UNIVERSITY OF LIVERPOOL

COMP 532
Machine Learning and BioInspired Optimization

Lecture 21: Multi-Agent Learning

Dr. Shan Luo

Department of Computer Science

shan.luo@liverpool.ac.uk

Outline (3-ish lectures)

- Introduction to Evolutionary Game Theory
  - Replicator Dynamics
  - Evolutionarily Stable Strategies
  - Example games
- Formal link between RL and EGT
  - Deriving the dynamics of Cross Learning
  - Extension to other RL algorithms
- Applications of this model
  - Parameter tuning
  - Analyzing complex strategic interactions

Recap: The Link between EGT and RL

- The **replicator dynamics** of EGT predict the expected **learning behaviour** of RL algorithms
- Different dynamics model different algorithms
- This allows us to compare those algorithms qualitatively
  - in terms of transient behaviour
  - and in terms of convergence

- Main reading material:

  Bloembergen, *et al.*., 2015. Evolutionary Dynamics of Multi-Agent Learning: A Survey. *JAIR*, *53*, pp.659-697.

  Available in Vital.

Applications of the Evolutionary Model

- The evolutionary model is useful to study dynamics of a given learner ..
  .. which can facilitate **parameter tuning**

- **Reverse approach**: design a learning algorithm that exhibits desired dynamics

- Study **complex strategic interactions** that normally defy formal analysis

Tuning the Exploration Rate

- Great challenge in RL: trade-off between **exploration** and **exploitation**

- Replicator Dynamics:
  - **Selection** favours good strategies over others:
    → greedy concept of exploitation
  - **Mutation** provides variety:
    → exploration

- The dynamics provide a visual clue to the effect of different exploration rates!

Tuning the Exploration Rate

Example: Battle of the Sexes

	B	S
B	2, 1	0, 0
S	0, 0	1, 2

- Boltzmann Q-learning
  - temperature τ
  - $\tau \to 0$: **exploitation**
  - $\tau \to \infty$: **exploration**

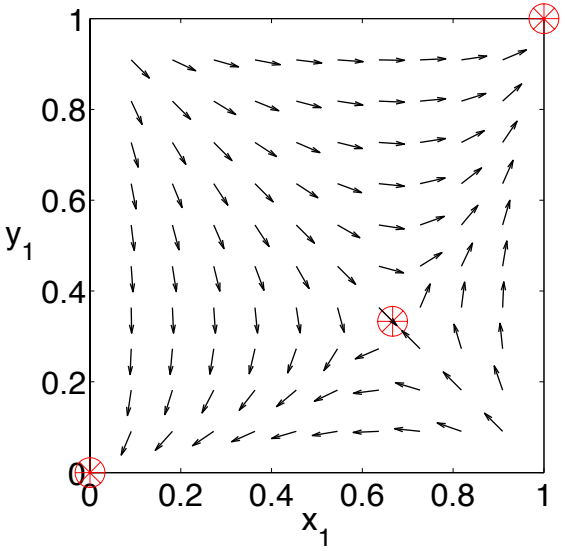
$$\pi(a) = \frac{e^{Q(a)/\tau}}{\sum_b e^{Q(b)/\tau}}$$

Tuning the Exploration Rate

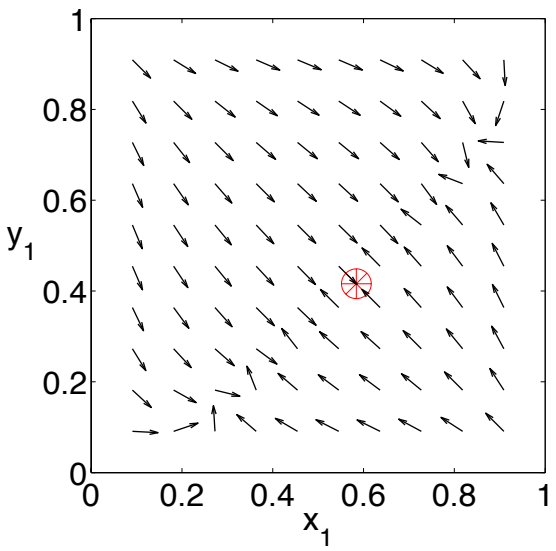
$\tau \to 0$: **exploitation**

$\tau \to \infty$: **exploration**

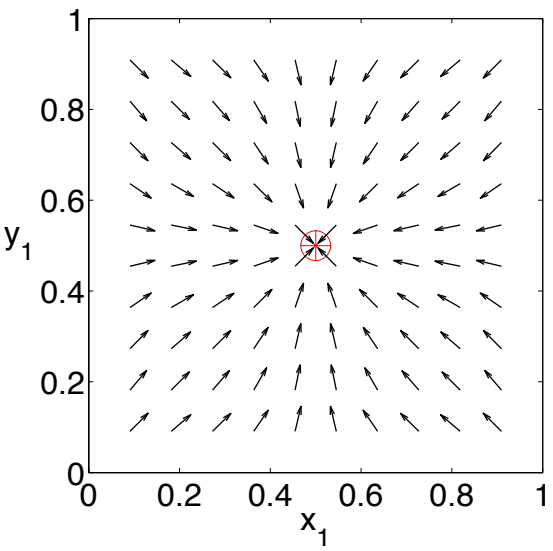
	B	S
B	2, 1	0, 0
S	0, 0	1, 2



$\tau \to 0$

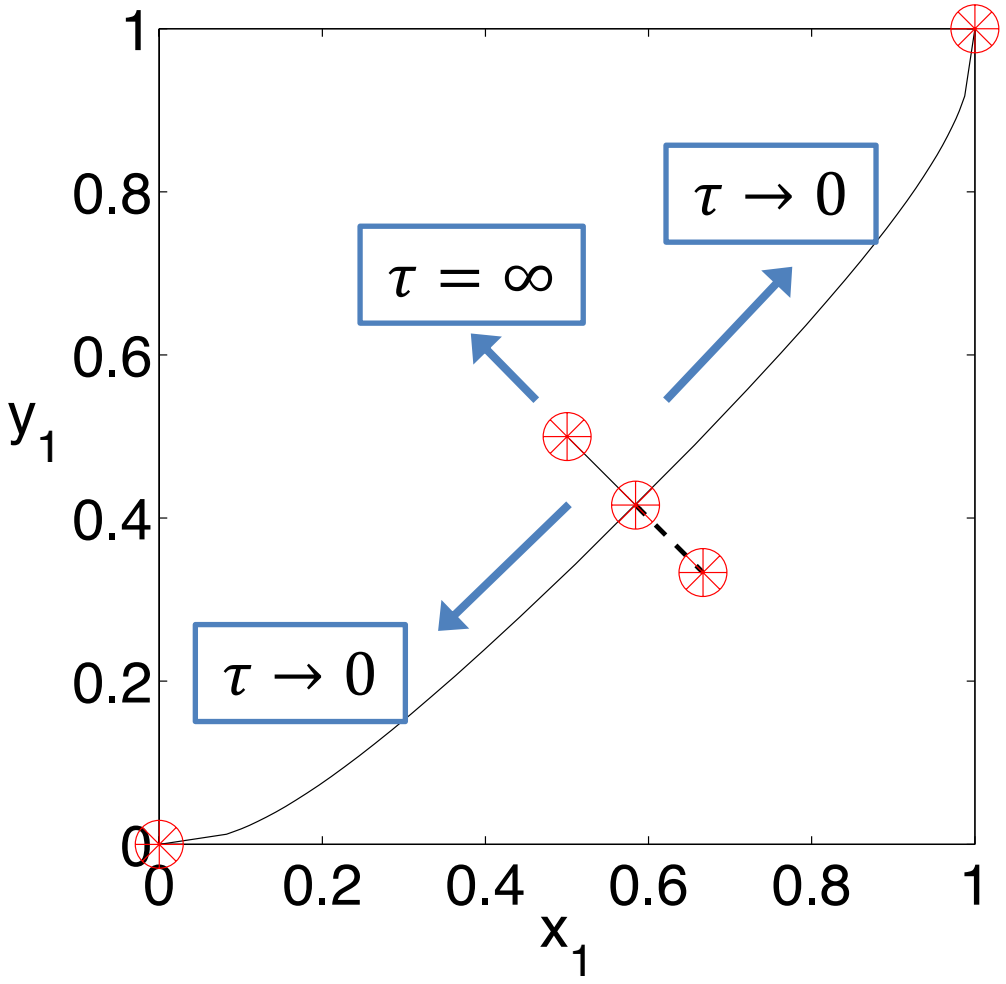


$\tau \approx 0.73$



$\tau \to \infty$

Tuning the Exploration Rate



Lenient Learning

- In cooperative / coordination settings:
  - Initial exploration in multi-agent learning causes (expected) mis-coordination
  - How to attribute low rewards? **Whose fault is it?**

- **Can lead to suboptimal convergence!**

- **Lenient Learning**: focus on maximal rewards only
  - **Leniency**: the quality of being more merciful or tolerant than expected

Lenient Learning

How to implement lenient learning?

- Collect k rewards for an action before doing an update based on the max of those rewards
  - **Optimistic** action values!
  - **Ignores low rewards** due to mis-coordination
- Can be incorporated into any RL algorithm

The Effect of Leniency

Example: **Stag Hunt**

- Two hunters go out to hunt. They can hunt for Stag or Hare.
  - **Stag**: requires working together
  - **Hare**: can be caught individually

	S	H
S	4, 4	1, 3
H	3, 1	3, 3

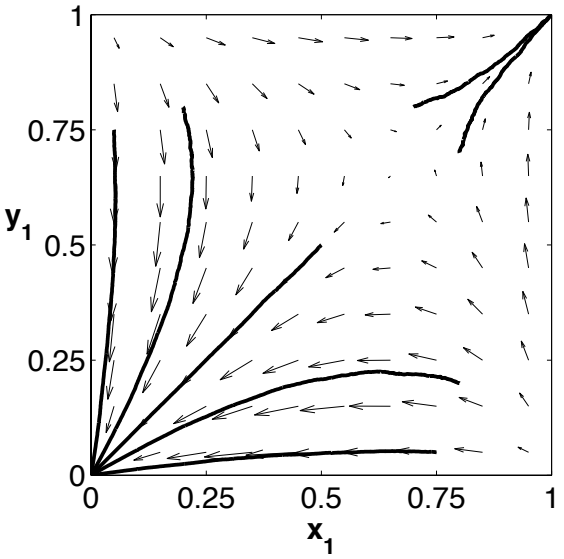
- Pure NE: (S, S) and (H, H)
  - (S,S) is optimal..
  - ..but (H,H) is safe

The Effect of Leniency

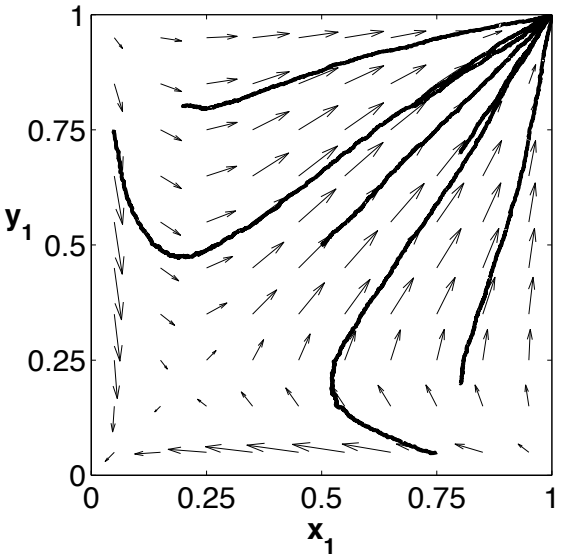
Example: **Stag Hunt**

- Pure NE: (S, S) and (H, H)
  - (S,S) is optimal..
  - ..but (H,H) is safe

	S	H
S	4, 4	1, 3
H	3, 1	3, 3



Q-learning



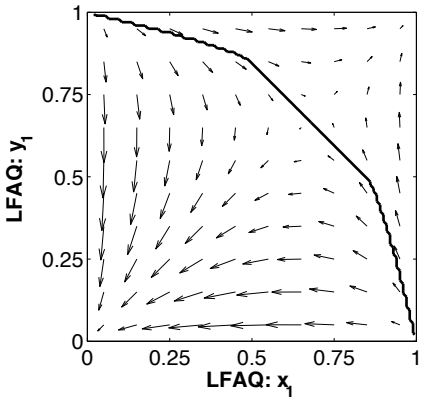
Lenient Q-learning

The Effect of Leniency

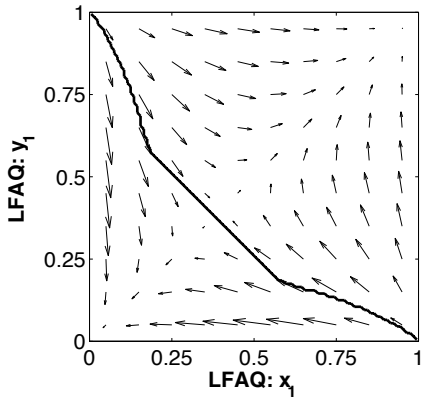
Basin of attraction:

- Area of the initial policy space that eventually converges to a specific NE
- Larger basin = more attractive NE

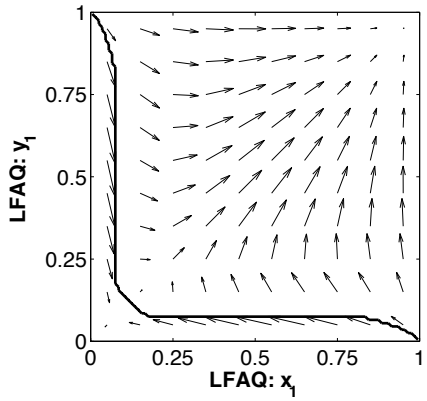
	S	H
S	4, 4	1, 3
H	3, 1	3, 3



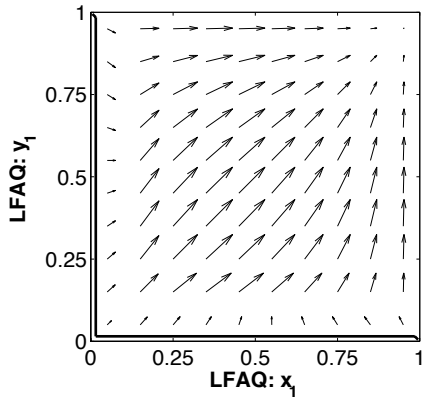
$k = 1$



$k = 2$



$k = 5$



$k = 25$

Complex Strategic Interactions

- Many real-world systems are too complex to model as a game
  - too many actions available
  - state space too big
  - impossible to write down the payoff function
- Sometimes we can take a high level view, by focusing on heuristic **meta-strategies**
- Naturally defined **types of behaviour** that can be made up of many atomic actions
  - E.g. styles of play in Poker (shark, fish)
  - trading strategies in stock markets
  - or collision avoidance methods in multi-robot systems

Complex Strategic Interactions

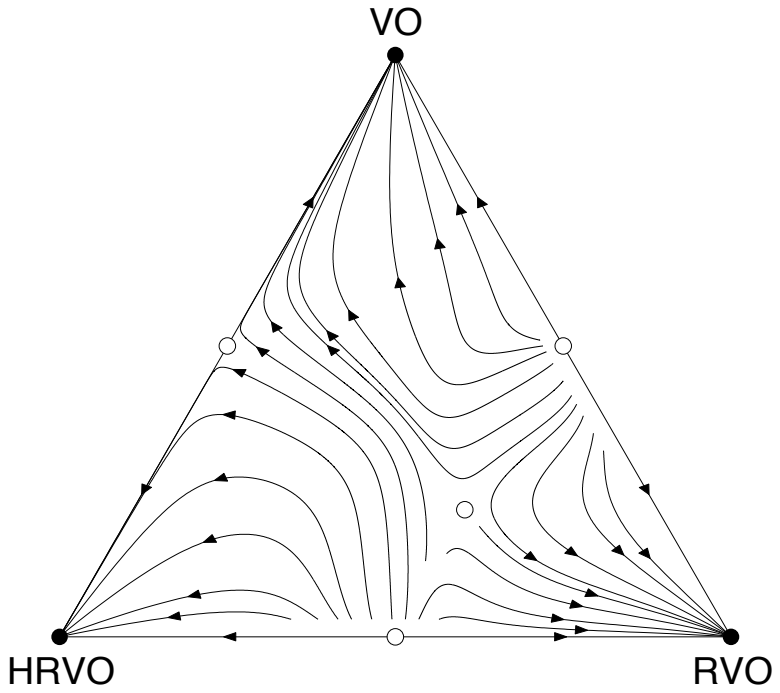
- Meta-strategies are useful to reduce the action space, but how to define the payoff function?

- **Empirical game theory** [Wellman2006]:
  - Estimate payoff function using simulation or real data
  - Use this estimate as input to game theoretic methods

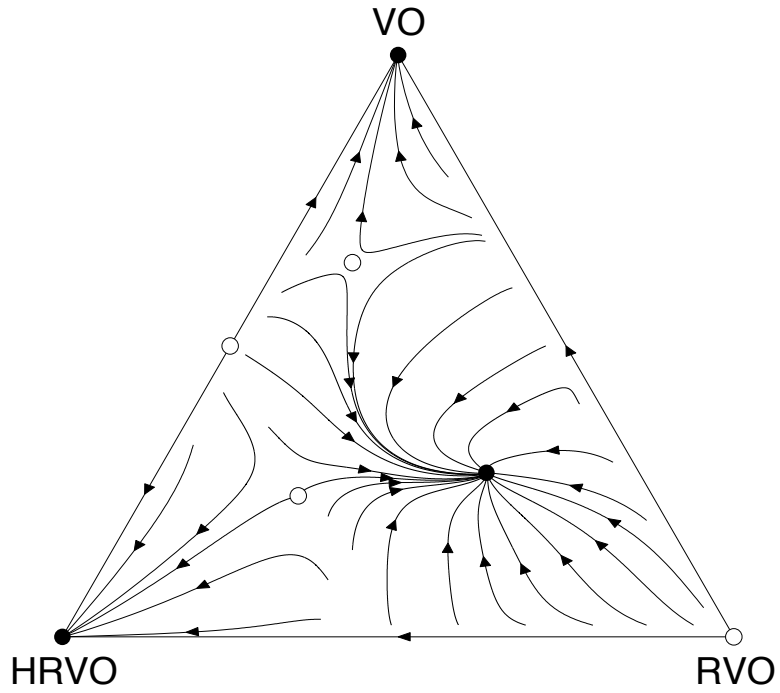
- Meta-strategies often indirectly collapse the state space as well, yielding a normal-form game
  - We then estimate a **heuristic payoff table**

Multi-Robot Collision Avoidance

Evolutionary dynamics of three strategies for multi-robot collision avoidance



Without “truncation”



With “truncation”

Other Complex Strategic Interactions

- The evolutionary model has been used to analyze:
  - different market mechanisms
  - trading strategies in stock markets
  - playing strategies in Texas Hold'em Poker
  - the space debris removal dilemma

- The **link with MARL** allows to predict what will happen when agents **learn** in these settings

Wrapping up

- Applications of the EGT models
  - Parameter tuning
  - Analyzing complex strategic interactions