SIT796 Reinforcement Learning

Multi-Agent Reinforcement Learning and Related Topics

Presented by:
Thommen George Karimpanal
School of Information Technology

Markov Models and Agents

	No Agents	Single Agent	Multiple Agents
State Known	Markov Chain	Markov Decision Process (MDP)	Markov Game (a.k.a. Stochastic Game)
State Observed Indirectly	Hidden Markov Model (HMM)	Partially-Observable Markov Decision Process (POMDP)	Partially-Observable Stochastic Game (POSG)

Markov Models and Agents

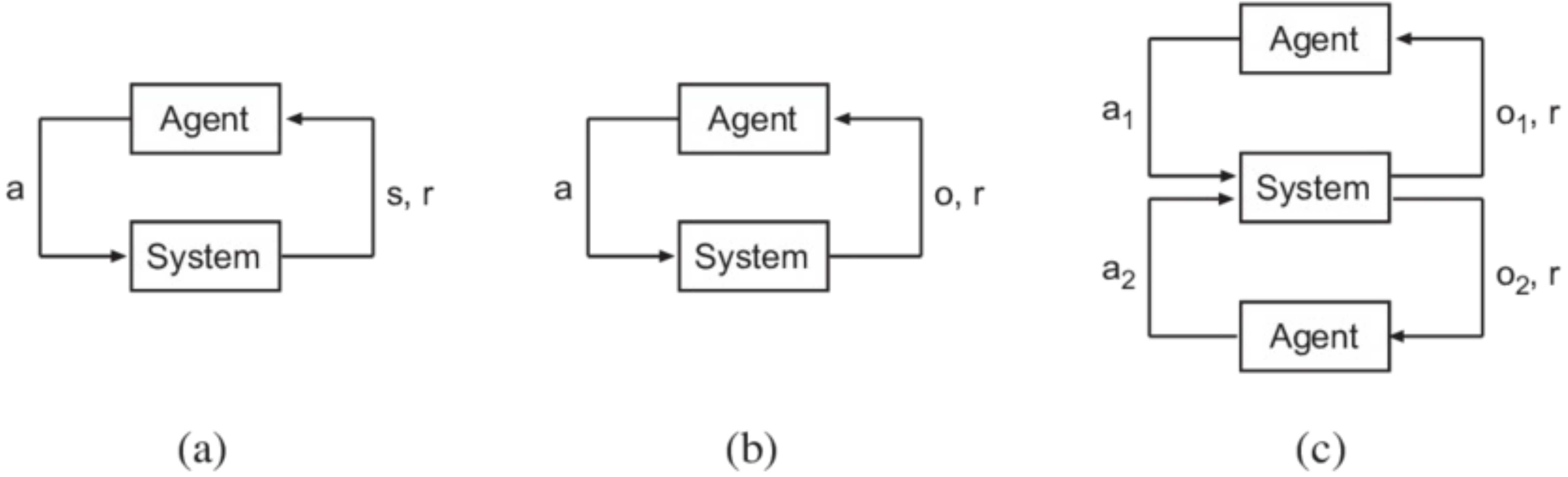


Figure: (a) Markov decision process (MDP) (b) Partially observable Markov decision process (POMDP) (c) Decentralized partially observable Markov decision process with two agents (Dec-POMDP)

Multi-agent Applications

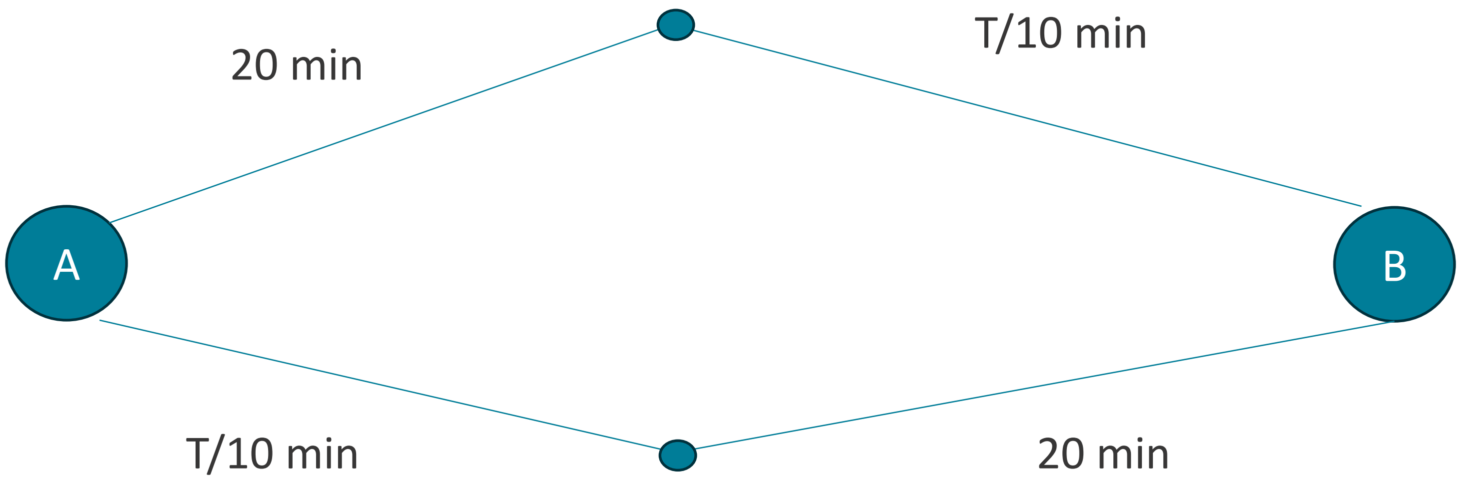
- Antenna tilt Control
  - The joint configuration of cellular base stations can be optimized according to the distribution of usage and topology of the local environment. (Each base station can be modelled as one of multiple agents covering a city.)

- Traffic congestion reduction
  - By intelligently controlling the speed of a few autonomous vehicles we can drastically increase the traffic flow
  - Other interesting phenomena (Braess Paradox)

Braess Paradox

- Traffic Control Strategies:

  -Build more roads where there is more traffic?

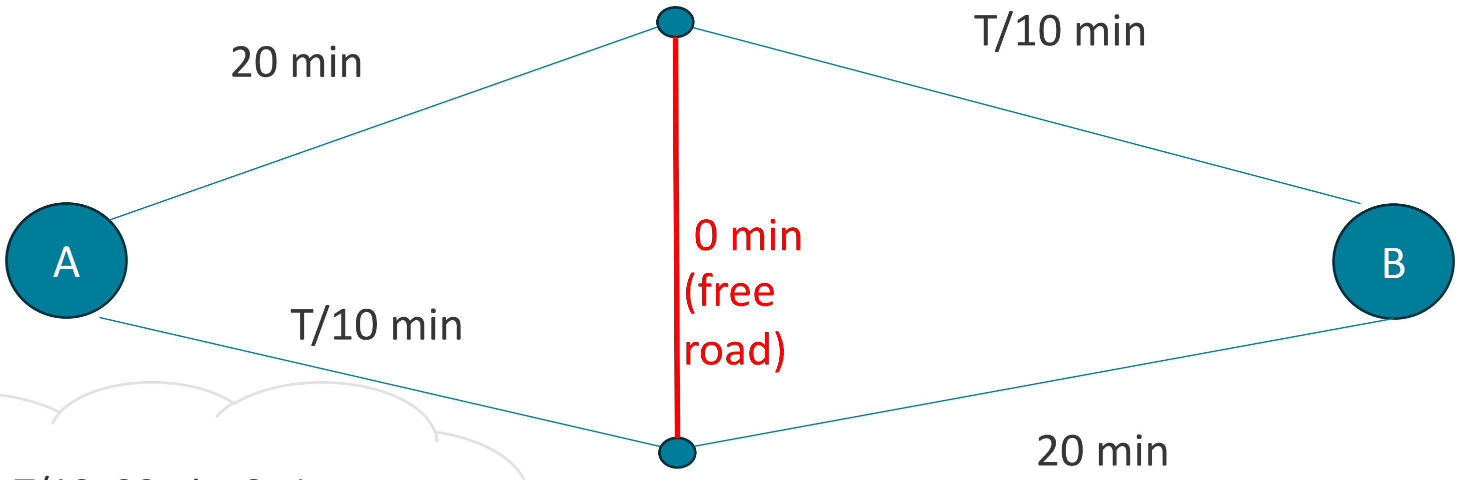


T=Traffic

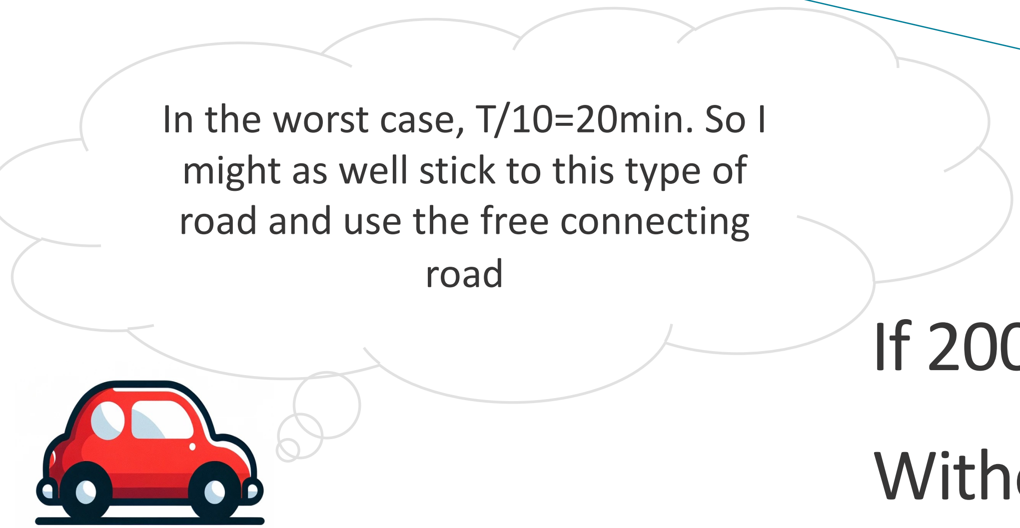
If 200 vehicles: Total time= 20+(100/10)=30min

(The 200 drivers split up as 100+100)

Braess Paradox



T=Traffic



If 200 vehicles: Total time= (200/10)+0+(200/10)=40min

Without free road: 30min

Sometimes, closing down roads can help traffic flow!

Multi-agent Applications

DEAKIN UNIVERSITY

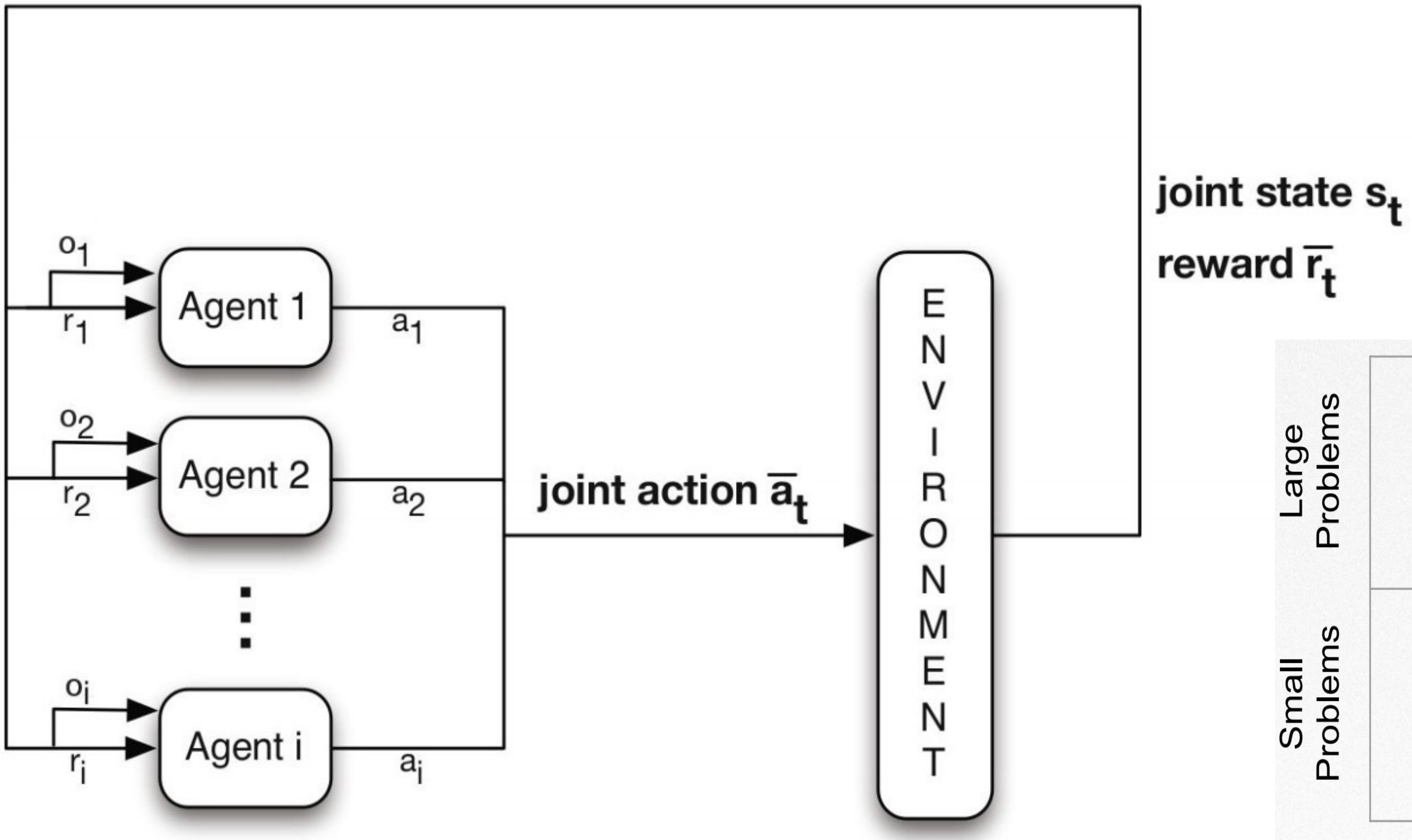
- OpenAI Five
  - Dota 2 AI agents are trained to coordinate with each other to compete against humans.
  - Each of the five AI players is implemented as a separate neural network policy and trained together with large-scale PPO.
  - They defeated a team of human pros.

SIT796 Reinforcement Learning

Multi-Agent Reinforcement Learning

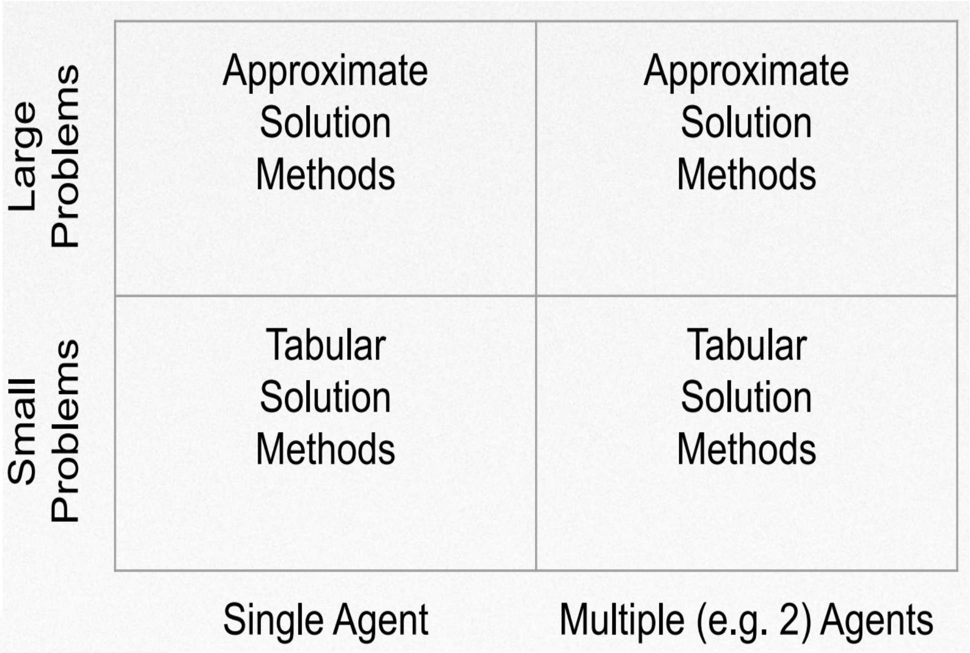
Presented by:
Thommen George Karimpanal
School of Information Technology

Multi-agent Reinforcement Learning (MARL)



Source: Nowe, Vrancx & De Hauwere 2012

- MARL
  - Multiple agents join to take joint actions



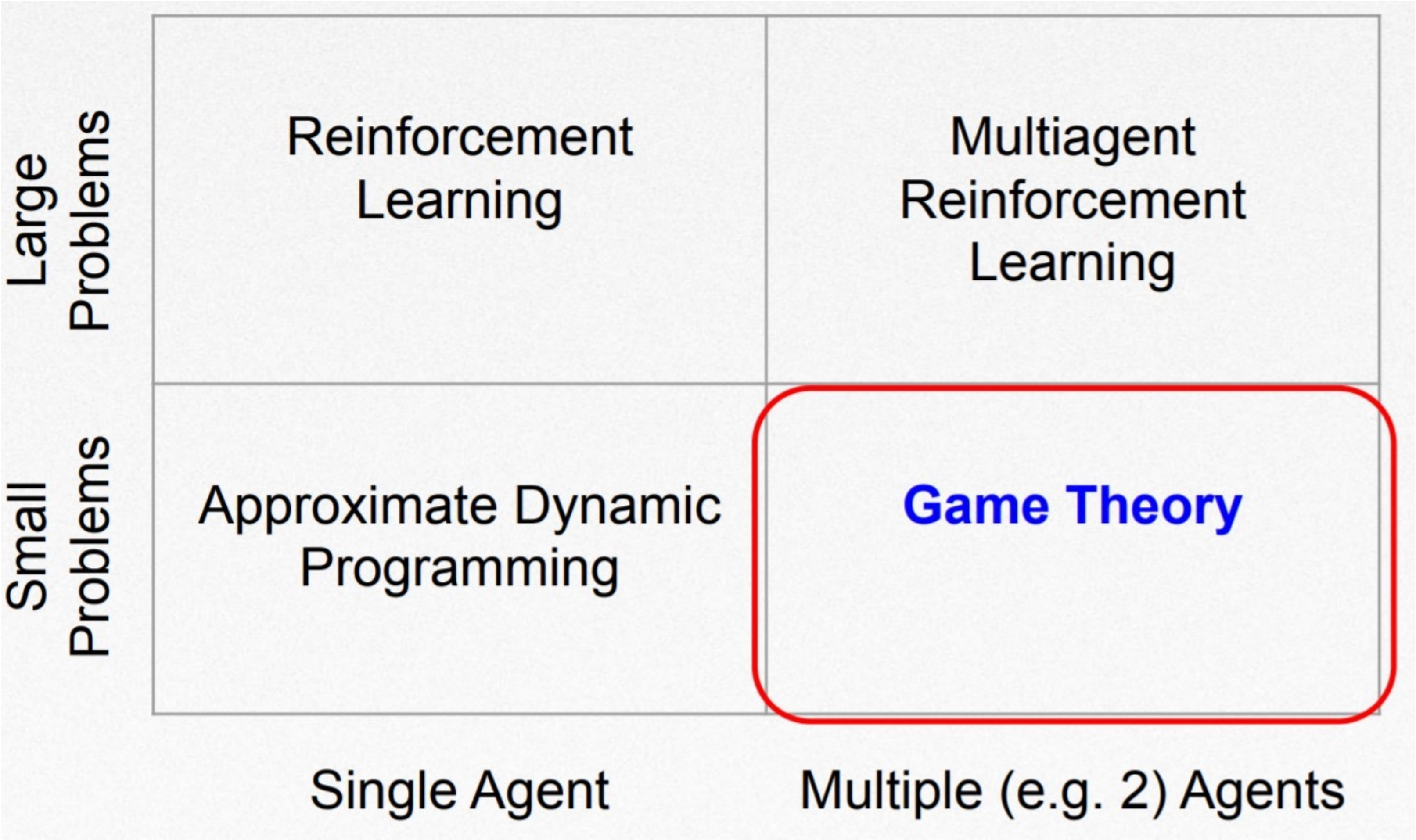
Types of MARL Settings

- **Decentralized:**
  - All agents learn individually
  - Communication limitations defined by environment
- **Descriptive:**
  - Forecast how agent will behave
- **Neither:**
  - Agents maximize their utility which may require cooperating and/or competing
  - General-sum game

VS

- **Centralized:**
  - One brain / algorithm deployed across many agents
- **Prescriptive:**
  - Suggests how agents should behave
- **Competitive:**
  - Agents compete against each other
  - Zero-sum games
  - Individual opposing rewards
- **Cooperative:**
  - Agents cooperate to achieve a goal
  - Shared team reward

Foundations of MARL



Benefits:

- **Sharing experience** via communication, teaching, imitation
- **Parallel computation** due to decentralized task structure
- **Robustness** redundancy, having multiple agents to accomplish a task

Challenges in Multi-agent Learning Systems

- **Curse of dimensionality**
  - Exponential growth in computational complexity from increase in state and action dimensions.
  - Also a challenge for single-agent problems.
- **Specifying a good (learning) objective**
  - Agent returns are correlated and cannot be maximized independently.
- **The system in which to learn is a moving target**
  - As some agents learn, the system which contains these agents changes, and so may the best policy.
  - Also called a system with non-stationary or time-dependent dynamics.
- **Need for coordination**
  - Agent actions affect other agents and could confuse other agents (or herself) if not careful. Also called destabilizing training.

Challenges: Non-stationarity of Environment

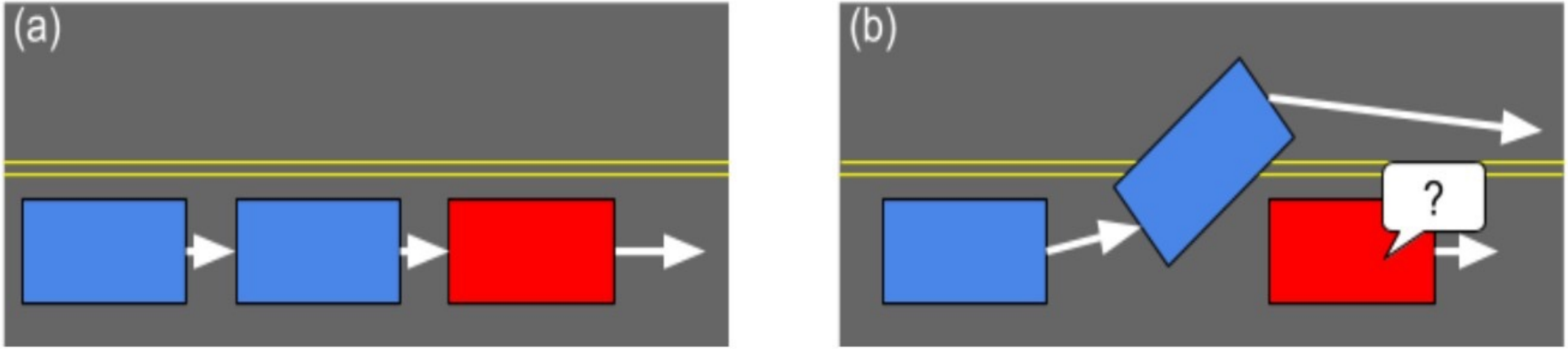


Figure 2: *Non-stationarity of environment: Initially (a), the red agent learns to regulate the speed of the traffic by slowing down. However, over time the blue agents learn to bypass the red agent (b), rendering the previous experiences of the red agent invalid.*

Challenges: High Variance of Estimates

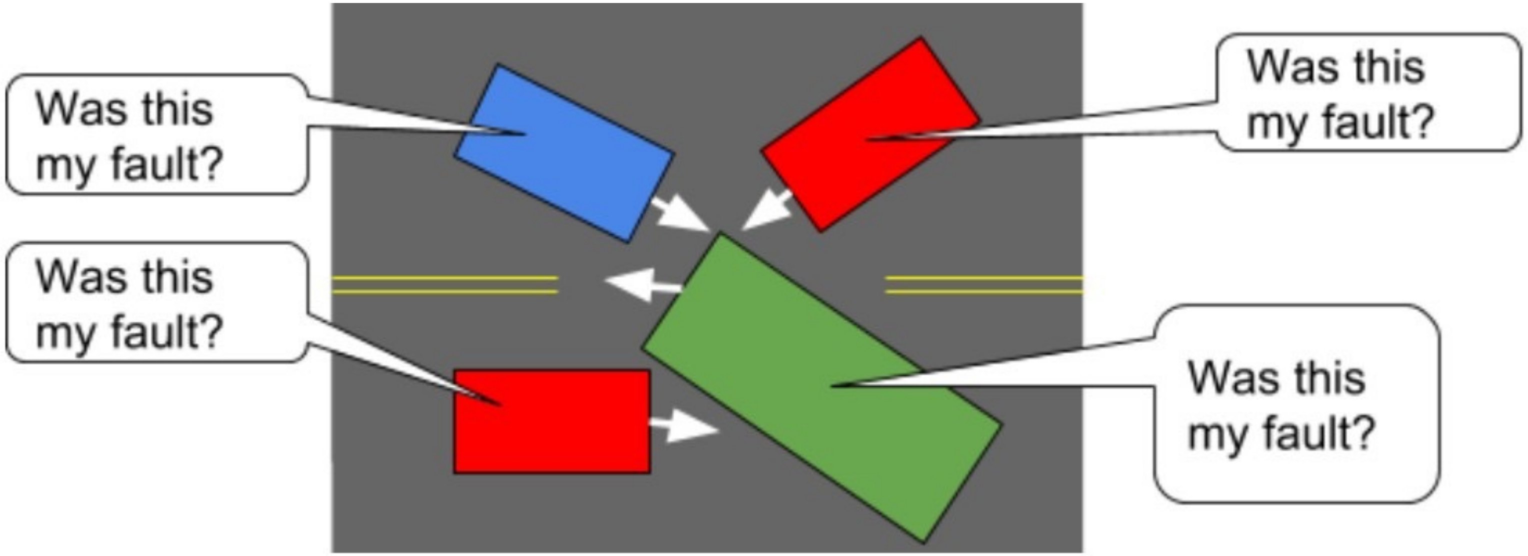


***Figure 4: High variance of advantage estimates**: In this traffic gridlock situation, it is unclear which agents' actions contributed most to the problem -- and when the gridlock is resolved, from any global reward it will be unclear which agents get credit.*

In Summary…

- In single agent RL, agents need only to adapt their behaviour in accordance with their own actions and how they change the environment.

- In MARL agents also need to adapt to other agents' learning and actions. The effect is that agents can execute the same action on the same state and receive different rewards.

SIT796 Reinforcement Learning

Game Theory

Presented by:
Thommen George Karimpanal
School of Information Technology

DEAKIN UNIVERSITY

Game Theory: Concepts

What is Game Theory?

-The mathematics of conflict

-Proposed by John Nash in his 27 page PhD thesis

-Assumes players are rational

-Increasing number of applications in AI

-Applications: economics, politics, robotics, etc.,

John Nash

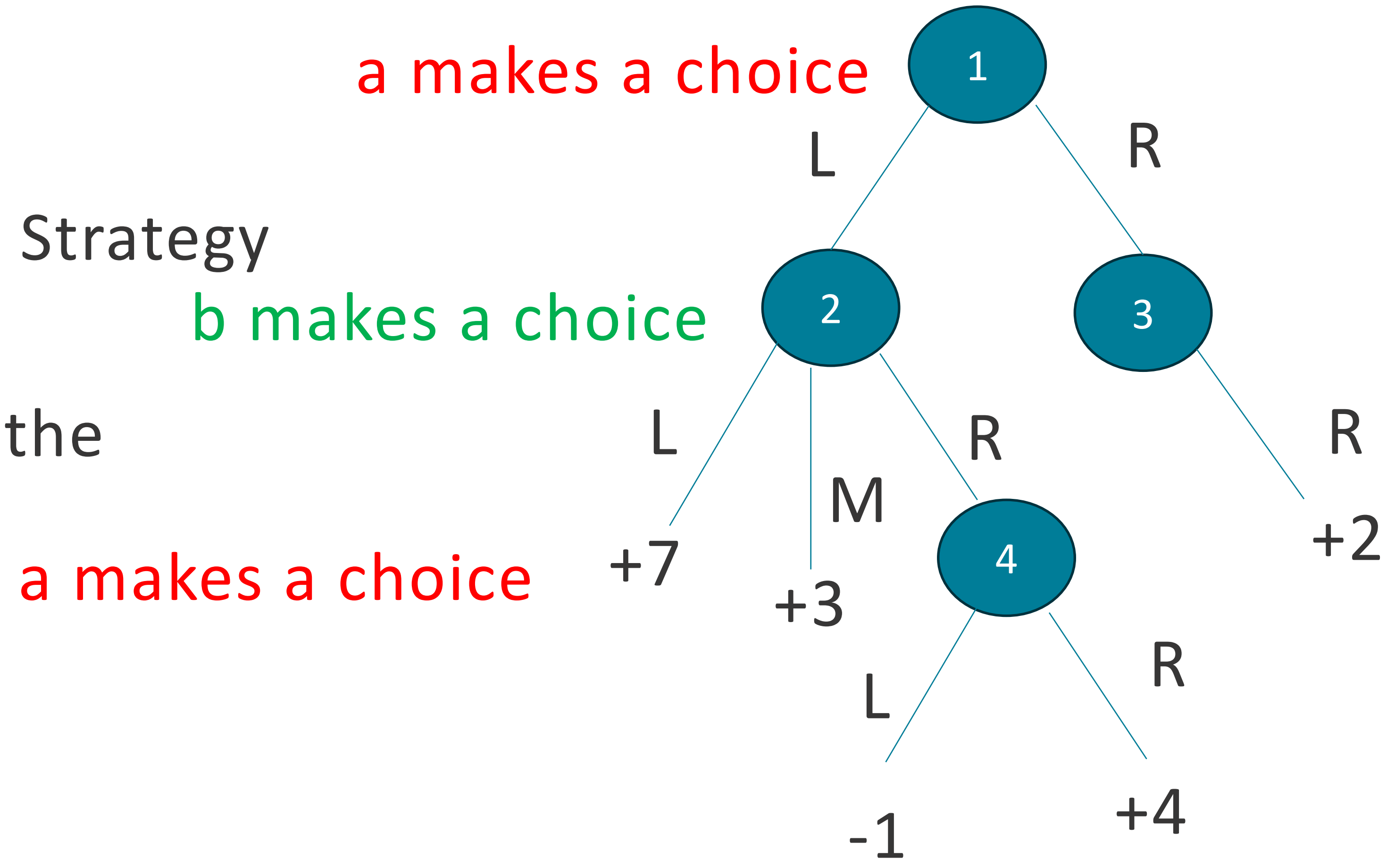
Source:https://www.scientificamerican.com/article/beautiful-mind-john-nash-s-schizophrenia-disappeared-as-he-aged/

A simple game

-a, b make choices (L/R)

- MDP: policy   Game Theory: Strategy

- Rewards of agents add up to the same number



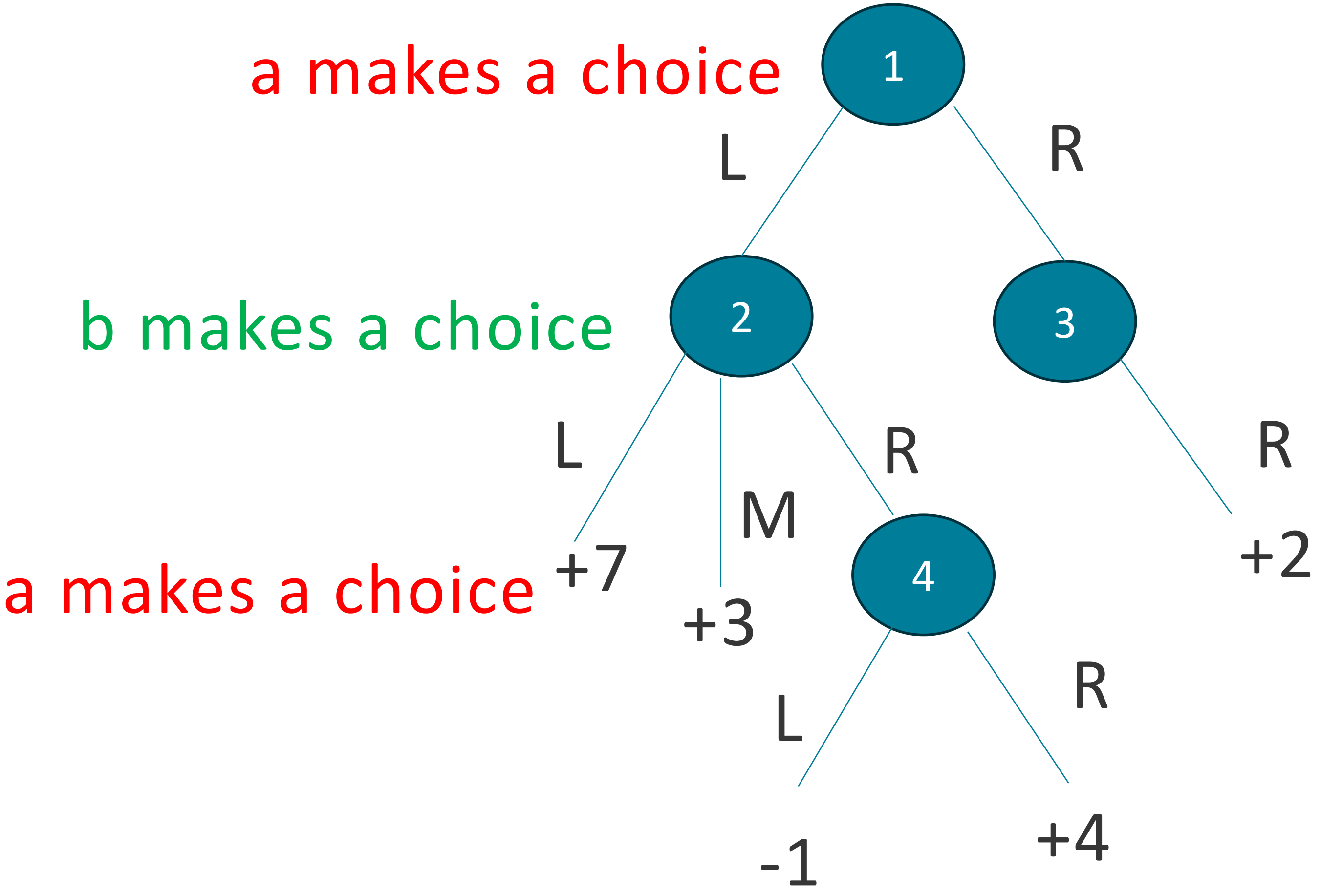
2 player zero sum finite deterministic game with perfect information

A simple game

b:

1	4			
		2	L	M	R
		3	R	R	R
a: L	L		7	3	-1
L	R		7	3	4
R	L		2	2	2
R	R		2	2	2

Matrix form of the game



2 player zero sum finite deterministic game with perfect information

A simple game: minimax

			b:		
		2	L	M	R
1	4	3	R	R	R
a: L	L		**7**	**3**	**-1**
L	R		**7**	**3**	**4**
R	L		**2**	**2**	**2**
R	R		**2**	**2**	**2**

Matrix form of the game

a tries to pick the best row for it

b tries to pick the best column for it

Or the other way around – one tries to 'max', the other tries to 'min'

Value of the game

Nash Equilibrium

Given n players with strategies: $S = \{S_0, \dots S_i, \dots S_n\}$

$S_0^* \in S_0, S_1^* \in S_1, S_2^* \in S_2 \dots\dots S_n^* \in S_n$ Are in Nash Equilibrium iff:

$$\forall_i S_i^* = argmax_{S^*} \, U_i(S_0^*, \dots S_n^*)$$

Basically, in a Nash Equilibrium, if you pick a player at random, they would prefer to not deviate from their optimal strategy, given the optimal strategies of other players

SIT796 Reinforcement Learning

Multi-Agent Reinforcement Learning Formulation

Presented by:
Thommen George Karimpanal
School of Information Technology

Stochastic Games

S:State space

A_i: Action space for each agent $a \in A_1, b \in A_2$

R_i: Rewards for each player i $R_1(s,(a,b)), R_2(s,(a,b))$

T: Transitions function $T(s,(a,b),s')$

γ: Discount factor

Generalisation of the MDP formulation (Shapley) – published before Bellman

Zero sum Stochastic Games: Bellman Equation

Single agent:

$$Q(s, a): \mathrm{R}(s, a) + \gamma \sum_{s'} T(s, a, s')\, max_{a'} Q(s', a')$$

Two agents (zero sum):

$$Q_i(s, (a, b)): R_i(s, (a, b)) + \gamma \sum_{s'} T(s, (a, b), s')\, \underbrace{max_{a' b'} Q(s', (a', b'))}$$

But we are no longer the only agent trying to maximise reward! Use minimax!

First MARL Algorithm: Minimax-Q (Littman '94)

Q-values are over joint actions: $Q(s, a, o)$

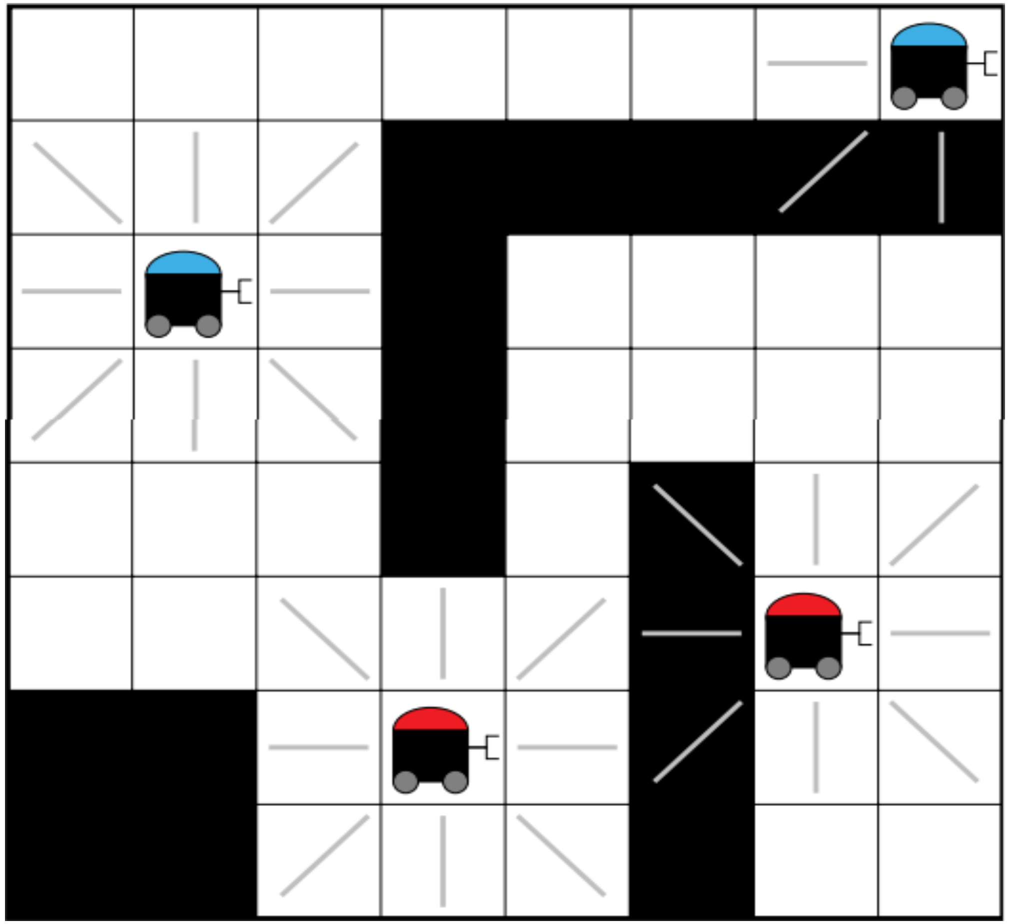
- s = state
- a = your action
- o = action of the opponent

Instead of updating Q values with max$Q(s', a')$, use **MaxMin**

$$Q(\mathrm{s}, (\mathrm{a}, \mathrm{b})) = Q(\mathrm{s}, (\mathrm{a}, \mathrm{b})) + \alpha[R_i(\mathrm{s}, (\mathrm{a}, \mathrm{b})) + \underbrace{\gamma minimax_{a'b'} Q(s', (a', b'))}_{\text{Only change from Q learning}} - Q(\mathrm{s}, (\mathrm{a}, \mathrm{b})]$$

Multi-agent Deep Q-Network (MADQN)

- MADQN is a Deep Q-Network for Multi-agent RL
  - *n* pursuit-evasion – a set of agents (the pursuers) are attempting to chase another set of agents (the evaders)
  - The agents in the problem are
  - self-interested (or heterogeneous), i.e. they have different objectives
  - The two pursuers are attempting to catch the two evaders



Other Deep RL approaches

- MADDPG (multi agent deep deterministic policy gradients): multiagent extension of DDPG
- Multi-Agent Common Knowledge Reinforcement Learning: more focused on cooperative tasks
- Qmix: For training decentralised policies

SIT796 Reinforcement Learning

Other Related Topics: Action Advising

Presented by:
Thommen George Karimpanal
School of Information Technology

Teacher-Student Framework

Does not explicitly fall under multiagent learning, but involves one agent teaching the other

Teacher already knows a good policy

Student learns from scratch, but can ask for advice

Advice is limited, can have an associated cost

Teachers cannot access student knowledge

How can the student quickly best leverage the provided advice while staying within the advice budget?

Action Advising

DEAKIN UNIVERSITY

n: Advice Budget

procedure EarlyAdvising(π, n)
  for each student state s **do**
    if $n > 0$ **then**
      $n \leftarrow n - 1$
      Advise $\pi(s)$

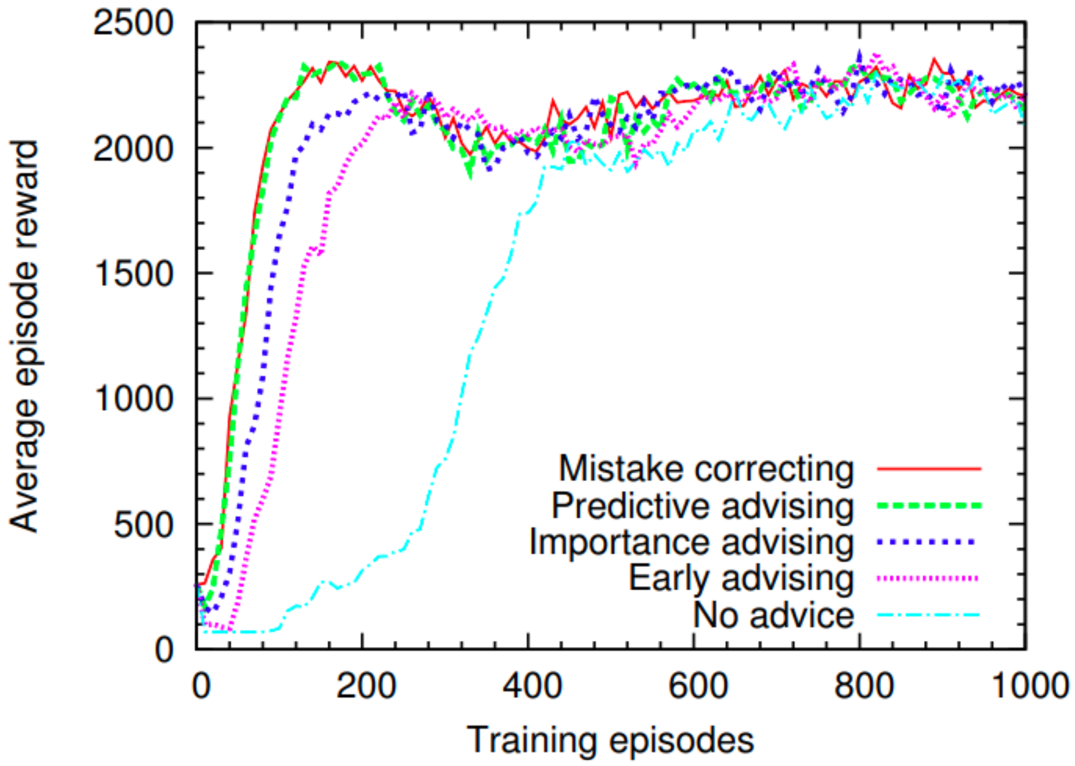
procedure MistakeCorrecting(π, n, t)
  for each student state s **do**
    Observe student's announced action a
    if $n > 0$ and $I(s) \geq t$ and $a \neq \pi(s)$ **then**
      $n \leftarrow n - 1$
      Advise $\pi(s)$

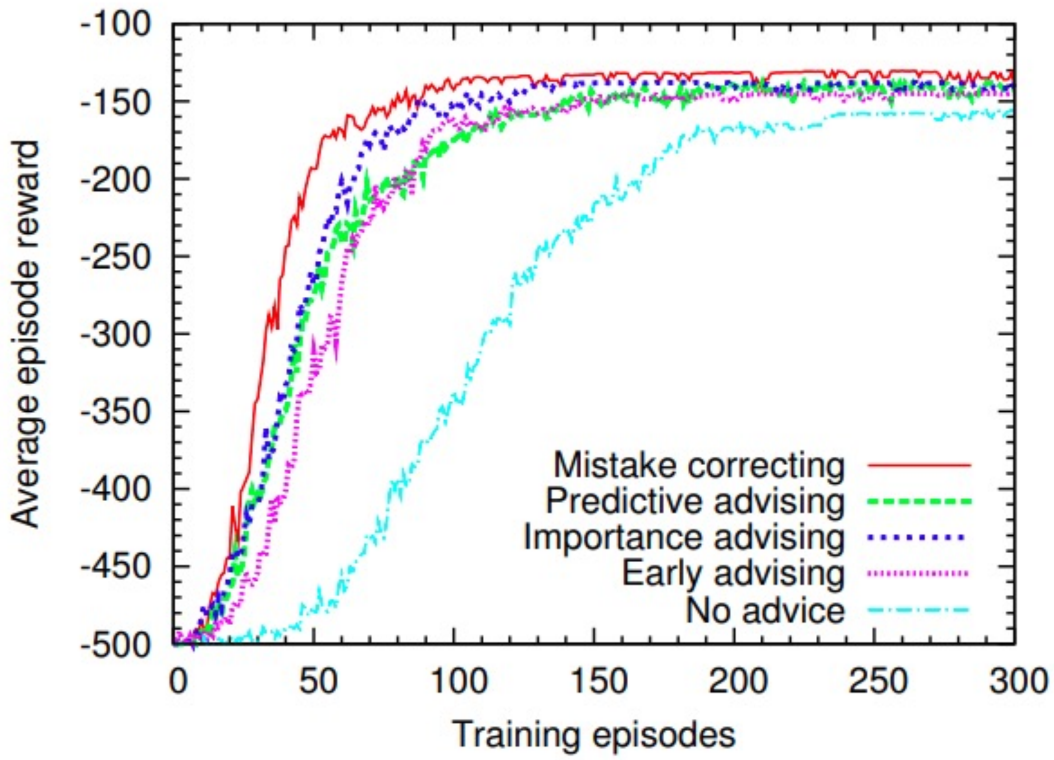
$$I(s) = \max_a Q(s,a) - \min_a Q(s,a)$$

procedure ImportanceAdvising(π, n, t)
  for each student state s **do**
    if $n > 0$ and $I(s) \geq t$ **then**
      $n \leftarrow n - 1$
      Advise $\pi(s)$

procedure PredictiveAdvising(π, n, t)
  for each student state s **do**
    Predict student's intended action a
    if $n > 0$ and $I(s) \geq t$ and $a \neq \pi(s)$ **then**
      $n \leftarrow n - 1$
      Advise $\pi(s)$

Action Advising





*Teaching on a Budget: Agents Advising Agents in Reinforcement Learning, Torrey& Taylor (AAMAS, 2013)

Readings

This lecture focused on introducing Multi-agent RL.

For more detailed information see:

- https://www.udacity.com/course/deep-reinforcement-learning-nanodegree--nd893
- Littman, Michael L. "Markov games as a framework for multi-agent reinforcement learning." *Machine learning proceedings 1994*. Morgan Kaufmann, 1994. 157-163.
- Teaching on a Budget: Agents Advising Agents in Reinforcement Learning, Torrey& Taylor (AAMAS, 2013)
- Multiagent Reinforcement Learning presentation by Marc Lanctot RLSS @Lille, July 11th 2019
  http://mlanctot.info/files/papers/Lanctot_MARL_RLSS2019_Lille.pdf
- Multiagent Learning Foundations and Recent Trends by Stefano Albrecht and Peter Stone Tutorial at IJCAI 2017 conference
  https://www.cs.utexas.edu/~larg/ijcai17_tutorial/