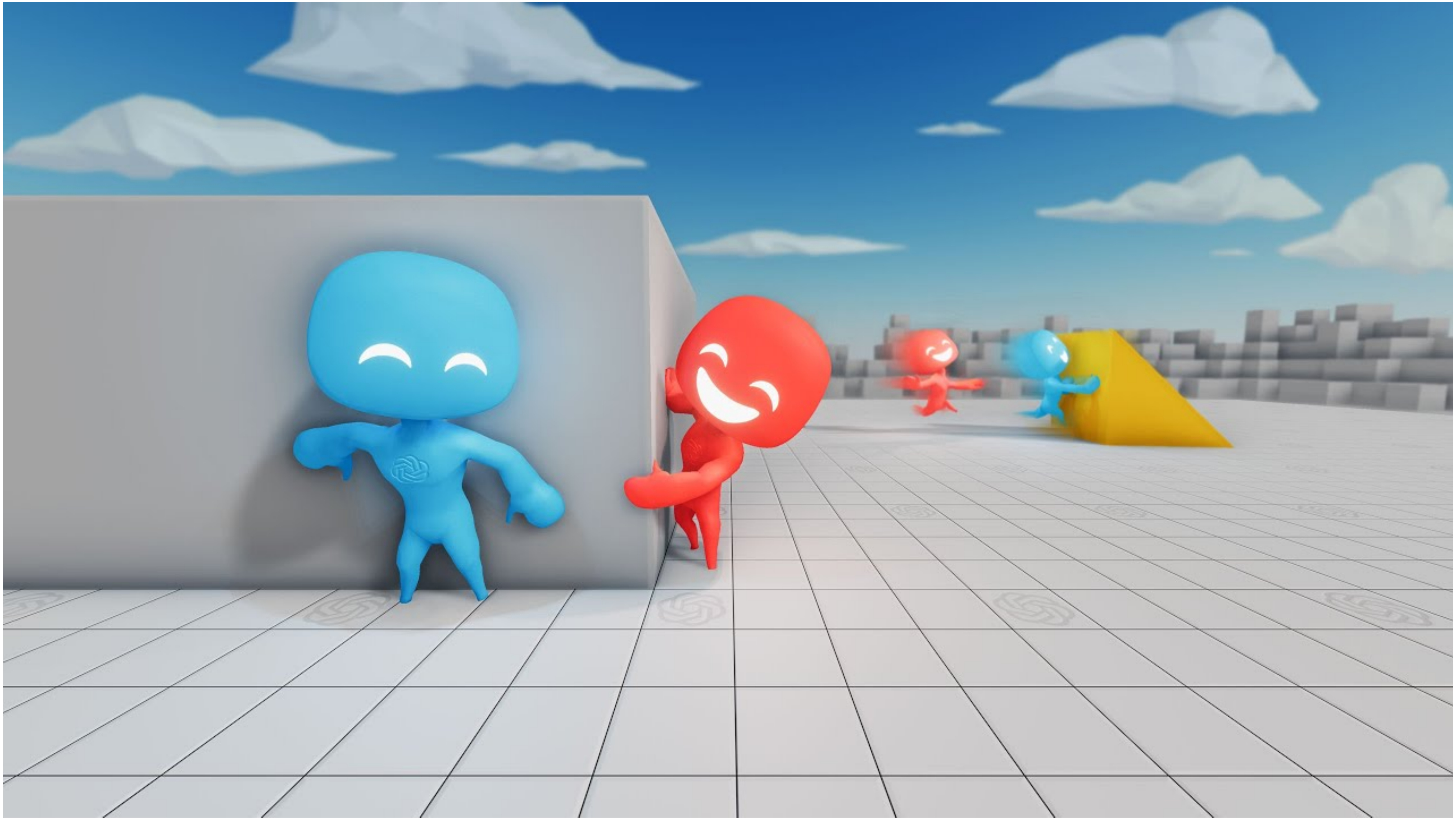

SIMPLE
HEURISTICS
THAT MAKE US
SMART
GERD GIGERENZER, PETER M. TODD,
AND THE ABC RESEARCH GROUP


HEURISTICS
THE FOUNDATIONS OF ADAPTIVE BEHAVIOR
EDITED BY GERD GIGERENZER,
RALPH HERTWIG, AND THORSTEN PACHUR
OXFORD

Emergent Cooperation

- [Baker, 2019]
- The agents can **move** by setting a force on themselves in the x and y directions as well as rotate along the z-axis.
- The agents can **see** objects in their line of sight and within a frontal cone.
- The agents can **sense** distance to objects, walls, and other agents around them using a lidar-like sensor.
- The agents can **grab and move** objects in front of them.
- The agents can **lock** objects in place. Only the team that locked an object can unlock it.

Hide and Seek



Mixed

- Prisoner’s dilemma
- Iterated prisoner’s dilemma
- Emerging social norms

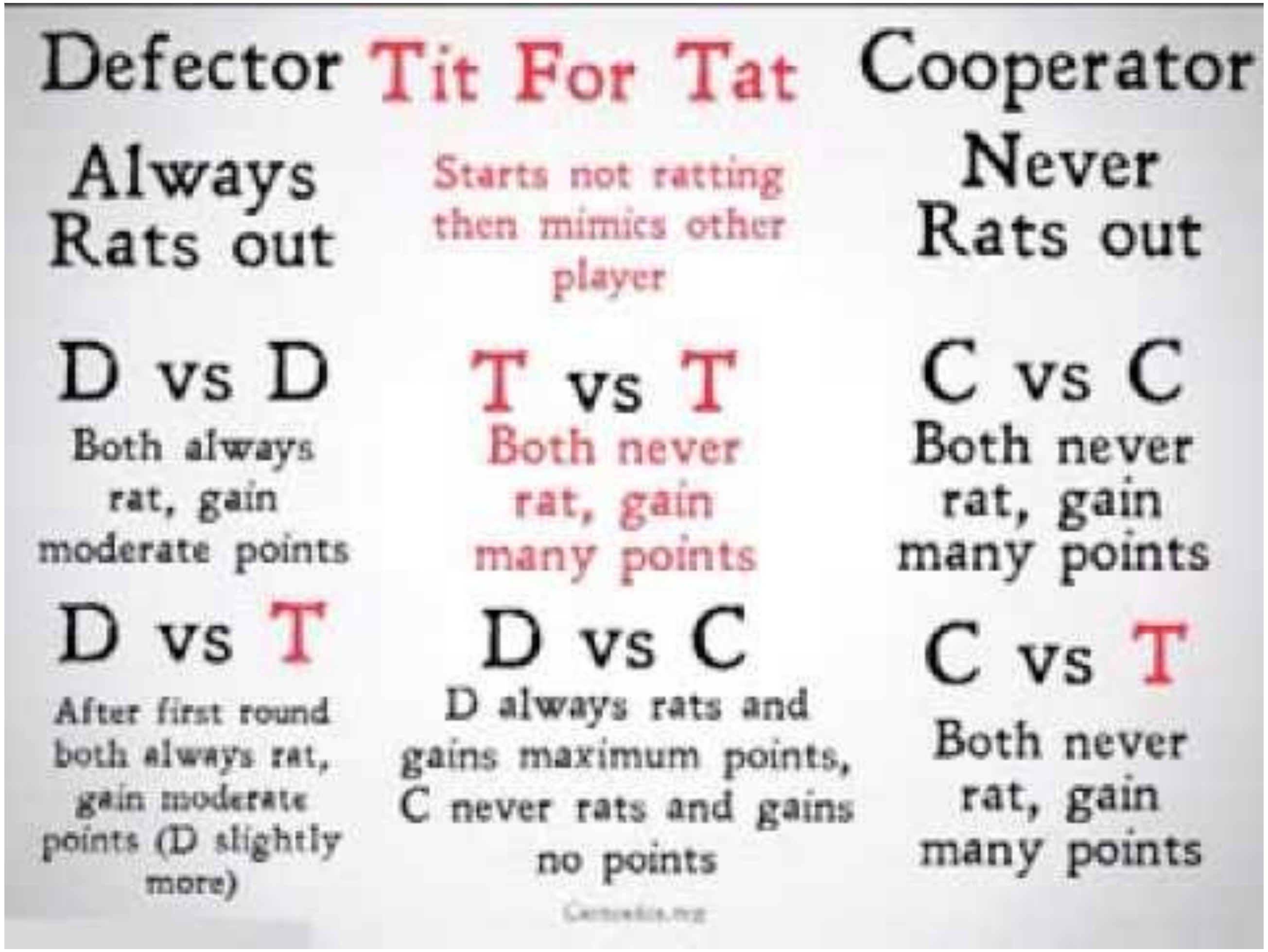
Prisoner's Dilemma

	Confess Defect	**Silent** Cooperate
Confess Defect	$(-5, -5)$ *Nash*	$(0, -10)$
Silent Cooperate	$(-10, 0)$	$(-2, -2)$ *Pareto*

Iterated Prisoner’s Dilemma

- You remember “opponent’s” behavior
- You will continue to meet your “opponents”
- Famous Experiment by Axelrod
- Rappoport introduced Tit for Tat
- You start being nice (Cooperating) and then do what the other did the previous round

Tit for Tat


Defector
Tit For Tat
Cooperator
Always Rats out
Starts not ratting then mimics other player
Never Rats out
D vs D
Both always rat, gain moderate points
T vs T
Both never rat, gain many points
C vs C
Both never rat, gain many points
D vs T
After first round both always rat, gain moderate points (D slightly more)
D vs C
D always rats and gains maximum points, C never rats and gains no points
C vs T
Both never rat, gain many points

Algorithms

Challenges

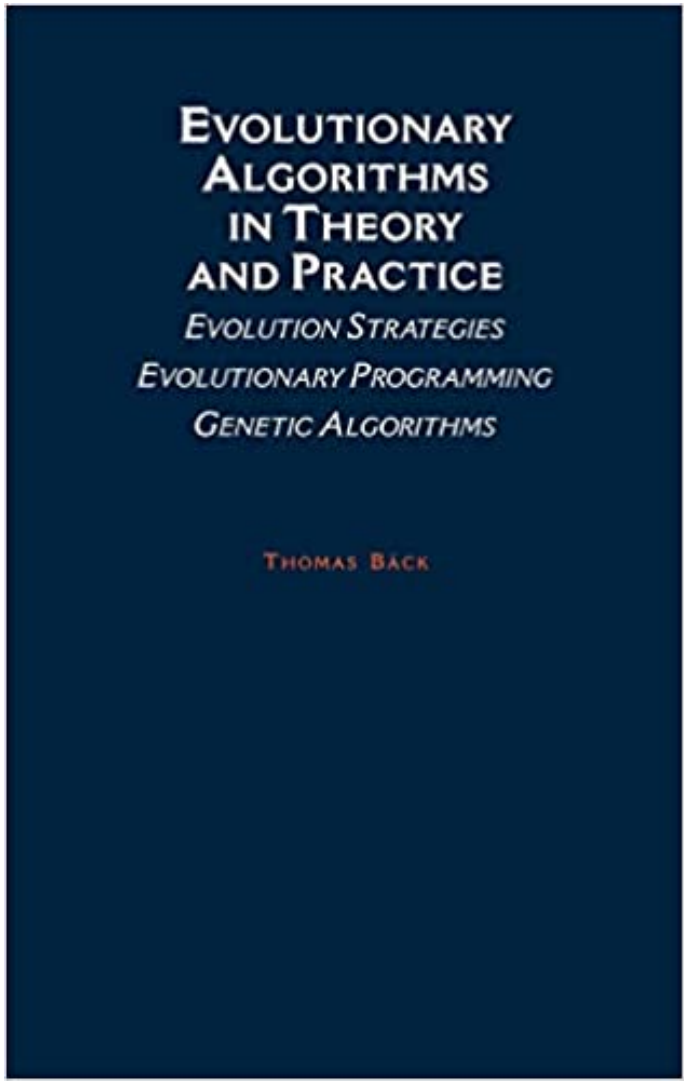
- Partial Observability -> Large State Space
  (Information sets)

- Nonstationary Environments -> Large State Space
  (Calculate all configurations)

- Multiple Agents -> Large State Space
  (Esp. with simultaneous actions)

Evolutionary Approaches

- Evolutionary Algorithms
- Swarm Computing
- Population based training (teams, HRL)



Evolutionary Framework

Algorithm 7.1 Evolutionary Framework [36]

```
Generate the initial population randomly
repeat
    Evaluate the fitness of each individual of the population
    Select the fittest individuals for reproduction
    Through crossover and mutation generate new individuals
    Replace the least fit individuals by the new individuals
until terminated
```

Evo

Multi-Agent Reinforcement Learning	Evolutionary Computation
agent	individual
some	many
all agents	population
environment	problem
reward	fitness
policy	genes
adaptation	mutation and combination
time step	generation
feedback	selection

- Highly parallel
- Multi-agent population based optimization
- Single-agent deep network policy optimization instead of backpropagation
- Single fitness function, determines cooperation or competition

Swarm Intelligence Algorithms

A Tutorial

Edited by
Adam Slowik

CRC Press
Taylor & Francis Group

Ant Colony Optimization

- ACO

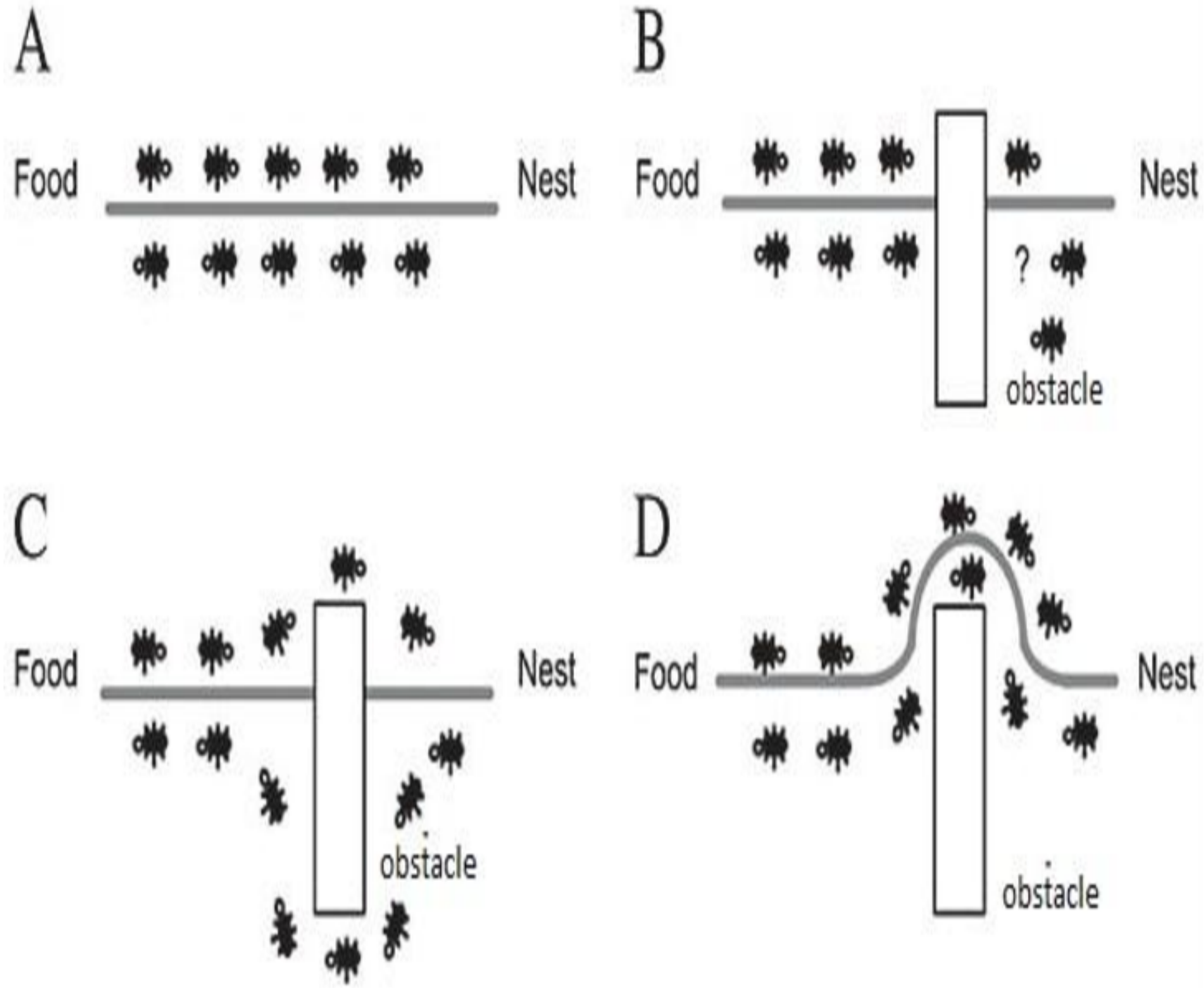


Fig 1.

A: Ants in a pheromone trail between nest and food.
B: an obstacle interrupts the trail.
C: Ants find two paths to go around the obstacle

Population-based training

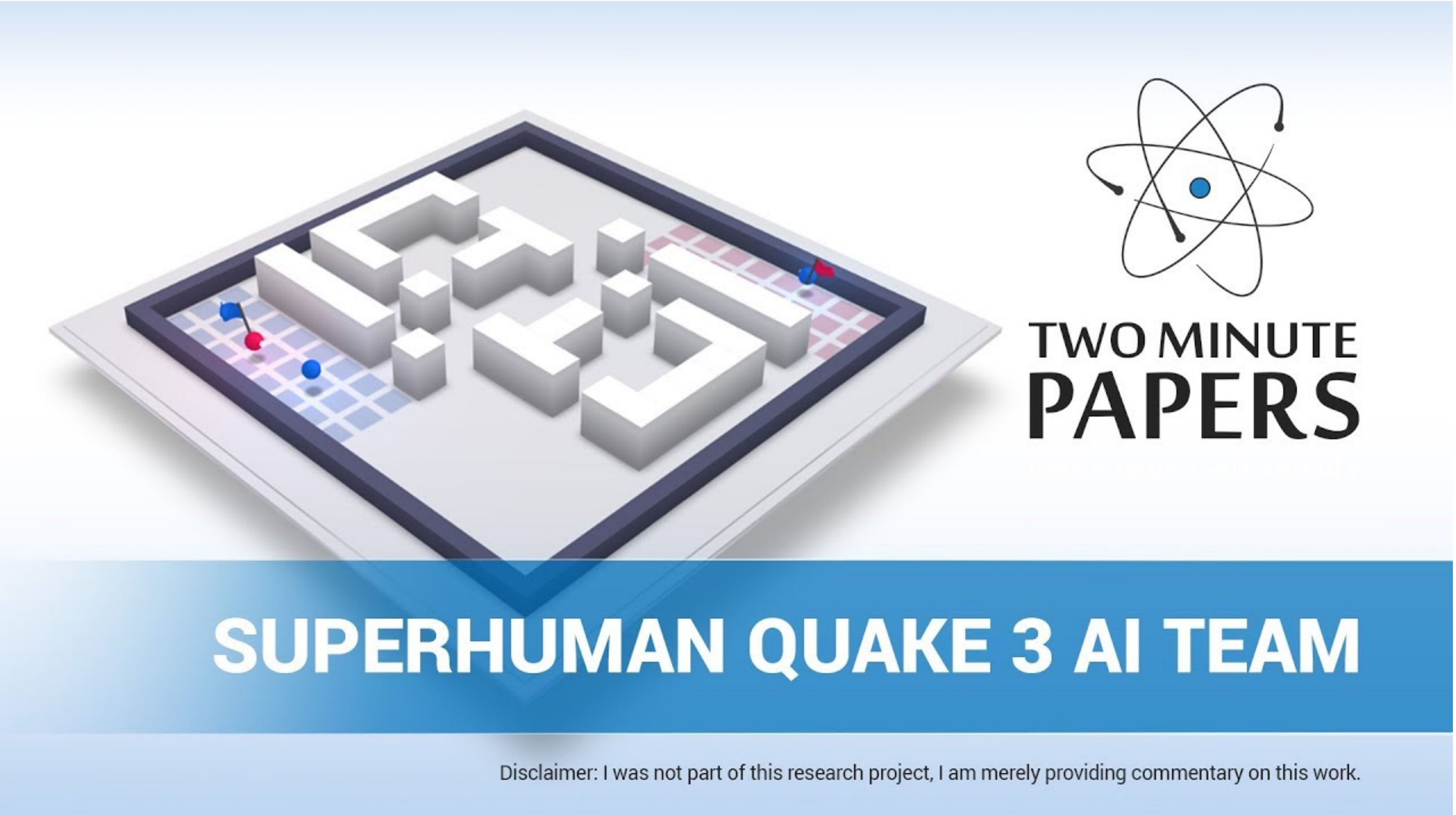
- Teams
- Hierarchical
- Cooperation, competition
- Within Teams, between teams
- Blends RL and Evo

Population-based training

Algorithm 7.2 Population Based Training [352]

```
procedure TRAIN(𝒫)                                                          ▷ initial population 𝒫
    for (θ, h, p, t) ∈ 𝒫 (asynchronously in parallel) do
        while not end of training do
            θ ← step(θ|h)                          ▷ one step of optimisation using hyperparameters h
            p ← eval(θ)                                                   ▷ current model evaluation
            if ready(p, t, 𝒫) then
                h′, θ′ ← exploit(h, θ, p, 𝒫)   ▷ use the rest of population for improvement
                if θ ≠ θ′ then
                    h, θ ← explore(h′, θ′, 𝒫)                  ▷ produce new hyperparameters h
                    p ← eval(θ)                                           ▷ new model evaluation
                end if
            end if
            update 𝒫 with new (θ, h, p, t + 1)                               ▷ update population
        end while
    end for
    return θ with the highest p in 𝒫
end procedure
```

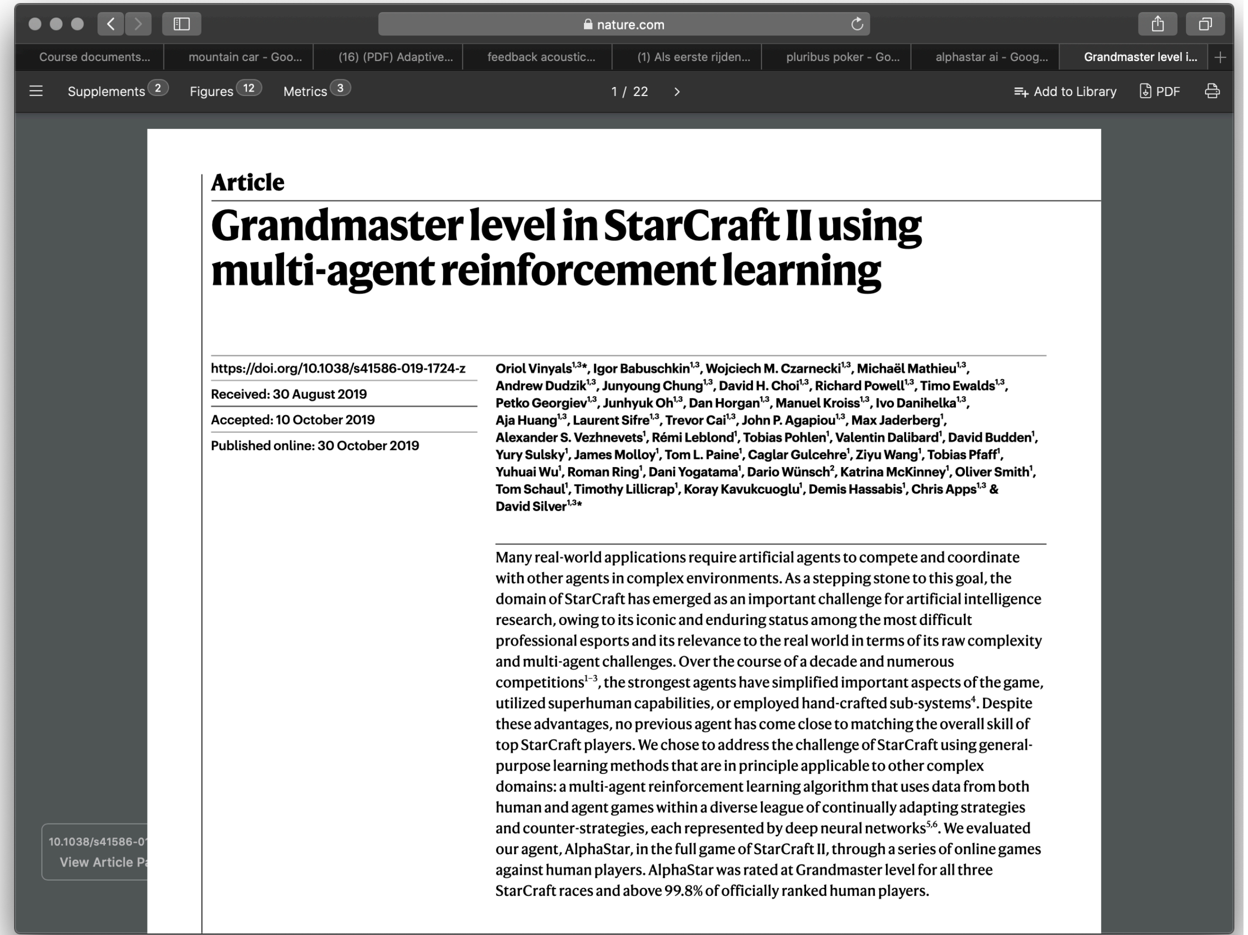
CTF


TWO MINUTE
PAPERS
SUPERHUMAN QUAKE 3 AI TEAM
Disclaimer: I was not part of this research project, I am merely providing commentary on this work.

StarCraft

StarCraft

- Real Time Strategy
- 10^1685
- AlphaStar
- Population based multi agent methods



Questions?