Sequential Decision Making and Reinforcement Learning

(SDMRL)

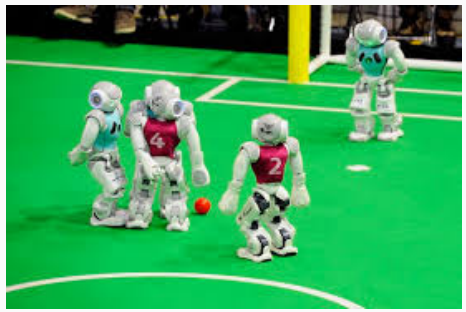
Conclusion

Sarah Keren

The Taub Faculty of Computer Science
Technion - Israel Institute of Technology

Multi-Agent AI

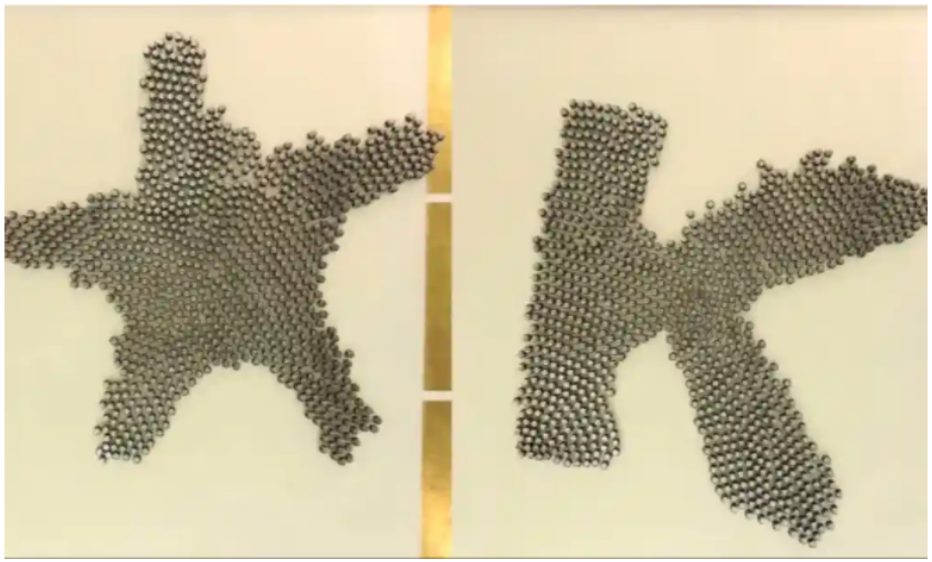
Challenges in multi-agent coordination



Challenges in multi-agent coordination

Harvard's 1,000 Kilobot swarm demonstrates the future of robotics

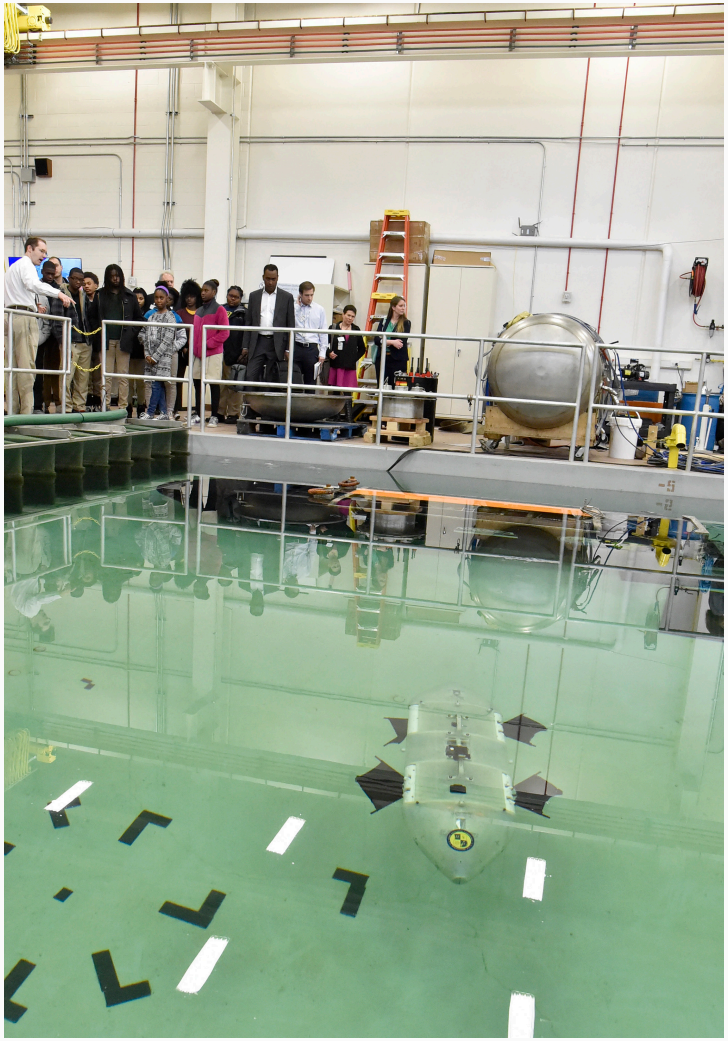
A new system has replicated the hive mind with over 1,000 simple robots collaborating without human control



Harvard's new Kilobots swarm and cooperate in a 1,000-strong mass to create complex shapes with no micromanagement. Photograph: Harvard Photograph: Harvard

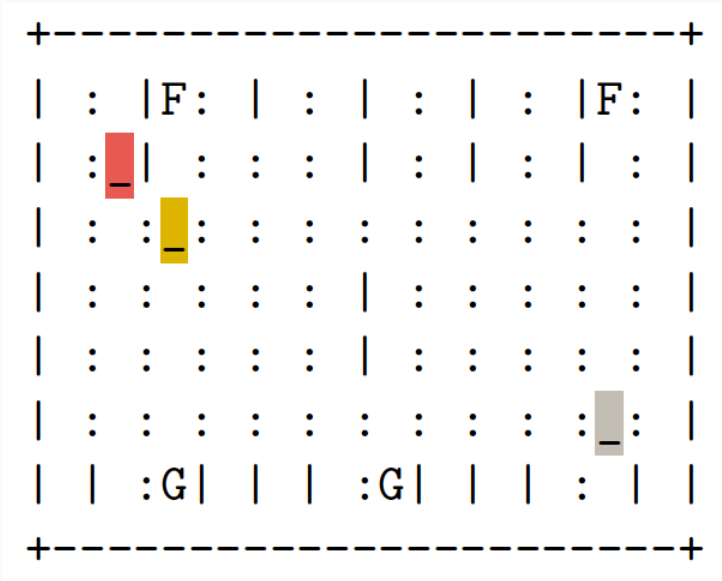
From Radhika Nagpal's lab.

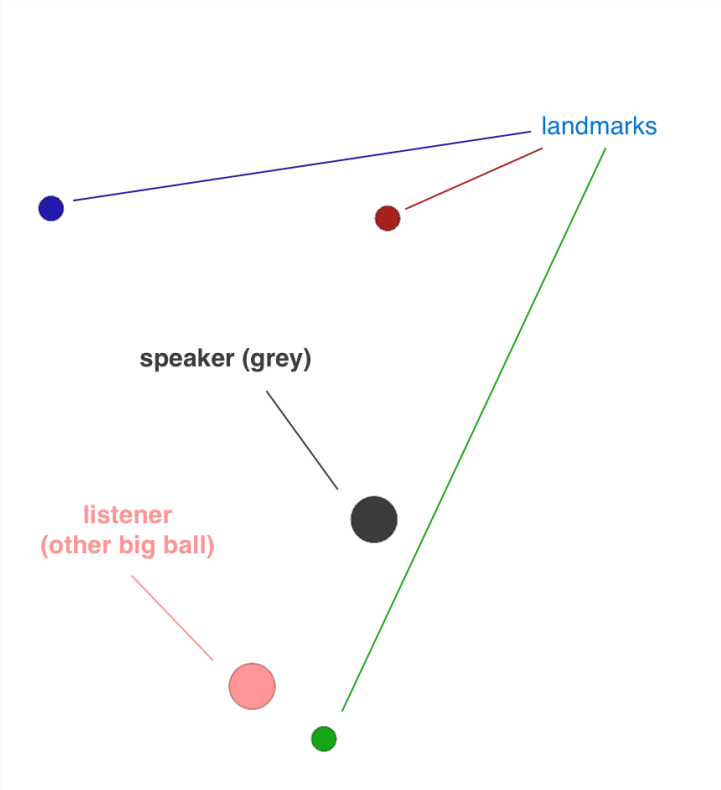
Challenges in multi-agent coordination



Naval Research Lab

Challenges in multi-agent coordination





Challenges in multi-agent coordination

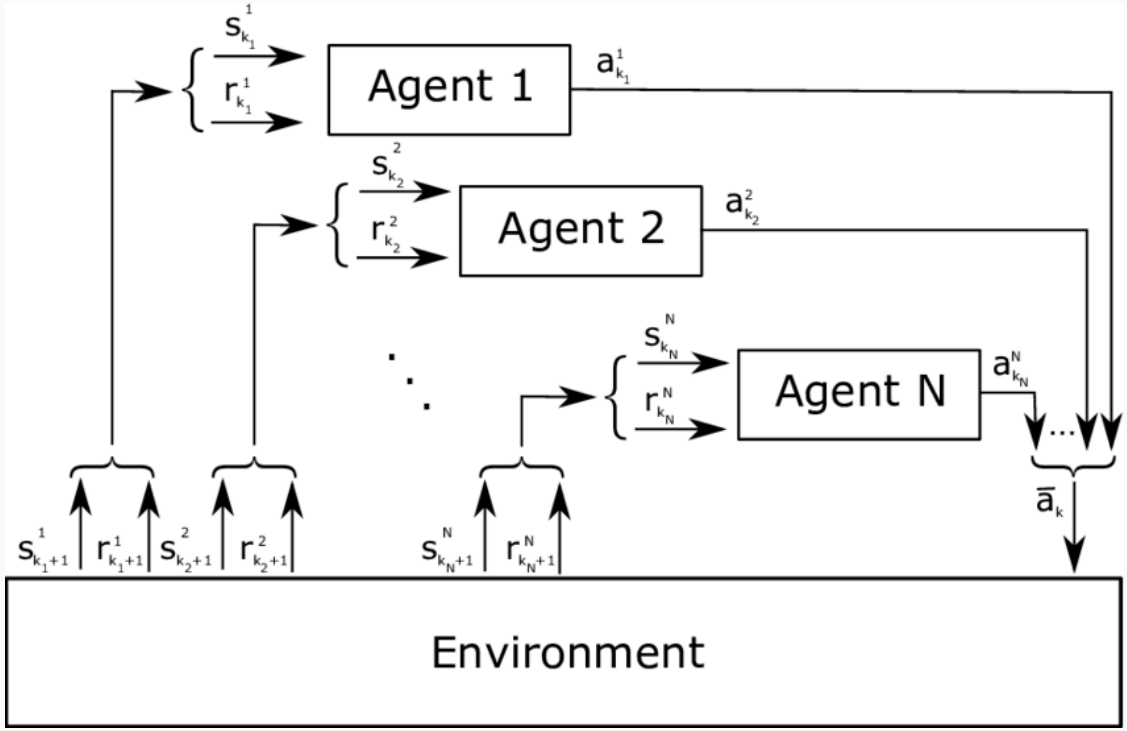
- Computational complexity and generality (even in full communication and centralized control settings)
- Communication Limitations
  - Low bandwidth
  - Unpredictably lossy networks
- Joint planning
  - Efficient coordination
  - Imperfect information

Why Multiple Agents?

- **Parallelism:** Many robots can accomplish the task faster
- **Redundancy:** Hazardous environment with chances of losing robots
- **Required:** Too difficult to do with a single size robot
- **Complex Tasks:** Need several specialized robots
- **Real-time Requirements:** Monitor large areas, respond quickly

Sometime it's simply a constraint imposed by the environment.

The Multi-agent-Environment Interface



To model a multi-agent setting, We will use a *Markov game*, or *stochastic game* which is a generalization of the MDP to multi-agent settings (Littman'94).

Image by Michele Chincoli and Antonio Liotta

Multi-Agent Markov Decision Process - Markov Games

A **Markov game** is defined by the tuple $\langle \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma \rangle$ where

- $\mathcal{S}$ is a finite set of states
- $\mathcal{A} = \{\mathcal{A}^i\}_{i=1}^n$ are the **joint actions**: a collection of action sets $\mathcal{A}^i$, one for each agent in the environment. At each timestep t, each agent i chooses an action $a_t^i \in \mathcal{A}^i$. The actions of all N agents are combined to form a **joint action** $\mathbf{a}_t = [a_t^0, \ldots, a_t^N]$.
- $\mathcal{P}$ describes the probability distribution over next states when a joint action is performed and produces a transition in the environment, where $\mathcal{P}_{s,s'}^{\mathbf{a}} = \mathcal{P}[S_{t+1} = s' | S_t = s, A_t = \mathbf{a}_t]$ describes the probability of ending at state s' when the joint action $\mathbf{a}_t \in \mathcal{A}$ is performed at state s.
- $\mathcal{R} = \{\mathcal{R}^i\}_{i=1}^n$ is a collection of rewards functions $\mathcal{R}^i$ defining the reward $r^i(a_t, s_t)$ each agent receives when the joint action a_t is performed at state s_t,
- γ is a discount factor $\gamma \in [0, 1]$.

Multi-Agent Markov Decision Process - Markov Games

What about partially observability?

Multi-Agent Markov Decision Process - Markov Games

In partially observable environments the ith agent can only view a portion of the true state, s_t^i.



Multi-Agent POMDP

- A **Multi-Agent POMDP** can be defined by the tuple $\langle \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma, \mathfrak{O}, \mathfrak{S}, \mathfrak{B}_0 \rangle$ where
  - $\mathcal{S}$, $\mathcal{A}$, $\mathcal{P}$, $\mathcal{R}$ and γ are as in the definition of the Markov Game,
  - $\mathfrak{O} = \{O^i\}_{i=1}^n$ are the **joint observations**: a collection of observation token sets O^i, one for each agent in the environment. At each timestep t, each agent i observes an observation $o_t^i \in O^i$. The observations of all N agents are combined to form a **joint observation** $\mathbf{o}_t = [o_t^0, \ldots, o_t^N]$.
  - $\mathfrak{S}$ is the **sensor function** describing the probability distribution over next joint observations when a joint action is performed in some state, where $\mathfrak{S}^{\mathbf{o}}_{s,\mathbf{a}} = \mathcal{P}[\mathbf{O}_{t+1} = \mathbf{o} | S_t = s, A_t = \mathbf{a}_t]$ describes the probability of receiving joint observation $\mathbf{o}$ when the joint action $\mathbf{a}_t$ is performed at state s (alternatively: $\mathfrak{S}^{\mathbf{o}}_{s} = \mathcal{P}[\mathbf{O}_{t+1} = \mathbf{o} | S = s]$.)
  - $\mathfrak{B}_0 = \{b_0^i\}_{i=1}^n$ is the initial joint belief, specifying for each agent i the probability distribution over states such that $b_0^i(s)$ stands for the probability agent i associates with s being the true initial state.

Dimensions of Multi-Agent Systems

- Control
- Decision-making approach
- Communication
- Observability
- Shared resources/interaction
- Objectives/utilities

Dimensions of Multi-Agent Systems

- **Control:** from centralized to decentralized.
- **Decision-making approach:** planning, learning, RL
- **Communication:** from no communication to limitless communication
- **Observability:** no sensing (conformant) to full information
- **Shared resources/interaction:** space, energy, etc.
- **Objectives/utilities:** from collaborative to adversarial

Multi-agent AI

Distributed AI - centralized design - a single body that can influence the preferences of all agents in the system.

Multi-agent AI - individual agents don't have a group sense of utility.

Dimensions of Multi-Agent Systems

- Control
- Decision-making approach
- Communication
- Observability
- Shared resources/interaction
- Objectives/utilities

Dimensions of Multi-Agent Systems

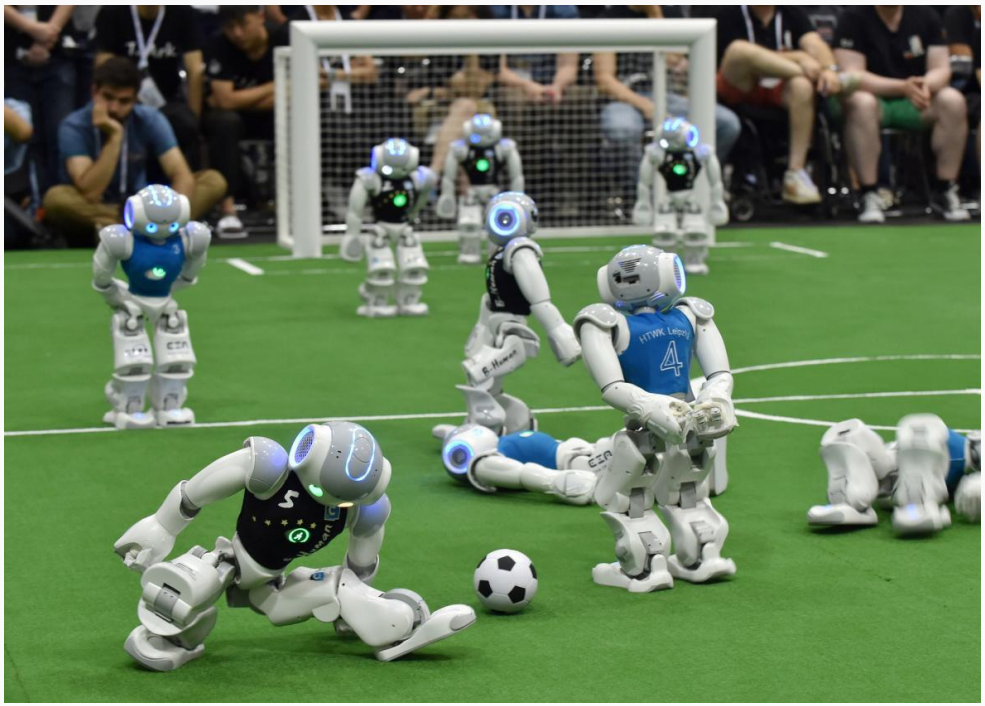
- **Control:** from centralized to decentralized.
- **Decision-making approach:** planning, learning, RL
- **Communication:** from no communication to limitless communication
- **Observability:** no sensing (conformant) to full information
- **Shared resources/interaction:** space, energy, etc.
- **Objectives/utilities:** from collaborative to adversarial

Autonomous Vehicles

- Control - Centralized/Decentralized/Hierarchical/Hybrid?
- Communication - Implicit/Passive Action Recognition/Explicit?
- Objectives
- Shared Resources
- Observability - What information is available to every vehicle?

Robotic soccer

- Control - Centralized/Decentralized/Hierarchical/Hybrid?
- Communication - Implicit/Passive Action Recognition/Explicit?
- Objectives
- Shared Resources
- Observability - What information is available to every agent?



Search and Rescue

- Control - Centralized/Decentralized/Hierarchical/Hybrid?
- Communication - Implicit/Passive Action Recognition/Explicit?
- Objectives
- Shared Resources
- Observability - What information is available to every agent?

Human-robot Collaboration

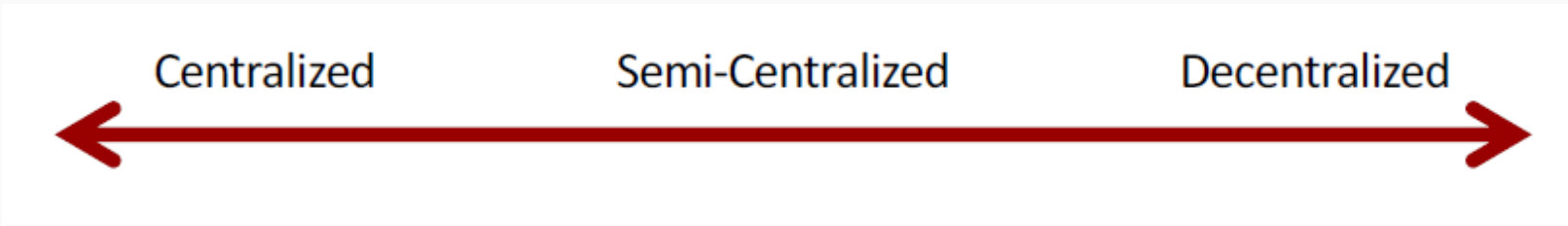
- Control - Centralized/Decentralized/Hierarchical/Hybrid?
- Communication - Implicit/Passive Action Recognition/Explicit?
- Objectives
- Shared Resources
- Observability - What information is available to each agent?

Amazon Warehouses

Movie: Amazon Warehouses

- Control - who controls the robots?
- Decision-making approach - how robots choose their actions?
- Communication - how robots communicate with each other?
- Shared resources - what resources are common to all robots?
- Objectives - what is the goal of the robots as a group?
- Robustness - how robust is the system to failures?

Architectures for Control



Architectures for Control

- Centralized
  - Global controller with global view
  - Good for tightly-coupled tasks, efficiency, adversarial
  - Good for small teams or highly structured environments
  - Requires: high bandwidth/computation/sensing (at least for the leader)
- Middle Ground: Semi-Centralized
  - Try to approximate the effect of a centralized system
  - Supervisor and Team (supervisor acts as global controller)
  - Hive-based (homebase or rendevous to deposit information)
  - Role-based coordination (pre-decide responsibilities)
  - When? Communication is available but slow or limited range.
- Decentralized
  - No one has a full world view (peer-to-peer system)
  - Independent acting robots (purely local or no communication)
  - Good for large distributed teams (no centralized bottleneck/failure)
  - Often biologically-inspired (swarm intelligence)

Resources

- *https://rlss.inria.fr/files/2019/07/RLSS_Multiagent.pdf*

Multi-agent AI

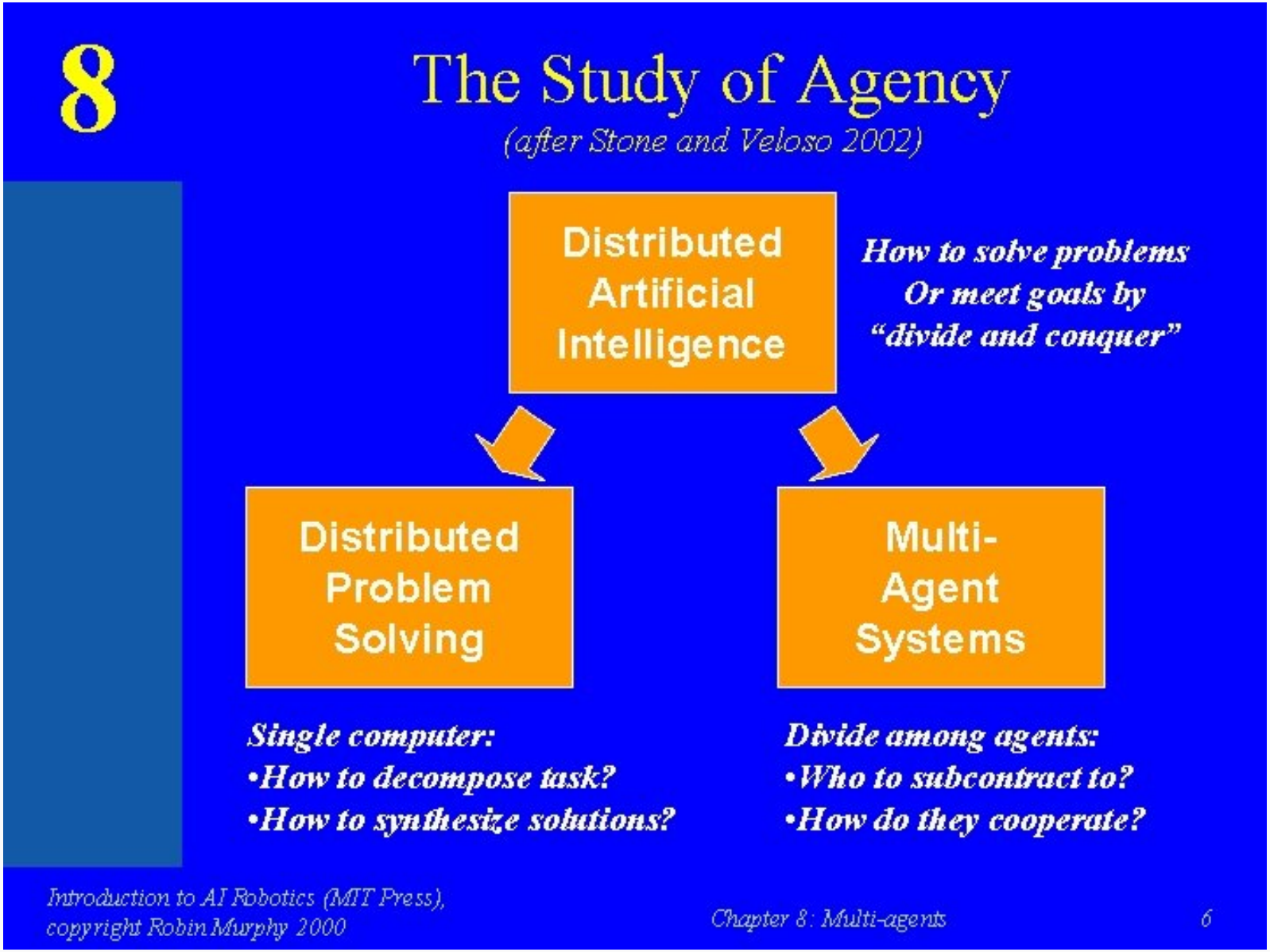
Distributed AI - centralized design - a single body that can influence the preferences of all agents in the system.

Multi-agent AI - individual agents don't have a group sense of utility.

Dimensions of Multi-Agent Systems

- Control
- Decision-making approach
- Communication
- Observability
- Shared resources/interaction
- Objectives/utilities

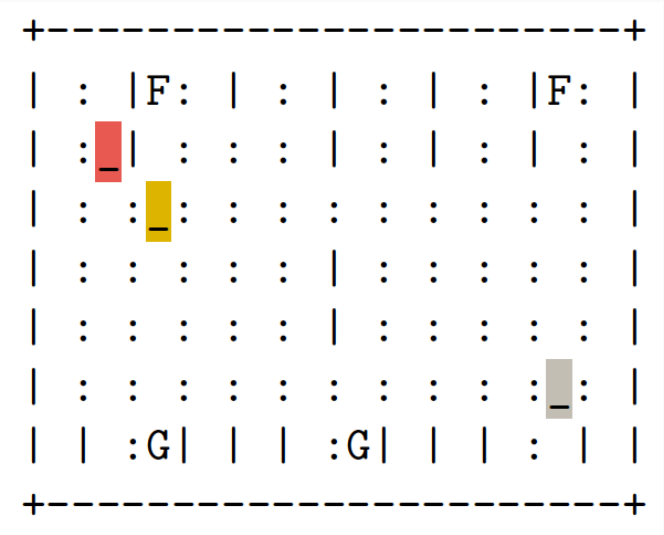
Architectures of Multi-agent Systems

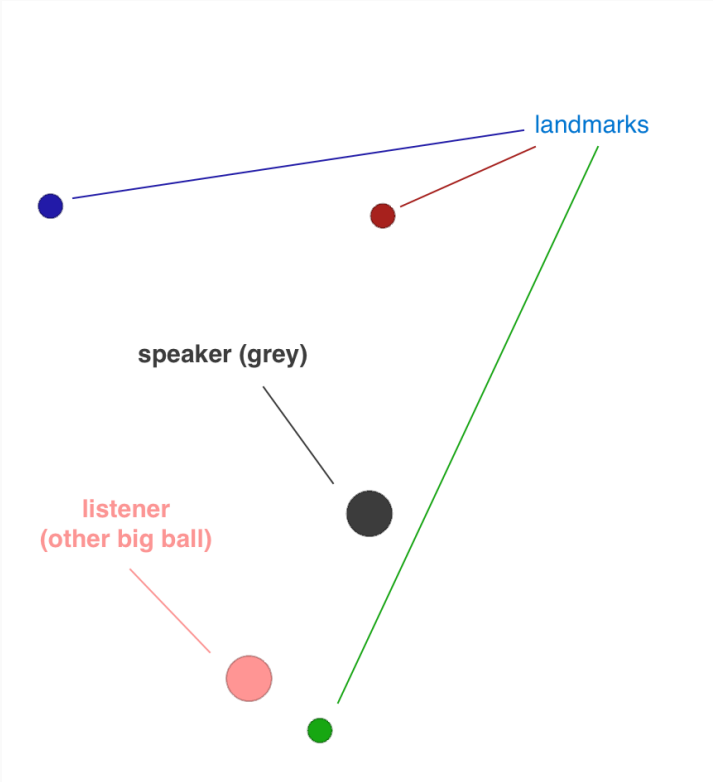


Multiple Agents Introduce Multiple Difficulties

- Communication
- Collisions
- Coordination
- Task Allocation
- Conflicting Objectives

Example Domains





Multi-Agent Planning

Multi-Agent Markov Decision Process - Markov Games

A **Markov game** is defined by the tuple $\langle \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma \rangle$ where

- $\mathcal{S}$ is a finite set of states
- $\mathcal{A} = \{\mathcal{A}^i\}_{i=1}^n$ are the **joint actions**: a collection of action sets $\mathcal{A}^i$, one for each agent in the environment. At each timestep t, each agent i chooses an action $a_t^i \in \mathcal{A}^i$. The actions of all N agents are combined to form a **joint action** $\mathbf{a}_t = [a_t^0, \ldots, a_t^N]$.
- $\mathcal{P}$ describes the probability distribution over next states when a joint action is performed and produces a transition in the environment, where $\mathcal{P}_{s,s'}^{\mathbf{a}} = \mathcal{P}[S_{t+1} = s' | S_t = s, A_t = \mathbf{a}_t]$ describes the probability of ending at state s' when the joint action $\mathbf{a}_t \in \mathcal{A}$ is performed at state s.
- $\mathcal{R} = \{\mathcal{R}^i\}_{i=1}^n$ is a collection of rewards functions $\mathcal{R}^i$ defining the reward $r^i(a_t, s_t)$ each agent receives when the joint action a_t is performed at state s_t,
- γ is a discount factor $\gamma \in [0, 1]$.

Multi-Agent POMDP

- A **Multi-Agent POMDP** can be defined by the tuple $\langle \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma, \mathfrak{O}, \mathfrak{S}, \mathfrak{B}_0 \rangle$ where
  - $\mathcal{S}$, $\mathcal{A}$, $\mathcal{P}$, $\mathcal{R}$ and γ are as in the definition of the Markov Game,
  - $\mathfrak{O} = \{O^i\}_{i=1}^n$ are the **joint observations**: a collection of observation token sets O^i, one for each agent in the environment. At each timestep t, each agent i observes an observation $o_t^i \in O^i$. The observations of all N agents are combined to form a **joint observation** $\mathbf{o}_t = [o_t^0, \ldots, o_t^N]$.
  - $\mathfrak{S}$ is the **sensor function** describing the probability distribution over next joint observations when a joint action is performed in some state, where $\mathfrak{S}^{\mathbf{o}}_{s,\mathbf{a}} = \mathcal{P}[\mathbf{O}_{t+1} = \mathbf{o} | S_t = s, A_t = \mathbf{a}_t]$ describes the probability of receiving joint observation $\mathbf{o}$ when the joint action $\mathbf{a}_t$ is performed at state s (alternatively: $\mathfrak{S}^{\mathbf{o}}_{s} = \mathcal{P}[\mathbf{O}_{t+1} = \mathbf{o} | S = s]$.)
  - $\mathfrak{B}_0 = \{b_0^i\}_{i=1}^n$ is the initial joint belief, specifying for each agent i the probability distribution over states such that $b_0^i(s)$ stands for the probability agent i associates with s being the true initial state.

Multi-Agent Planning

- Devising effective action policies or strategies for a set of n agents
- In **fully collaborative** settings, the key challenge is coordinating the actions of the individual agents so that the shared goals are achieved efficiently
  - Many of the interesting problems in cooperative game theory (such as coalition formation and negotiation) disappear.
  - Rather it becomes more like a standard (one-player) decision problem, where the collection of n players can be viewed as a single player trying to optimize its behavior against nature.

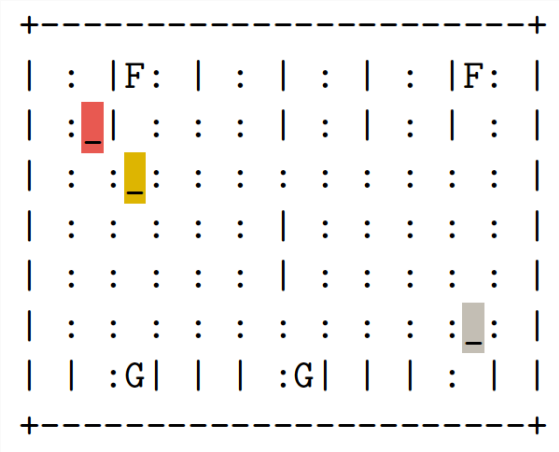
Planning, Learning and Coordination in Multiagent Decision Processes February 1970 Authors: Craig Boutilier Google Inc.

A Market Approach to Multirobot Coordination. Dias et al. 2001

Market-Based Multirobot Coordination: A Survey and Analysis. Dias et al. 2006

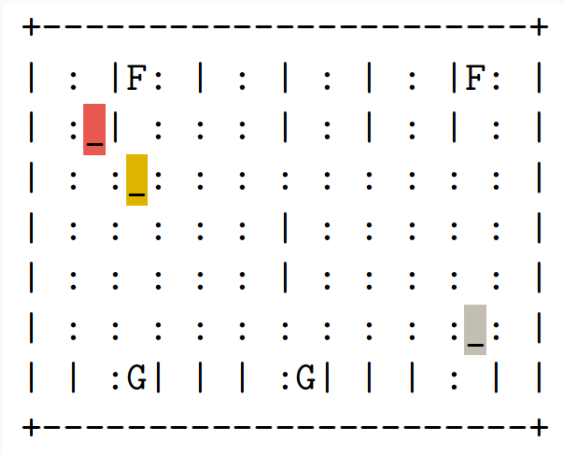
https://ieeexplore.ieee.org/document/1677943

Centralized Multi-Taxi

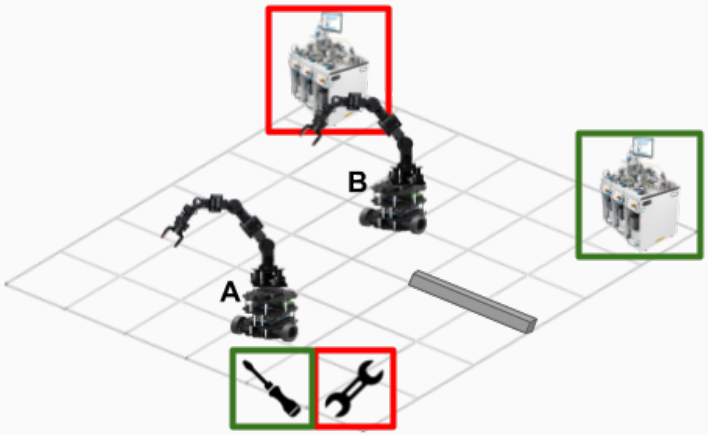


- What is the joint state space and action space?
- What are the assumptions made ? What kind of interactions/ collaborations we support?

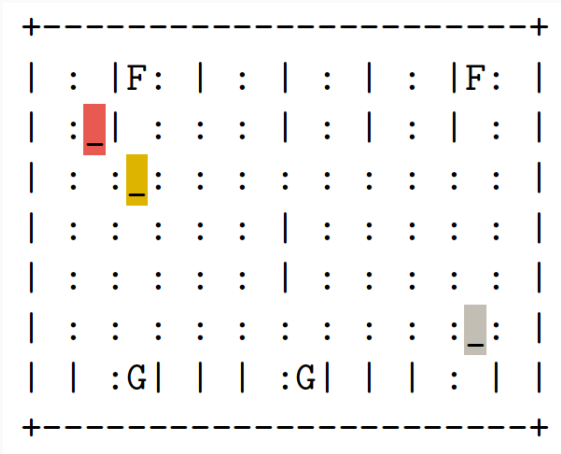
Centralized Multi-Taxi



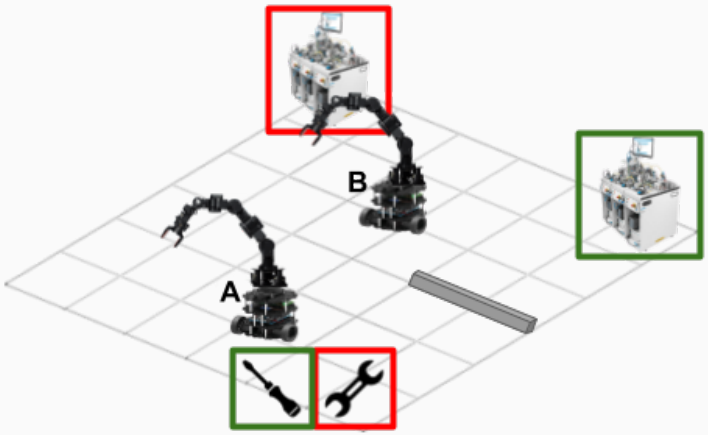
- What is the joint state space and action space?
- What are the assumptions made ? What kind of interactions/ collaborations we support?

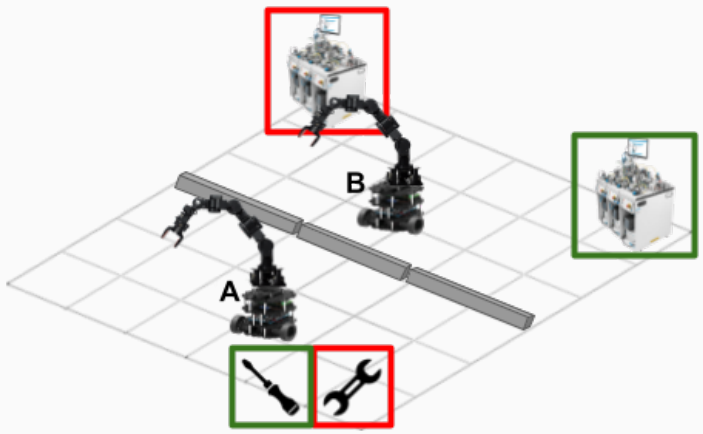


Centralized Multi-Taxi



- What is the joint state space and action space?
- What are the assumptions made ? What kind of interactions/ collaborations we support?





Multi-Agent Planning

- A single agent with joint actions at its disposal allows one to compute (or learn) optimal joint policies by standard methods, provided the agents are of "one mind."
- However:

Multi-Agent Planning

- A single agent with joint actions at its disposal allows one to compute (or learn) optimal joint policies by standard methods, provided the agents are of "one mind."
- However:
  - In many settings, agents are self interested
    - may be motivated by non-aligned or conflicting objectives

Multi-Agent Planning

- A single agent with joint actions at its disposal allows one to compute (or learn) optimal joint policies by standard methods, provided the agents are of "one mind."
- However:
  - In many settings, agents are self interested
    - may be motivated by non-aligned or conflicting objectives
  - Even in collaborative settings, we expect agents to plan or learn independently, with a limited ability to share information.
    - However, choices made separately may be jointly suboptimal.
    - Need methods for coordination: we must ensure that the individual decisions made can be coordinated so that joint optimality is achieved

Multi-Agent Planning

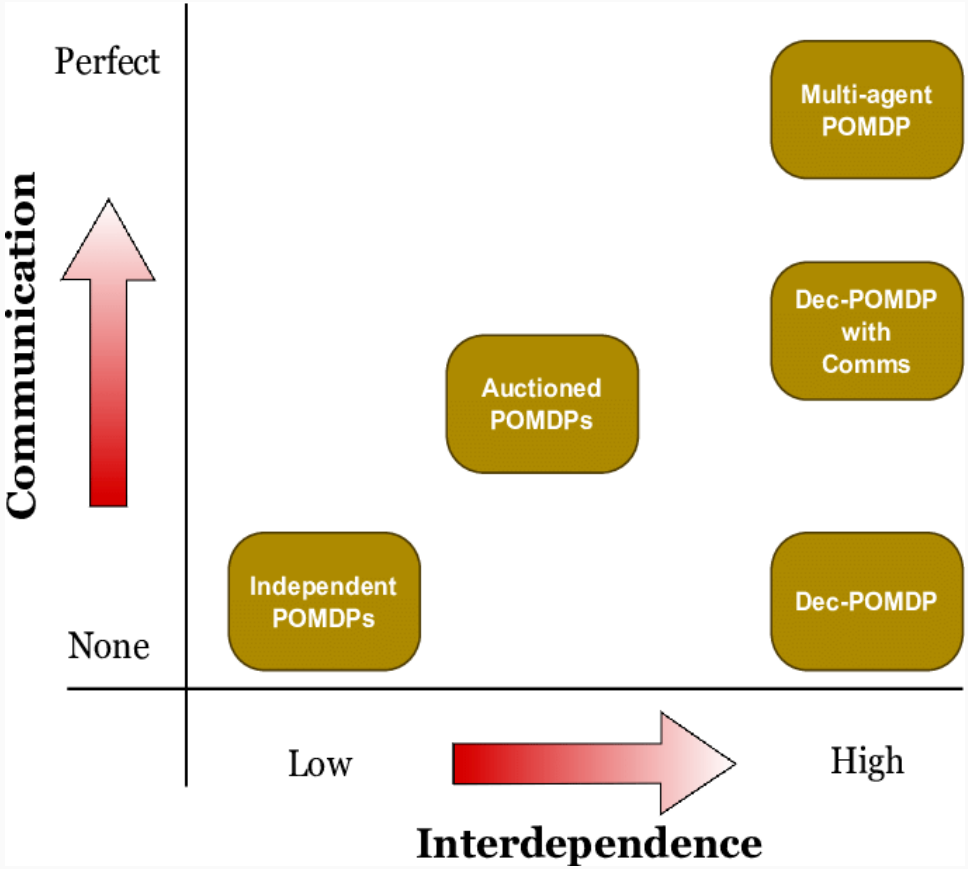
- A single agent with joint actions at its disposal allows one to compute (or learn) optimal joint policies by standard methods, provided the agents are of "one mind."
- However:
  - In many settings, agents are self interested
    - may be motivated by non-aligned or conflicting objectives
  - Even in collaborative settings, we expect agents to plan or learn independently, with a limited ability to share information.
    - However, choices made separately may be jointly suboptimal.
    - Need methods for coordination: we must ensure that the individual decisions made can be coordinated so that joint optimality is achieved
  - Even in fully-collaborative settings optimal coordination is computationally difficult:
    - best known algorithms are exponential in complexity.
    - approaches striving to compute globally optimal solutions become intractable for teams larger than a few

Multi-Agent Coordination

- Solutions to the coordination problem can be divided into four general classes:
  - **communication-based** (e.g., agents might communicate in order to determine task allocation)
  - **convention-based** (e.g., social laws might be imposed by the system to avoid collisions)
  - **learning-based** (e.g., coordinated policy learned through repeated interaction)
  - **market-based architecture** (e.g., form centralized sub-groups to improve efficiency)

We will examine each in the coming weeks.

Multi-Agent Planning



https://www.researchgate.net/publication/221456711_Decentralized_decision_support_for_an_agent_population_in_dynamic_and_uncertain_domains

Multi-agent RL

References

- *https://cse.buffalo.edu/~avereshc/rl_fall19/lecture_24_MARL.pdf*
- Stefano V. Albrecht and Filippos Christianos and Lukas Schäfer, Multi-Agent Reinforcement Learning: Foundations and Modern Approaches, MIT Press, 2023, *https://www.marl-book.com*

What is Multi-Agent RL (MARL)

- In **Single-agent RL (RL)** a single agent learns an optimal good decision policies by trying actions and receiving rewards (typically) with the objective of maximising the sum of received rewards over time.
- In **Multi-agent RL (MARL)** the focus is on learning optimal policies for **multiple** agents and on facing the unique challenges that arise in this learning process.

MARL Setup

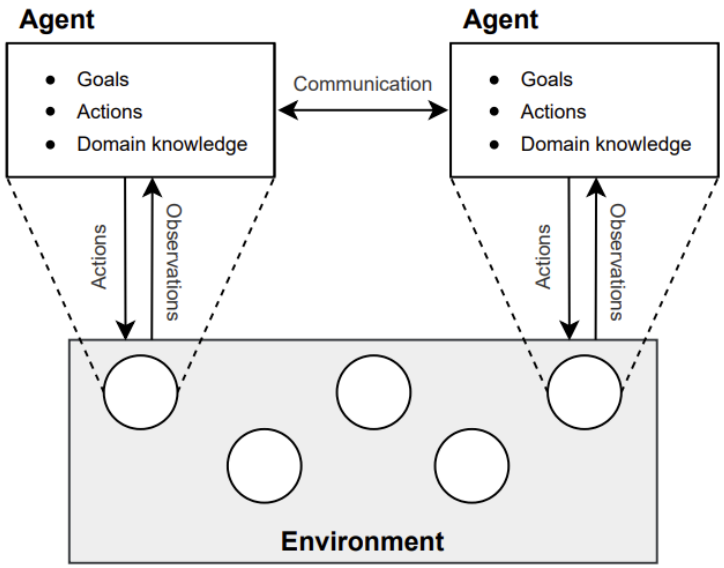


Figure 1.1: Schematic of a multi-agent system. A multi-agent system consists of an environment and multiple decision-making agents. The agents can observe information about the environment and take actions to achieve their goals.

Image from *https://www.marl-book.com/*

Environment:

- physical or virtual
- state evolves over time and is influenced by agent actions
- specifies the actions each agent can take at any point in time, as well as the observations that individual agents receive about the state of the environment

MARL Setup

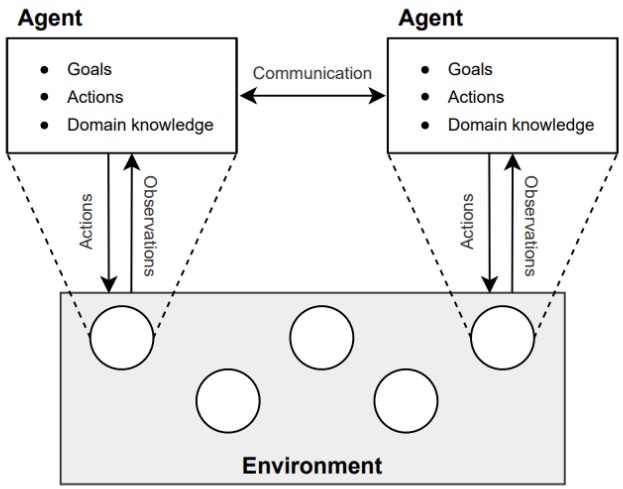


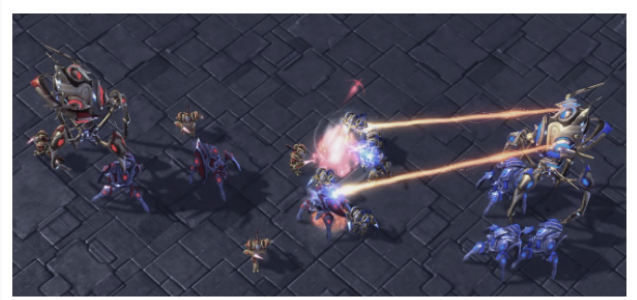
Figure 1.1: Schematic of a multi-agent system. A multi-agent system consists of an environment and multiple decision-making agents. The agents can observe information about the environment and take actions to achieve their goals.

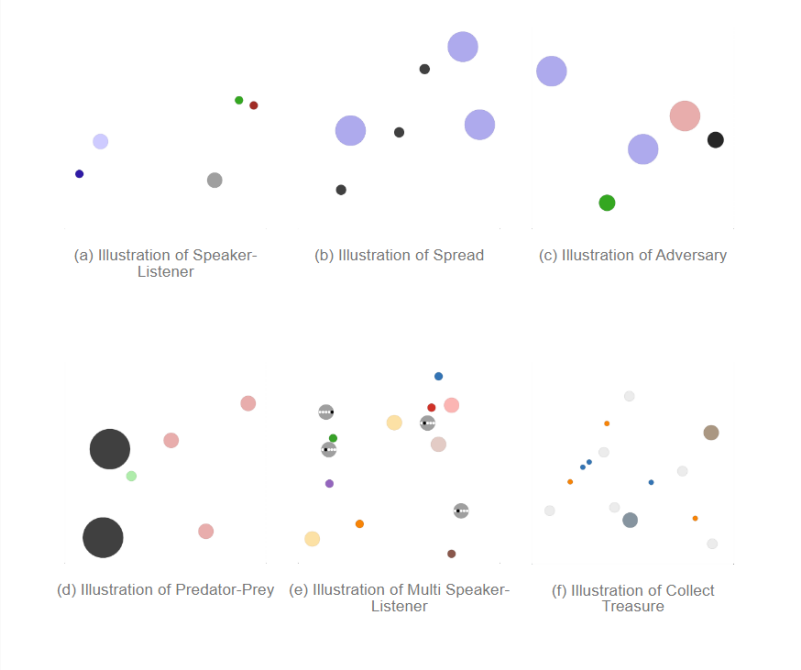
Image from *https://www.marl-book.com/*

Agent:

- an entity that receives information about the state of the environment and can choose different actions in order to influence the state
- goal-directed - chooses actions in order to achieve the objective, typically specified via a scalar reward signal receive after taking certain actions in certain states
- In MARL, agents may have different prior knowledge about the environment and different objectives

Example of MARL benchmarks





see *https://pettingzoo.farama.org/* and *https://agents.inf.ed.ac.uk/blog/multiagent-learning-environments/* for more settings

MARL Dynamics

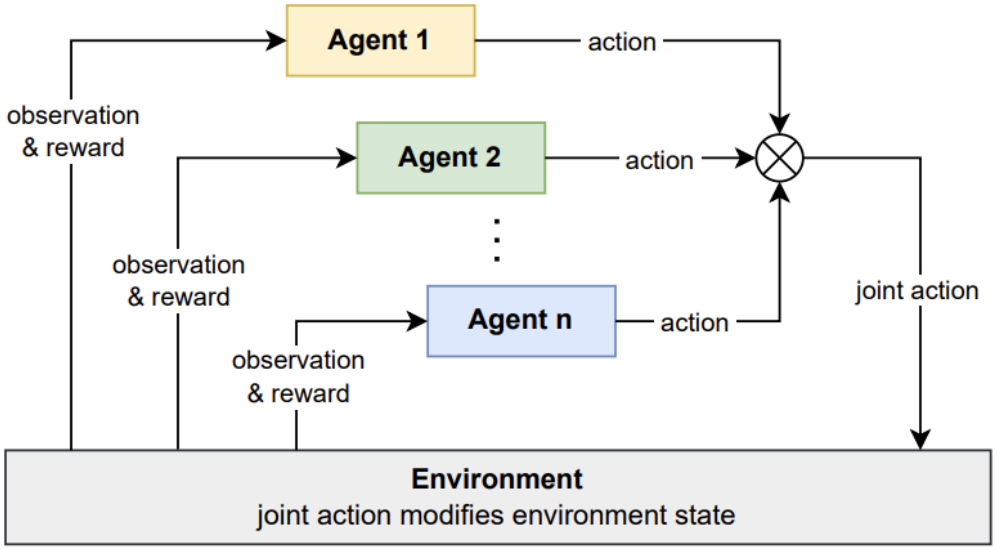


Figure 1.3: Schematic of multi-agent reinforcement learning. A set of n agents receive individual observations about the state of the environment, and choose actions to modify the state of the environment. After taking an action, each agent receives a scalar reward and a new observation, and the loop repeats.

Reminder: Dimensions of Multi-Agent Systems

- Control
- Decision-making approach
- Communication
- Observability
- Shared resources/interaction
- Objectives/utilities

Reminder: Multi-Agent MDP - Markov Games

A **Multi-Agent MDP** or **Markov game** is defined by the tuple $\langle \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma \rangle$ where

- $\mathcal{S}$ is a finite set of states
- $\mathcal{A} = \{\mathcal{A}^i\}_{i=1}^n$ are the **joint actions**: a collection of action sets $\mathcal{A}^i$, one for each agent in the environment.
  - At each timestep t, each agent i chooses an action $a_t^i \in \mathcal{A}^i$. The actions of all N agents are combined to form a **joint action** $\mathbf{a}_t = [a_t^0, \ldots, a_t^N]$.
- $\mathcal{P}$ describes the probability distribution over next states when a joint action is performed and produces a transition in the environment, where $\mathcal{P}^{\mathbf{a}}_{s,s'} = \mathcal{P}[S_{t+1} = s' | S_t = s, A_t = \mathbf{a}_t]$ describes the probability of ending at state s' when the joint action $\mathbf{a}_t \in \mathcal{A}$ is performed at state s.
- $\mathcal{R} = \{\mathcal{R}^i\}_{i=1}^n$ is a collection of rewards functions $\mathcal{R}^i$ defining the reward $r^i(a_t, s_t)$ each agent receives when the joint action a_t is performed at state s_t,
- γ is a discount factor $\gamma \in [0, 1]$.

Reminder: Factored State Space Representations

Instead of an explicit representation of the state space $\mathcal{S}$, it is common to use *factored state representations**, where the set of states is described via a set of random variables $X = X_1, \ldots, X_n$, and where each variable X_i takes on values in some finite domain $Dom(X_i)$. A state is an assignment of a value $X_i \in Dom(X_i)$ for each variable X_i.

What's the benefit of using a factored representation ?

*Boutilier, Craig, Richard Dearden, and Moisés Goldszmidt. "Stochastic dynamic programming with factored representations." Artificial intelligence 121.1-2 (2000): 49-107 and Guestrin, C., Koller, D., Parr, R., and Venkataraman, S. (2003). Efficient solution algorithms for factored MDPs. Journal of Artificial Intelligence Research, 19, 399-468.

MARL Challenges

Challenges of MARL

Various challenges stem from agents having conflicting goals and different partial views of their environment, and from agents learning concurrently to optimize their policies.

Challenges of MARL

Non-stationarity

- continually changing policies of learning agents
- **moving target problem:** each agent adapts to the policies of other agents whose policies in turn also adapt to changes in other agents - cyclic and unstable learning dynamics.

Challenges of MARL

Multi-agent credit assignment -

- determining which past actions contributed to a received reward is hard for single agent settings.
- In MARL, the additional problem of determining whose action contributed to the reward.
- Ideas based on **counterfactual reasoning** can address this problem in principle, but still an open problem.

Challenges of MARL

Optimality of policies and equilibrium selection

- In single-agent RL, a policy is optimal if it achieves maximum expected returns.
- In MARL, the returns of one agent's policy also depends on the other agents' policies.
- **Equilibrium** means that no individual agent can deviate from its policy in the solution to improve its outcome.
- Range of solution concepts:
  - Most anchored in some notion of equilibrium and seek an **equilibrium joint-policy** in which gent's policies are in some (specific) sense optimal with respect to the other's policies.
  - May be multiple equilibrium solutions, and each equilibrium may entail different returns to different agents.
  - Additional challenge of agents having to negotiate during learning which equilibrium to converge to (Harsanyi and Selten 1988).
  - **A central goal of MARL:** develop learning algorithms which robustly converge to a particular solution type.

Challenges of MARL

Scaling in number of agents

- Total number of possible action combinations between agents may grow exponentially with the number of agents.
- Particularly if each added agent comes with its own additional action variables and affects other agents.
- In the past, it was common to use only two agents, but even with today's deep learning-based MARL, it is common to use 2-10 agents.
- How to handle many more agents in an efficient and robust way is an important goal in MARL research.

Solution Approaches

Approaches for overcoming MARL challenges

- Solutions to the coordination problem can be divided into four general classes:
  - **communication-based**
  - **convention-based**
  - **learning-based**
  - **market-based architecture**

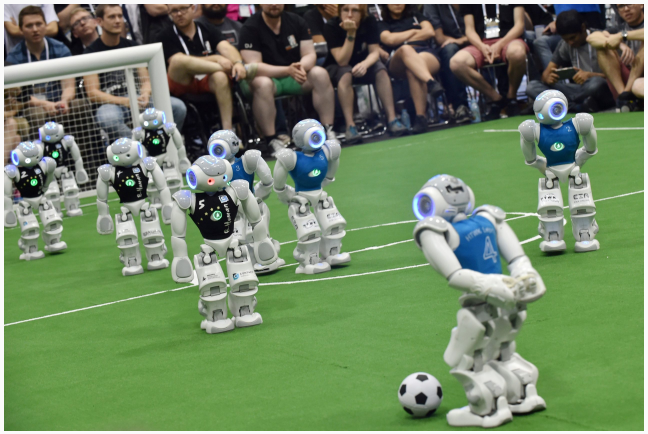
  We already saw these when discussing multi-agent AI more broadly. What are the specific challenges to MARL?

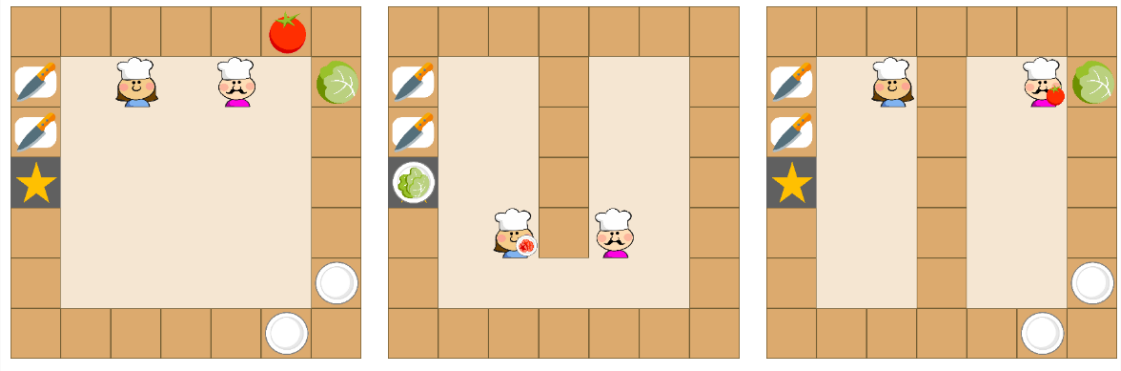
Approaches for overcoming MARL challenges

- Because of the rich range of frameworks and settings, there is a variety of solution approaches to MARL.
- We will cover a very small part of them.
- We will focus on non-adversarial domains.

Coordination

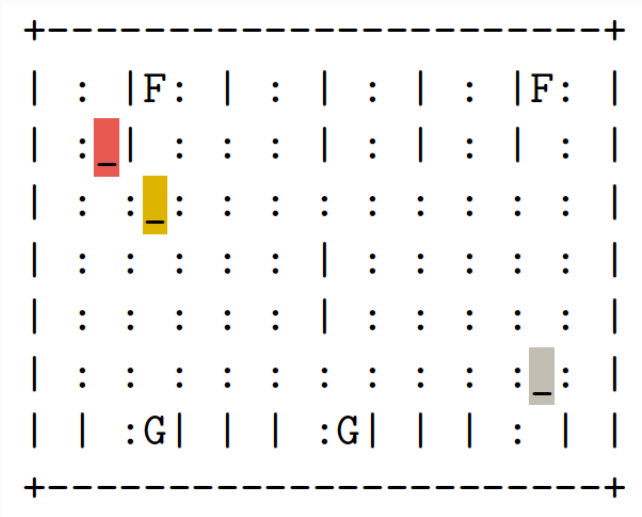
How do agents, that share some common objective, coordinate their actions to avoid collisions and maximize overall performance?

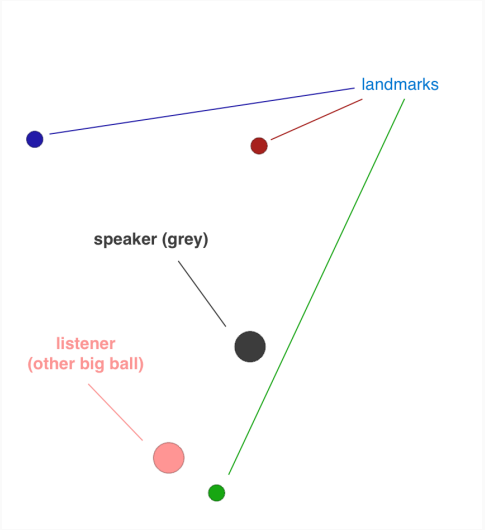




Overcooked from Wang et al. 2020: *https://arxiv.org/pdf/2003.11778.pdf*

Coordination





How to achieve effective coordination?

Multi-Agent Coordination (by Ronen Brafman)

- Most work in multi-agent coordination focuses on **swarms** - a homogeneous group of agents with few simple capabilities.
- Very few approaches for heterogeneous teams with cognitive agents that
  - have diverse skills
  - need to operate in a stochastic environment
  - need to adapt to changes in their environment while depending on their (possibly noisy) sensors and limited and noisy communication with other agents [1]

[1] See AAAI Sprint Symposium: Can We Talk?' How to Design Multi-Agent Systems In the Absence of Reliable Communications.

Multi-Agent Coordination (by Ronen Brafman)

- Two general approaches:
  - **Model-based approaches:** A model of the environment and the effect actions have on it is given, but agents only have partial knowledge of the current state of the world.
  - **Reinforcement learning approaches:** don't assume agents know the effect of their actions on the environment.
- The most popular model is a **decentralized partially observable Markov decision process (Dec-POMDP)** which is a special case of a multi-agent POMDP in which agents share a common reward function.
- We can distinguish between offline and online approaches
  - **Offline:** a fixed policy is determined for each agent
  - **Online:** agents start with some initial policy but may update it as new information is received.

Multi-Agent Coordination (by Ronen Brafman)

- Solutions to the coordination problem can be divided into four general classes we have discussed:
  - **communication-based** (e.g., agents might communicate in order to determine task allocation)
  - **convention-based** (e.g., social laws might be imposed by the system to avoid collisions)
  - **learning-based** (e.g., coordinated policy learned through repeated interaction)
  - **market-based architecture** (e.g., form centralized sub-groups to improve efficiency)

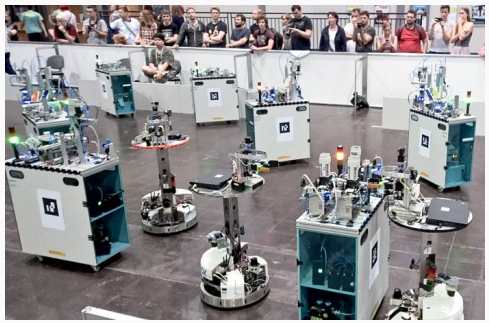
Communication based Coordination

Three ways to communicate:

- **Explicit Communication** - agents communicate relevant information about their actions
- **Implicit Communication** - agents communicate through the world (stigmergy), they sense the effects of teammates actions through their effects on the world.
- **Passive Action Recognition** - agents use sensors to directly observe the actions of their teammates. .

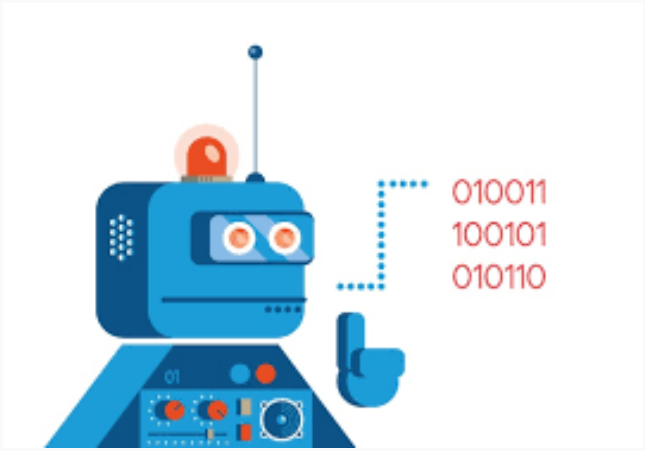
Limited Communication

- Impossible or undesirable for decision-makers to share all their knowledge all the time.
- Exchanging information may incur a cost associated with the required bandwidth or with the risk of revealing it to competing agents.
- Communication may not be reliable

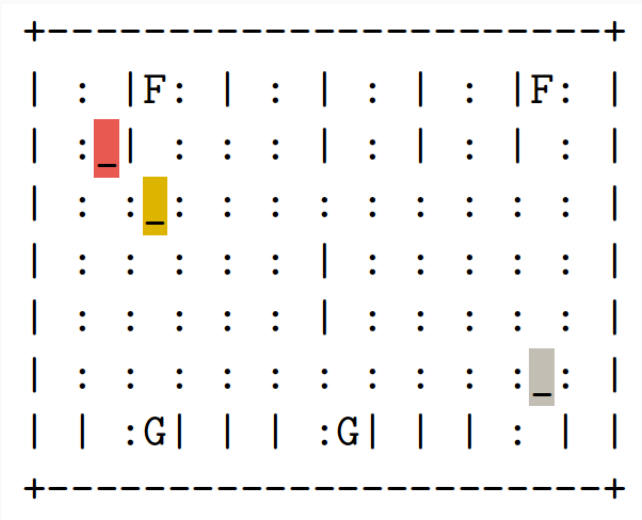


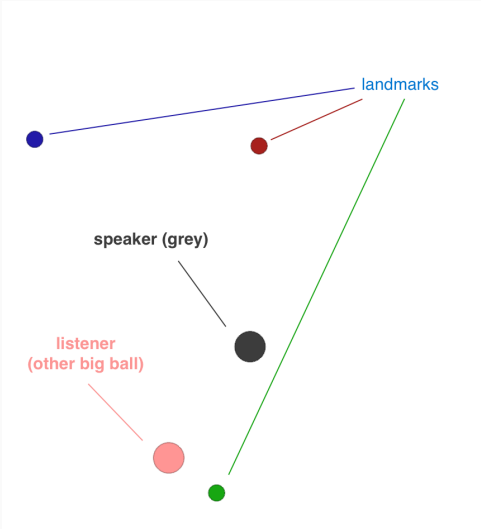
Fixed vs. Emergent Communication

- Fixed semantics
  - question of when still needs to be resolved.
- Emergent communication
  - ”Language derives meaning from its use” (Wittgenstein, 1953)



Example Domains





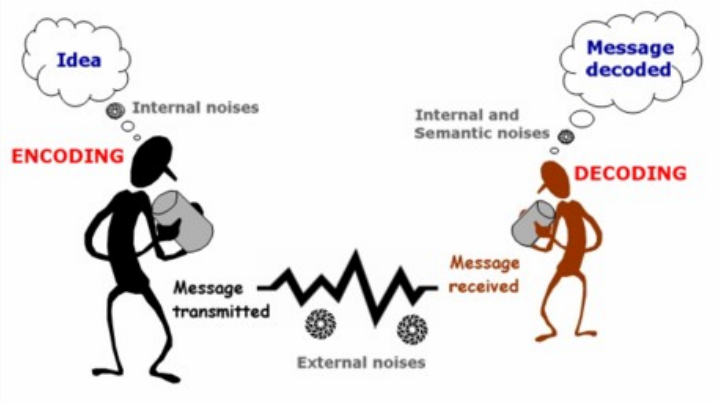
How to effectively communicate to coordinate?

Modeling Communication in Multi-Agent POMDPs

- **Multi-Agent POMDP** $\langle \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma \rangle$
- What about communication?

Modeling Communication in Multi-Agent POMDPs

- **Multi-Agent POMDP** $\langle \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma \rangle$
- What about communication?



Two general approaches:

- Separate policy for acting and for communicating
- View communication as an action (with an associated cost)

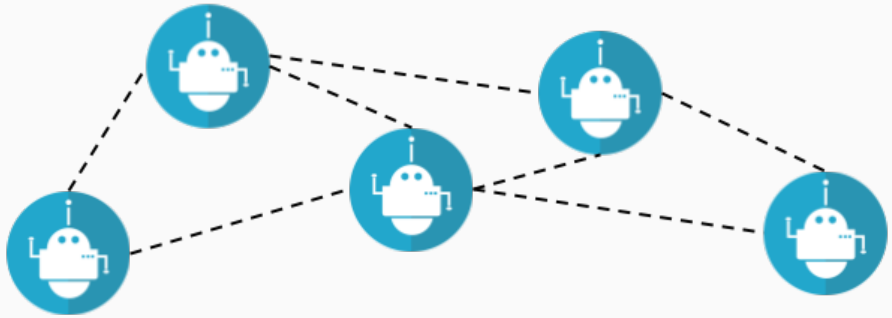
Value of Information

- We can think of POMDPs as requiring a search in belief-state space.
  - An action changes the belief state, not just the physical state.
  - Hence, the action is evaluated at least in part according to the information the agent acquires as a result.
  - A decision-theoretic agent needs to take into account the value of information and will execute information-gathering actions where appropriate.
  - One of the most important parts of decision making is knowing what questions to ask.
- **Value of information** (Lesser 99) computed as:
  Expected utility given information minus Expected utility not given information

How does this translate to multi-agent settings?

Communication

Slide by Amanda Prorok



How to effectively communicate to coordinate ?

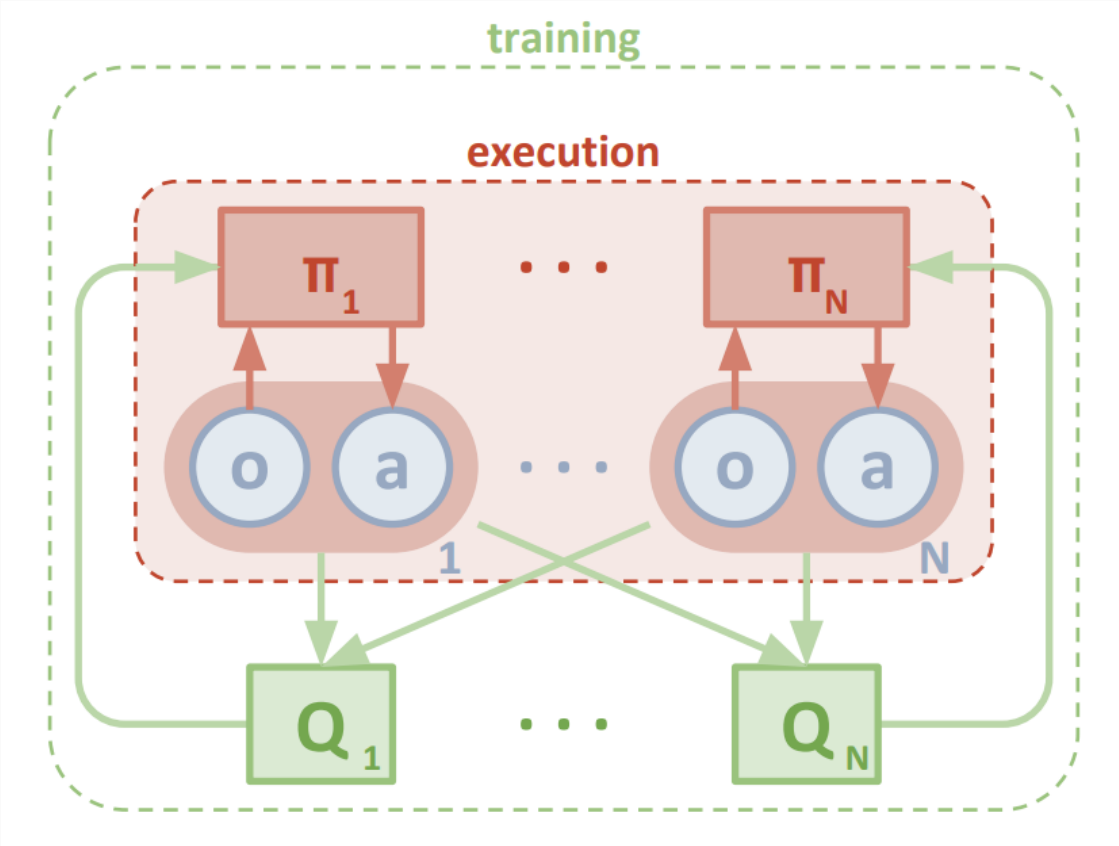
first-principles-based — **data-driven approaches**

Hardness of coordinate decision-making	**Decentralization required communication: who, what, when ?**	**Need for robustness due to noise, delays, faults, …**
Yu & LaValle 2013	Foerster et al. 2016	Parker 1998
Amato et al. 2015	Paulos et al. 2019	Gil et al. 2017
Halstea et al. 2012	Gama et al. 2021	Saulnier et al. 2017
		Mayya et al. 2021

How to effectively communicate to coordinate ?

Multi-Agent Actor-Critic for Mixed Cooperative-Competitive Environments - Lowe et al. 2017

An adaptation of actor-critic methods to multi-agent settings



Multi-Agent Actor-Critic for Mixed Cooperative-Competitive Environments - Lowe et al. 2017

- Assumptions on the setting:
  - the learned policies can only use local information (i.e. their own observations) at execution time,
  - does not assume a differentiable model of the environment dynamics
  - does not assume any particular structure on the communication method between agents (that is, there is no assumption on a differentiable communication channel).

  This provides a general-purpose multi-agent learning algorithm that could be applied not just to cooperative games with explicit communication channels, but competitive games and games with only physical interactions between agents.

Multi-Agent Actor-Critic for Mixed Cooperative-Competitive Environments - Lowe et al. 2017

- Goal accomplished by adopting the framework of **Centralized Training with Decentralized Execution (CTDE)**.
- Policies use extra information to ease training, which is not used at test time.
- Extension of actor-critic policy gradient methods where the critic is augmented with extra information about the policies of other agents.
- With N agents with policies parameterized by $\theta = \{\theta_1, ..., \theta_N\}$, and $\pi = \{\pi_1, ..., \pi_N\}$ as the set of all agent policies.
  - A **decentralized** gradient update by which policies are updated according to the independent value over the joint action
  - A **centralized action-value function** that takes as input the actions of all agents $a_1, ..., a_N$, in addition to some state information x (e.g.. agent observations), and outputs the Q-value for agent i.

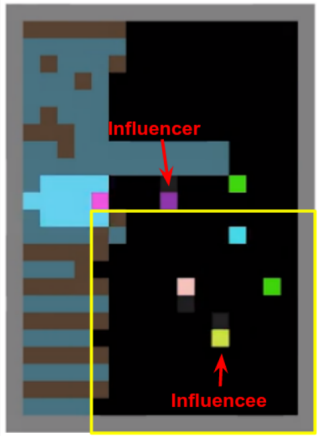
Multi-Agent Actor-Critic for Mixed Cooperative-Competitive Environments - Lowe et al. 2017

- Since each Q_{π_i} is learned separately, agents can have arbitrary reward structures, including conflicting rewards in a competitive setting.
- A primary motivation behind MADDPG is that, if we know the actions taken by all agents, the environment is stationary even as the policies change. This is not the case if we do not explicitly condition on the actions of other agents, as done for most traditional RL methods.

Social Influence as Intrinsic Motivation for Multi-Agent Deep Reinforcement Learning Jaques et al. 2018

- A MARL setting where multiple agents are trained to independently maximize their own individual reward; agents do not share weights.
- Each agent then receives its own reward which may depend on the actions of other agents.
- Each agent can only view a portion of the true state.
- **Communication:** similar to the approach of Foerster et al. (2016).:
  - at each timestep, each agent k chooses a discrete communication symbol m_t^k;
  - all symbols are concatenated into a combined message vector $m_t = [m_t^0, m_t^1, \ldots, m_t^N,]$ for N agents, which is given as input to every other agent in the next timestep.

Social Influence as Intrinsic Motivation for Multi-Agent Deep Reinforcement Learning Jaques et al. 2018



A moment of high influence when the purple influencer signals the presence of an apple (green tiles) outside the yellow influencee's field-of-view (yellow outlined box).

Social Influence as Intrinsic Motivation for Multi-Agent Deep Reinforcement Learning Jaques et al. 2018

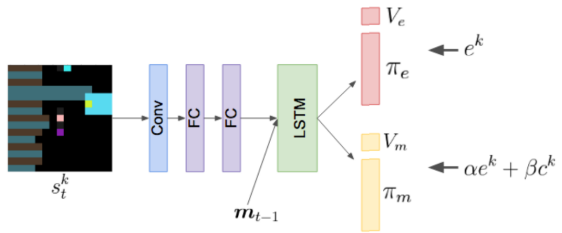
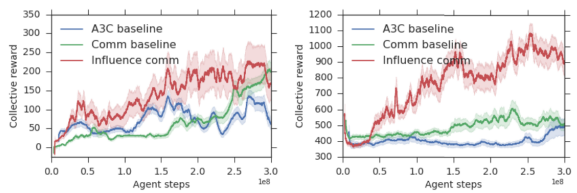


Figure 3: The communication model has two heads, which learn the environment policy, π_e, and a policy for emitting communication symbols, π_m. Other agents' communication messages $\boldsymbol{m}_{t-1}$ are input to the LSTM.



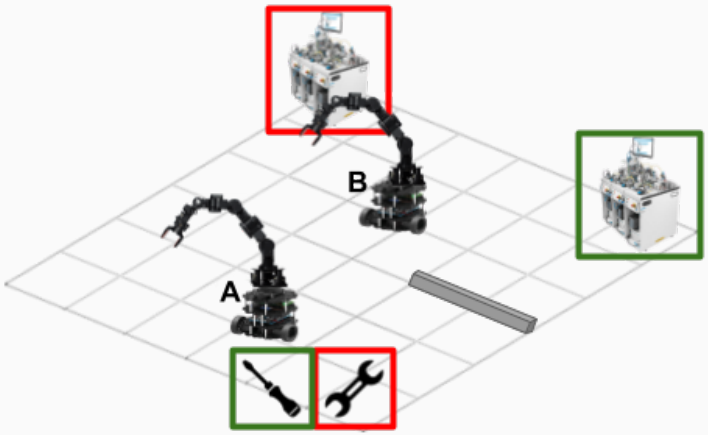
(a) *Cleanup* (b) *Harvest*

Figure 4: Total collective reward for deep RL agents with communication channels. Once again, the influence reward is essential to improve or achieve any learning.

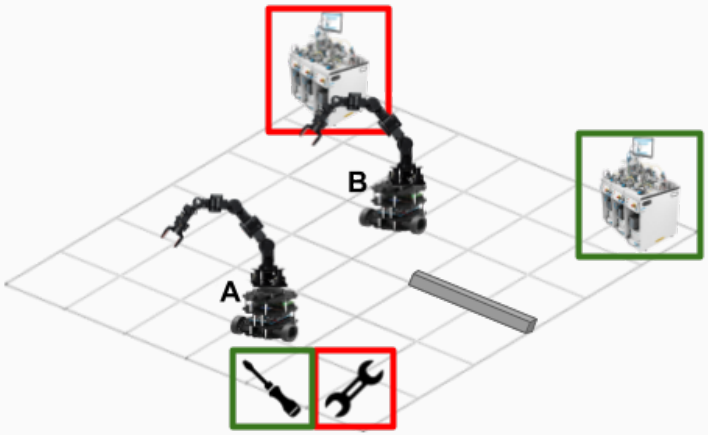
The limited success of emergent communication has led to a recent increase in semi-grounded and grounded communication settings.

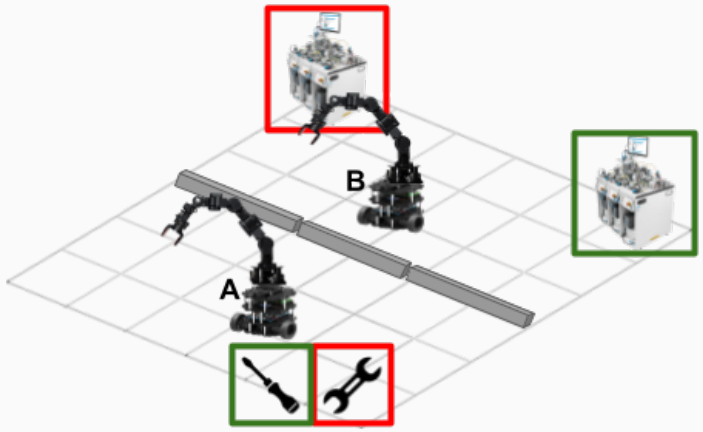
Promoting Resilience in Multi-Agent Reinforcement Learning via Confusion-Based Communication - Keren et al.

Promoting Resilience in Multi-Agent Reinforcement Learning via Confusion-Based Communication - Keren et al.



Promoting Resilience in Multi-Agent Reinforcement Learning via Confusion-Based Communication - Keren et al.





Promoting Resilience in Multi-Agent Reinforcement Learning via Confusion-Based Communication - Keren et al.

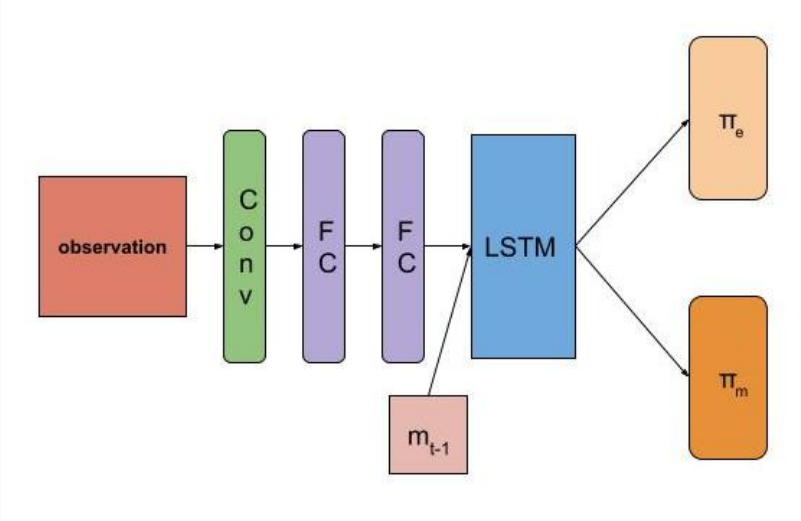
Definition (Misalignment)

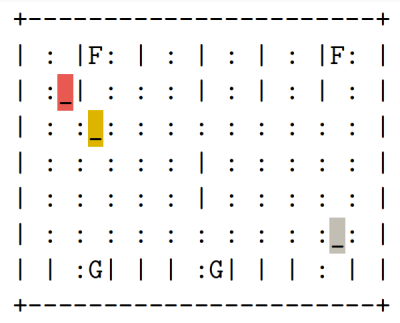
Let r_t and $\hat{r_t}$ be the observed and estimated reward of agent p after taking action a_t in s_t. The misalignment for agent p at s_t after taking action a_t, denoted J_{s_t,a_t}, is defined as:

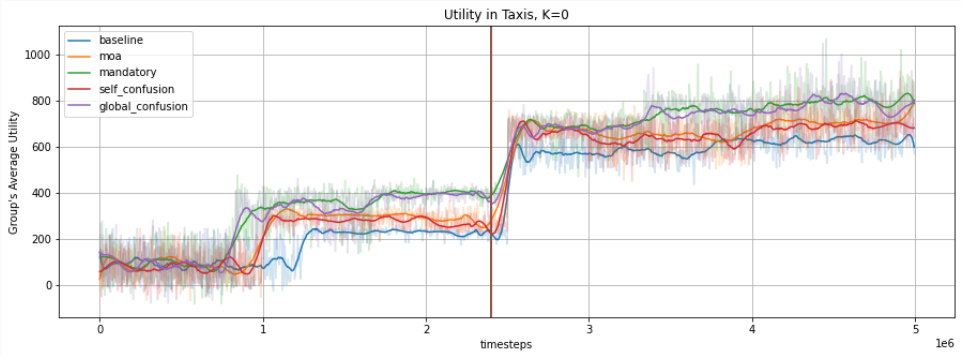
$$J^p_{s_t,a_t} = \frac{|r_t - \hat{r_t}|}{r_t}$$

Misalignment corresponds to the agents' familiarity with the environment, which may increase due to perturbations. By communicating misaligned transitions, agents increase familiarity of the environment for the other agents. We support mandatory and emergent communication.

Promoting Resilience in Multi-Agent Reinforcement Learning via Confusion-Based Communication - Keren et al.





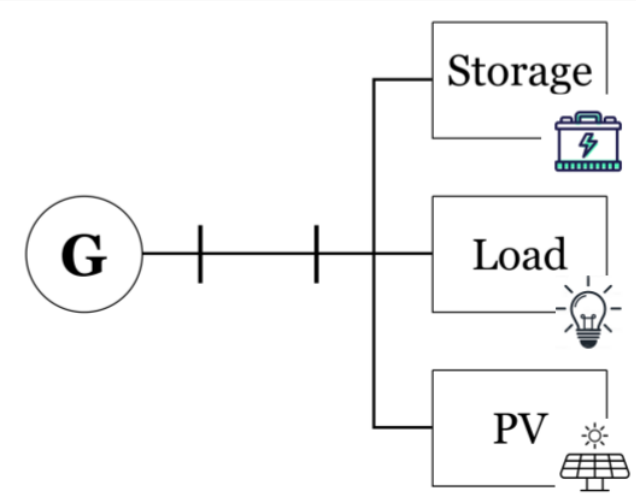


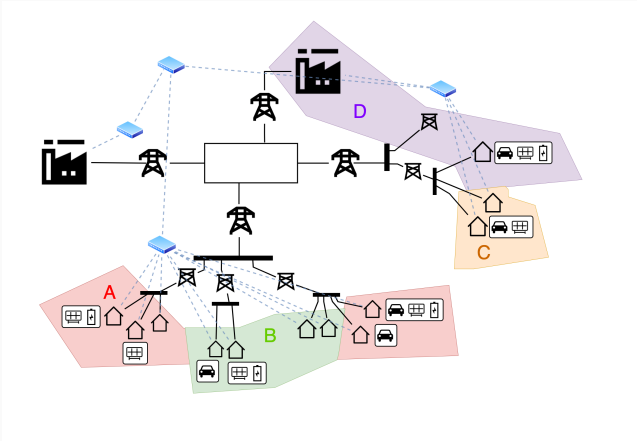
So much more we didn't cover

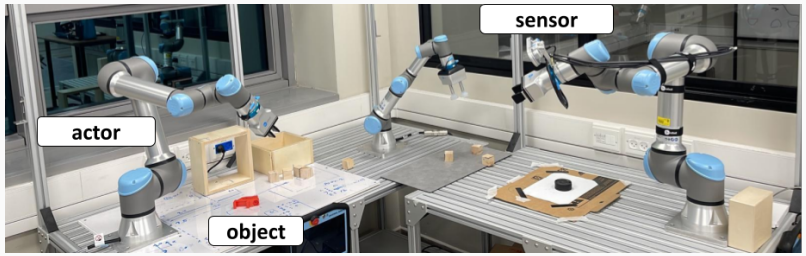
Additional References

- The Autonomous Agents Research Group led by Stefano V. Albrecht at the University of Edinburgh. *https://agents.inf.ed.ac.uk/*
- Jack Foerster's group https://eng.ox.ac.uk/people/jakob-foerster/
- Strong groups at BIU and BGU
- Many many more...

MARL Examples from the CLAIR Lab



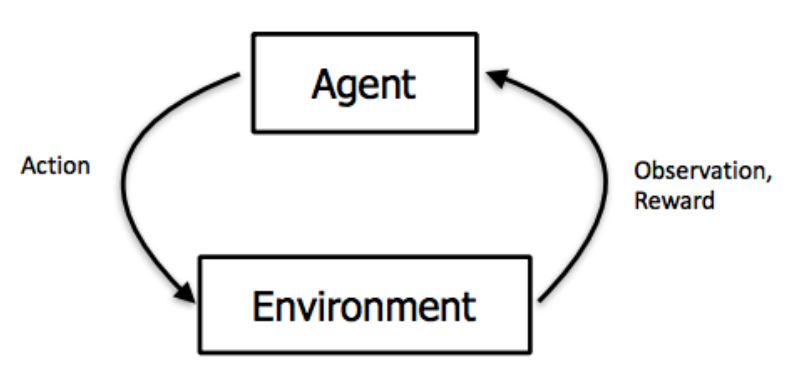




Conclusion

What is this class about ?

Sequential Decision Making: deciding how to act now in the world in order to accomplish a long-term objective



What is this class about ?

- From observations to decisions
- From low-level to high-level reasoning
- From theory to practice to theory ...
- **From single agent to multi-Agent**

Our Focus

Spectrum of single agent and multi-agent AI research questions.

- Non-adversarial settings
- Complex tasks
- Combinations of approaches (and principled ways to choose the best approach)

Approaches to Decision Making

We are seeking a **policy**:
mapping from state space $\mathcal{S}$ to actions $\mathcal{A}$.

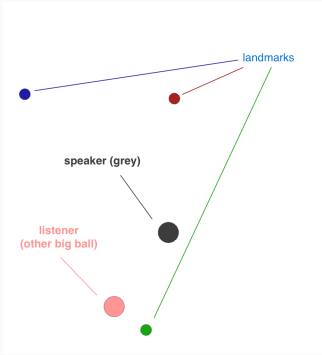
Can be deterministic $\pi : \mathcal{S} \mapsto \mathcal{A}$ or probabilistic $\pi : \mathcal{S} \times \mathcal{A} \mapsto [0, 1]$

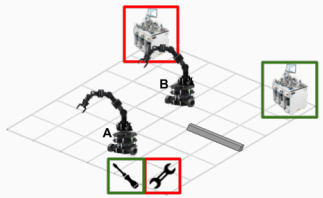
More accurately : we seek a mapping from observations Ω to actions.

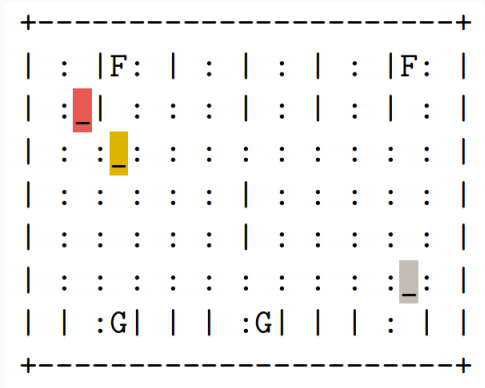
Basic approaches:

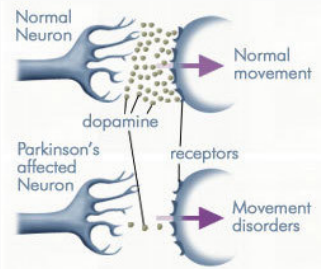
- Programming/reactive
- Model-Based Planning (using a domain-independent descriptive language)
- Machine Learning (from data and experience)
- Reinforcement Learning

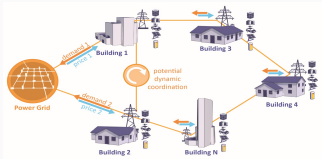
What is a good (enough) model?





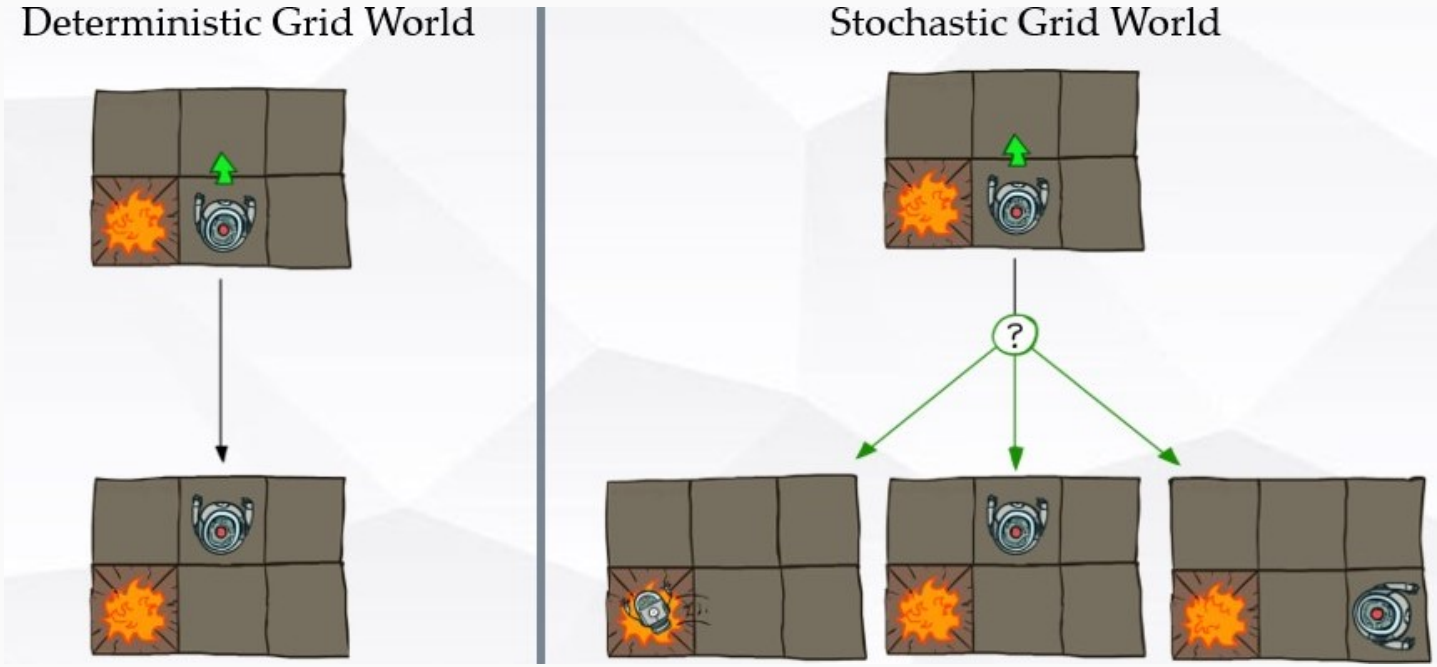






What do we need to model ?

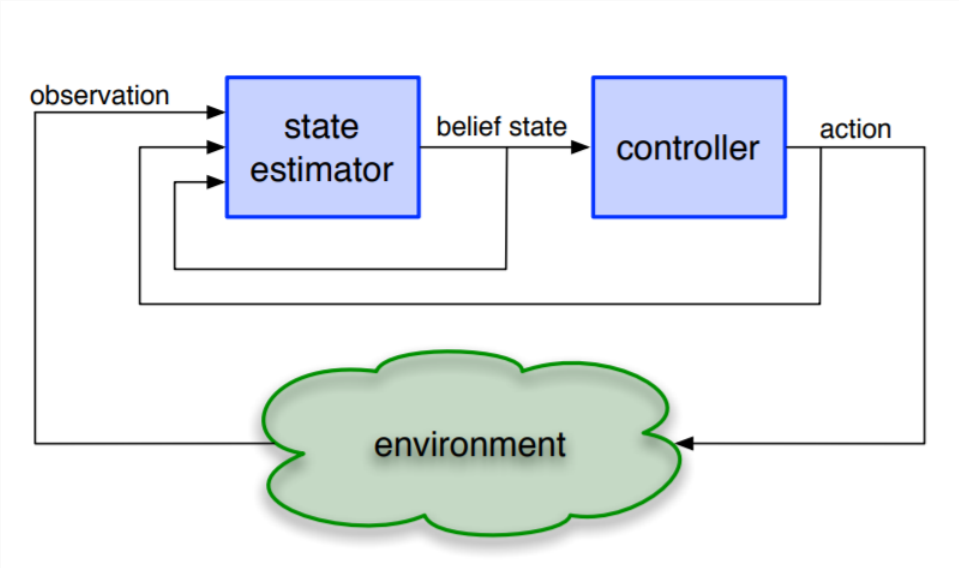
From Deterministic to Stochastic Domains



Partial Observability

Beliefs and Belief Tracking

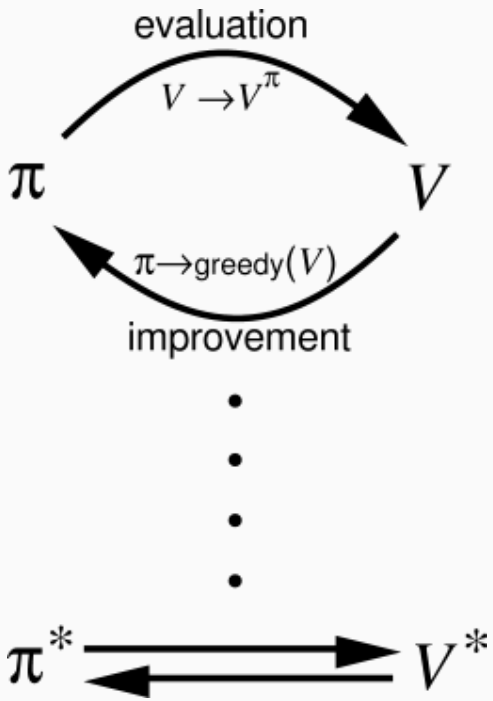
- A **belief** is a representation of the possible world states.
- In partially observable domains, we may have a **sensor model** represented as a mapping function from what is observed to the actual world state.
- The agent maintains its belief via a *state estimator* - which we will refer to as the process of **Belief Tracking**.



From Kaelbling, L. P., and T. Lozano-Perez. "Integrated Task and Motion Planning in Belief Space" 2013 *https://dspace.mit.edu/bitstream/handle/1721.1/87038/Kaelbling_Integrated%20task_pdf?sequence=1&isAllowed=y*

Policy Evaluation and Policy Update

- Prediction / Evaluation: evaluate the future given a policy
- Control/ Update / Improvement: optimize the policy.



Solution Approaches

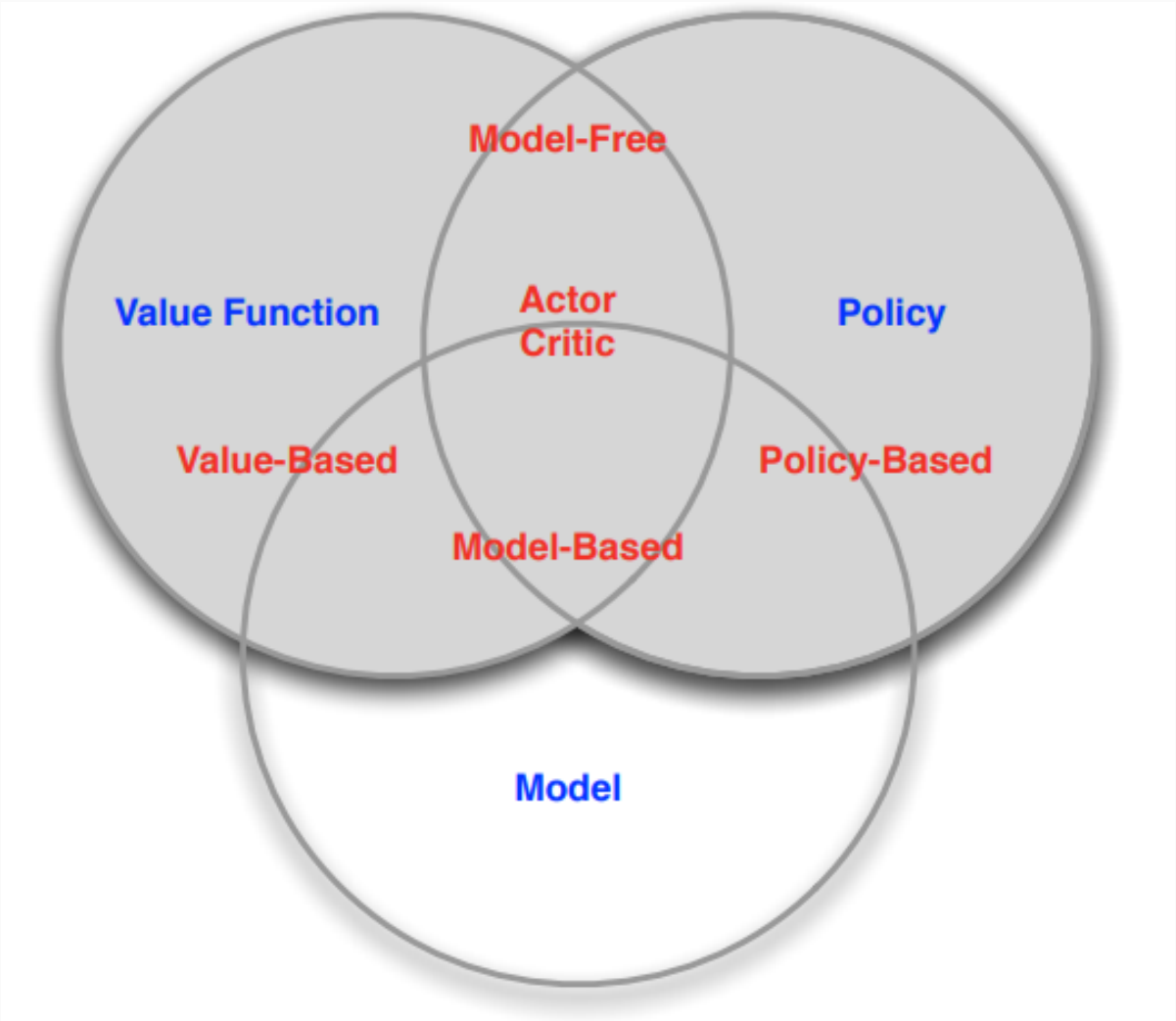
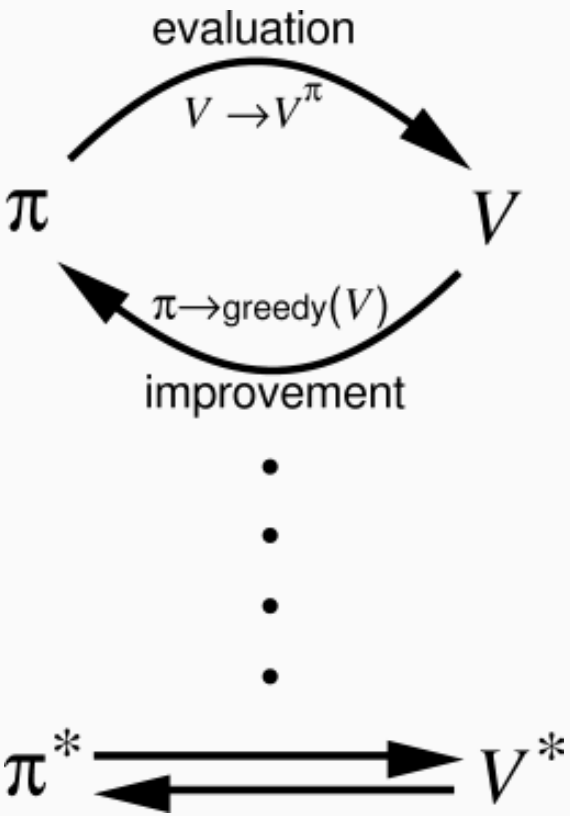
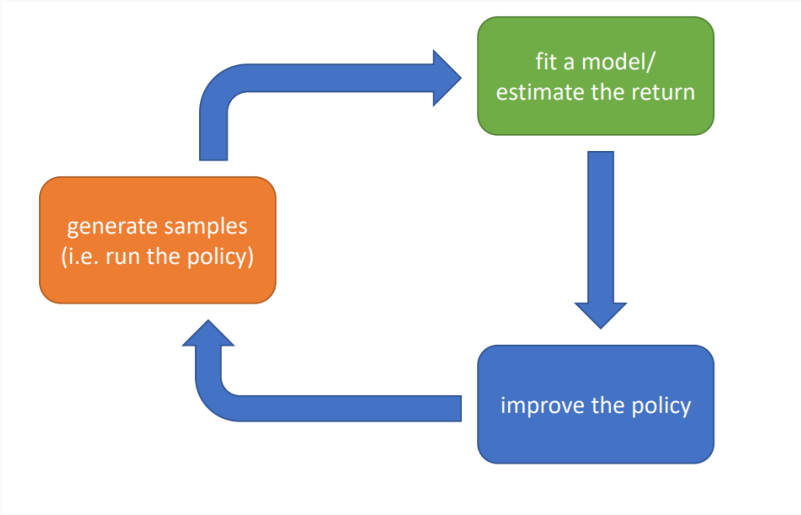


image by David Silver

Control Approaches Skeleton





Left image by Sutton and Barto. Right image By Sergey Levine

Approaches to Control

- Supervised learning
- Model Based Planning
- Monte-Carlo methods
- Temporal-Difference methods
- Combined approaches

AO* (Nilsson 1980)

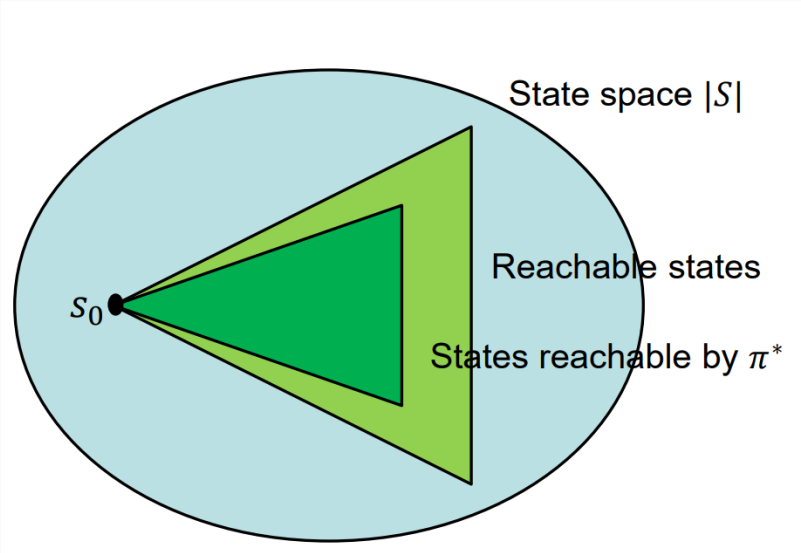


Image by Pascal Poupart 2013.

Update Formulas

$$v_\pi(s) \doteq \mathbb{E}_\pi \left[G_t \mid S_t = s \right]$$

$$= \mathbb{E}_\pi \left[\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \mid S_t = s \right]$$

$$= \mathbb{E}_\pi \left[r_{t+1} + \gamma G_{t+1} \mid S_t = s \right]$$

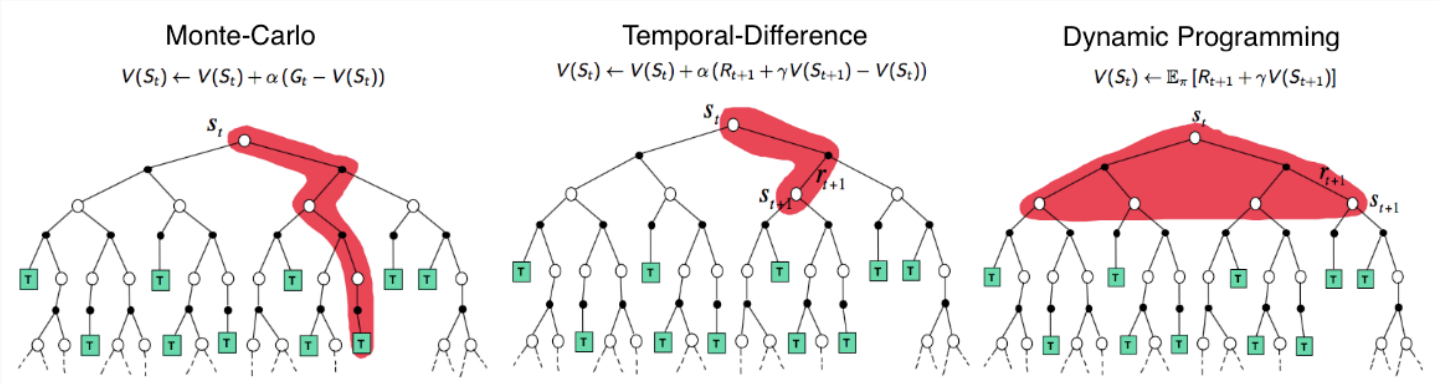
$$= \mathbb{E}_\pi \left[r_{t+1} + \gamma \sum_{k=0}^{\infty} \gamma^k r_{t+k+2} \mid S_t = s \right]$$

$$= \sum_a \pi(a|s) \sum_{s'} \sum_r p(s', r|s, a) \left[r + \gamma \mathbb{E}_\pi \left[\sum_{k=0}^{\infty} \gamma^k R_{t+k+2} \mid S_{t+1} = s' \right] \right]$$

$$= \sum_a \pi(a|s) \sum_{s'} \sum_r p(s', r|s, a) \left[r + \gamma \mathbb{E}_\pi \left[G_{t+1} \mid S_{t+1} = s' \right] \right]$$

$$= \sum_a \pi(a|s) \sum_{s'} \sum_r p(s', r|s, a) \left[r + \gamma v_\pi(s') \right], \forall s \in \mathcal{S}$$

Planning vs. Monte Carlo



Which approach is best ?

Refresher: Model-Free vs. Model-Based RL

Model-based approach to RL

- learn the MDP model, or an approximation of it
- use it for policy evaluation or to find an optimal policy

Model-free approach to RL

- derive an optimal policy without explicitly learning the model.
- useful when model is difficult to represent and/or learn

We will consider both types of approaches

Refresher: On-Policy vs. Off-Policy

On-Policy Learning

- "Learn on the job"
- Learn about policy π from experience sampled from π

Off-Policy Learning

- "Look over someone's shoulder"
- Learn about policy π from experience sampled from μ

Greedy in the Limit with Infinite Exploration (GLIE)

- All state-action pairs are explored infinitely many times,

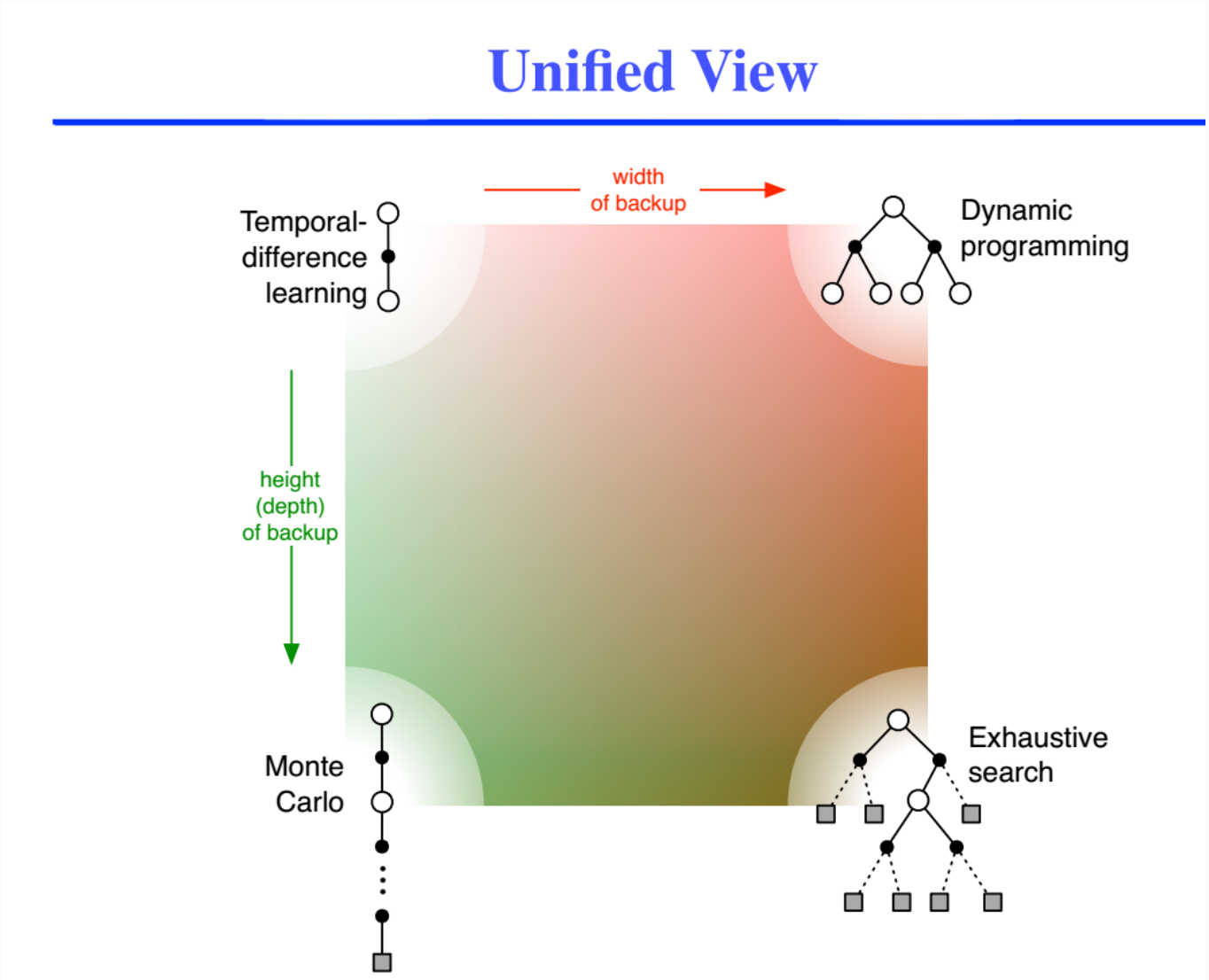
$$\lim_{k \to \infty} N_k(s, a) = \infty$$

- The policy converges on a greedy policy,

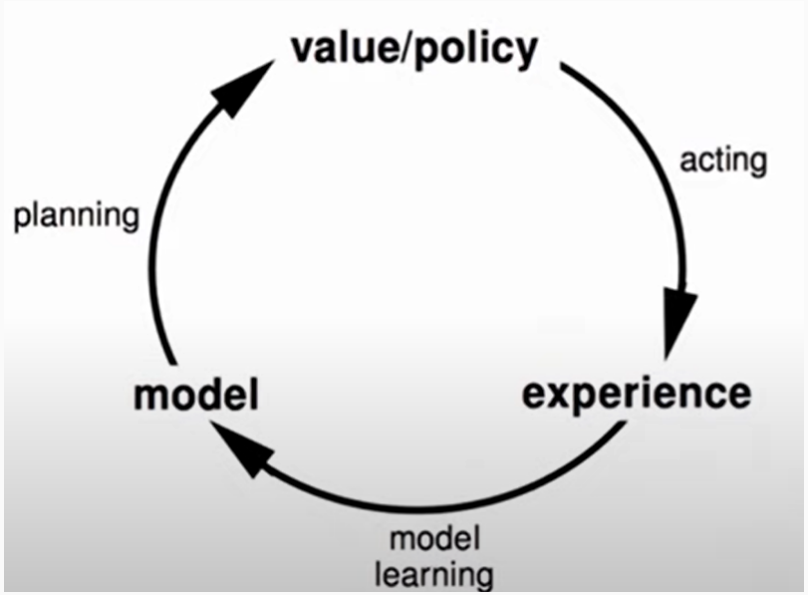
$$\lim_{k \to \infty} \pi_k(a|s) = \mathbf{1}(a = \arg\max_{a' \in \mathcal{A}} Q_k(s, a'))$$

For example, ϵ-greedy is GLIE if α reduces to zero at $\epsilon_k = \frac{1}{k}$.

Unified View of Reinforcement Learning



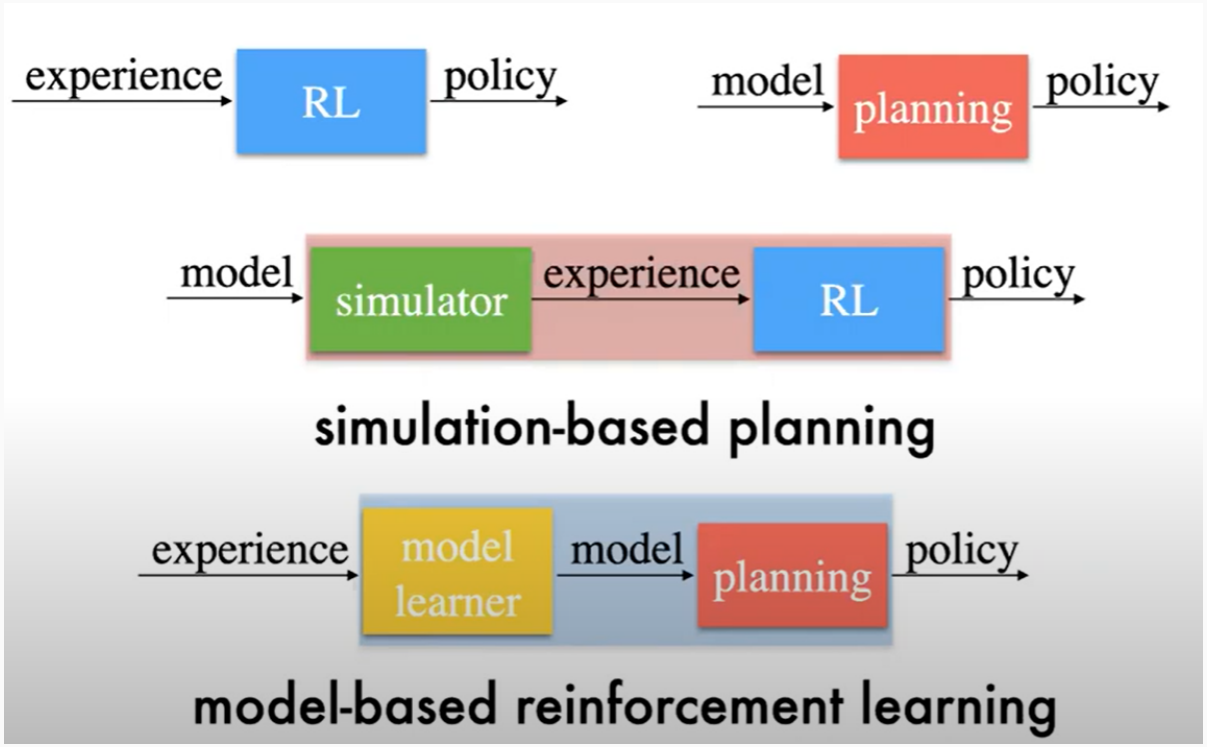
Model-Based RL



By Emma Brunskill

https://www.youtube.com/watch?v=vDF1BYWhqL8

Models in RL

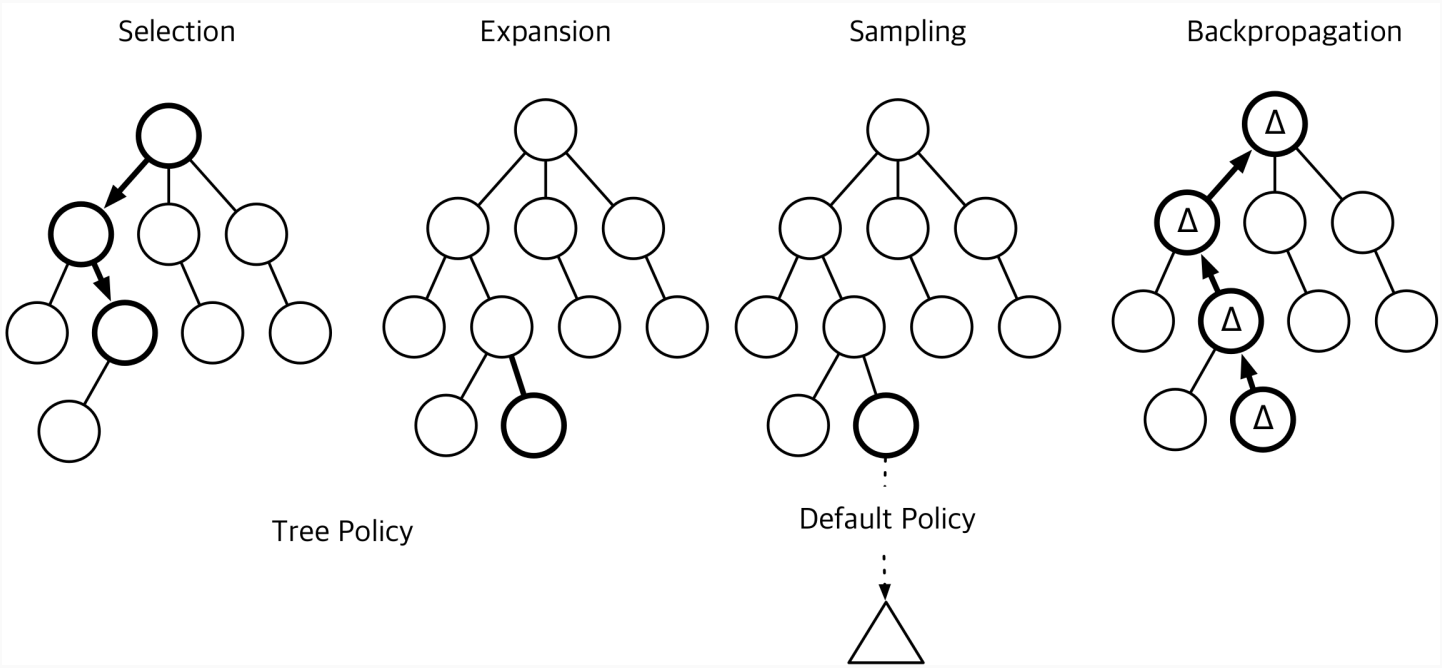


Talk by Michael Littman (the funniest AI researcher in the world!)

https://youtu.be/45FKxa3qPHo?t=265

Monte Carlo Tree Search

function MONTE-CARLO-TREE-SEARCH(*state*) **returns** *an action*
  tree ← NODE(*state*)
  while IS-TIME-REMAINING() **do**
    leaf ← SELECT(*tree*)
    child ← EXPAND(*leaf*)
    result ← SIMULATE(*child*)
    BACK-PROPAGATE(*result*, *child*)
  return the move in ACTIONS(*state*) whose node has highest number of playouts



Types of Value Function Approximation

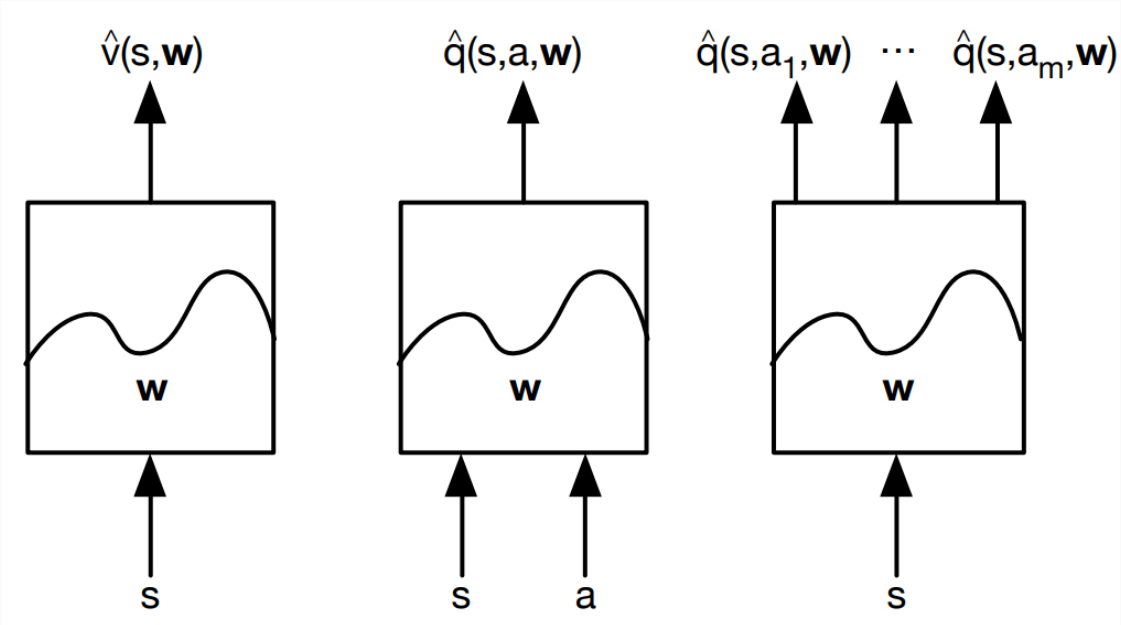


Image by David Silver

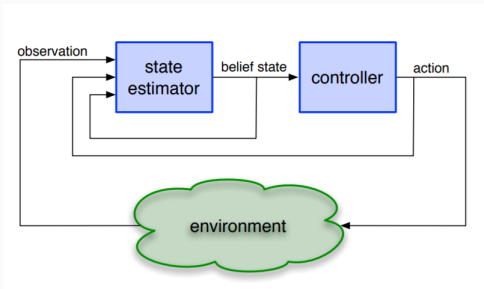
Complex Tasks

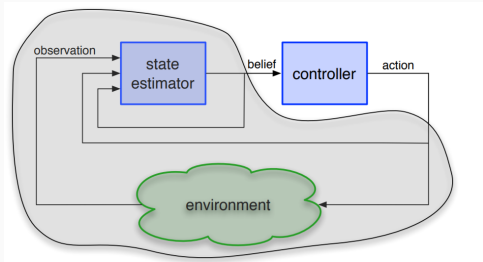
”Doing for our robots what nature did for us” - Leslie Kaelbling (MIT) *https://youtu.be/5R-xL9YmdR0*

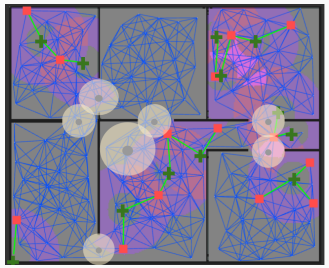
Planning in Belief Space: Solution Approaches

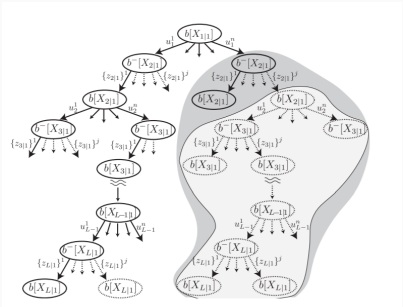
Combinations of different approaches:

- Planning in an MDP with beliefs as states
- Sampling / discretization
- Approximations / relaxations





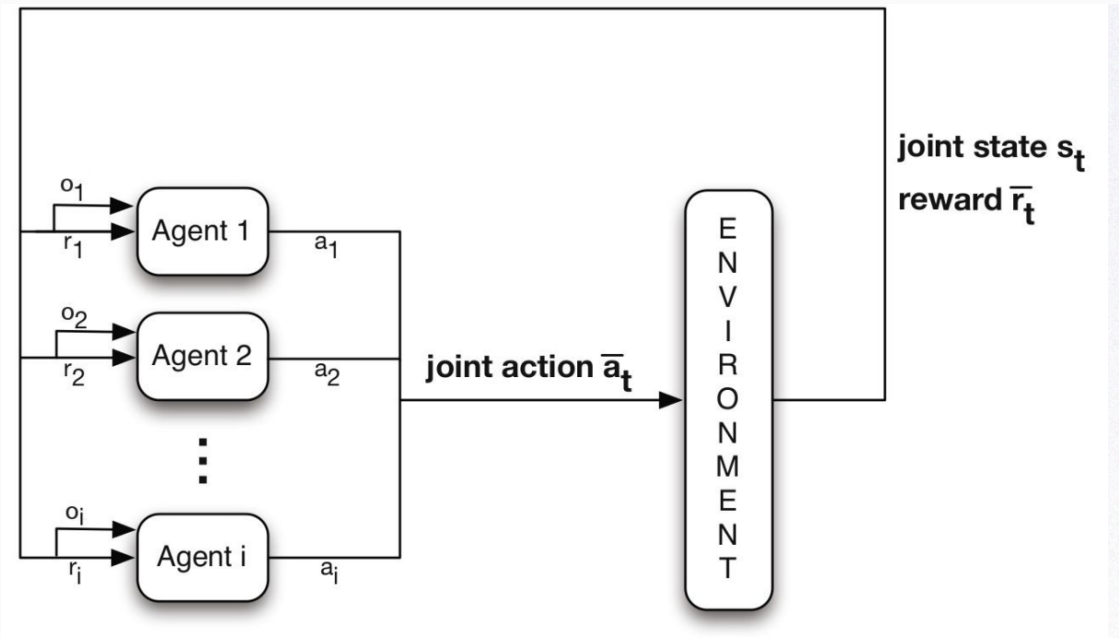




See work by Vadim Indelman from the Technion, e.g.,

https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8793548

The Multi-agent-Environment Interface



Source: Nowe, Vrancx & De Hauwere 2012

To model a multi-agent setting, We will use a *Markov game*, or *stochastic game* which is a generalization of the MDP to multi-agent settings (Littman'94).

Dimensions of Multi-Agent Systems

- **Control:** from centralized to decentralized.
- **Decision-making approach:** planning, learning, RL
- **Communication:** from no communication to limitless communication
- **Observability:** no sensing (conformant) to full information
- **Shared resources/interaction:** space, energy, etc.
- **Objectives/utilities:** from collaborative to adversarial

MARL Dynamics

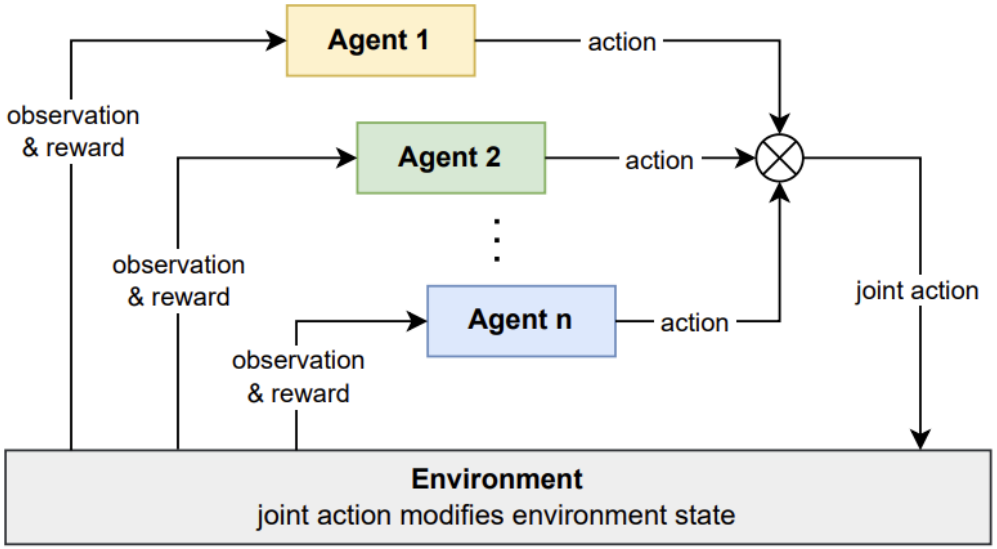


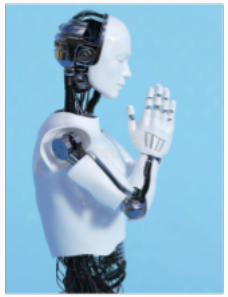
Figure 1.3: Schematic of multi-agent reinforcement learning. A set of n agents receive individual observations about the state of the environment, and choose actions to modify the state of the environment. After taking an action, each agent receives a scalar reward and a new observation, and the loop repeats.

Three Invited Talks

- Ophir Gershon - Hierarchical planning and RL in practice
- Or Rivlin - Model-free vs. model-based RL in the wild
- Nitay Alon - Theory of Mind in Multi-agent RL

What I Hope You Take From the Course

The Big Picture



Human, grant me the serenity to accept the things I cannot learn; data to learn the things I can; and wisdom to know the difference.

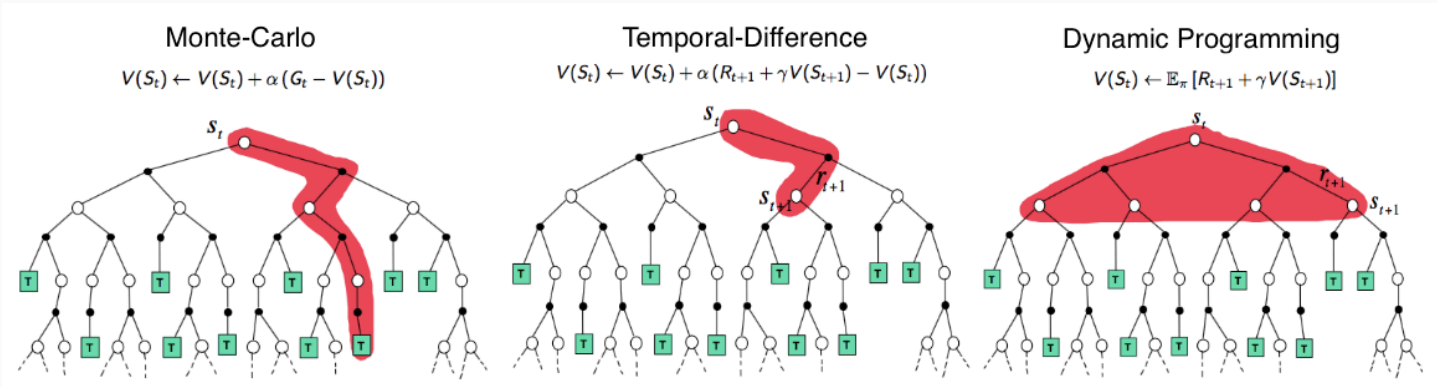
\+ *and to figure out how to best combine the data & models that are available*

God, grant me the serenity to accept the things I cannot change, courage to change the things I can & wisdom to know the difference

Subbarao Kambhampati

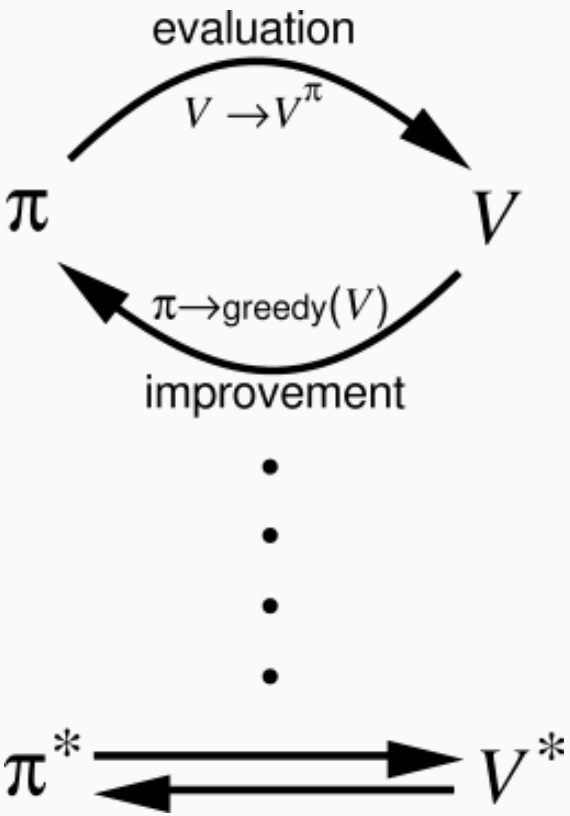
See David Cox's talk if you need convincing:
https://vimeo.com/389562304

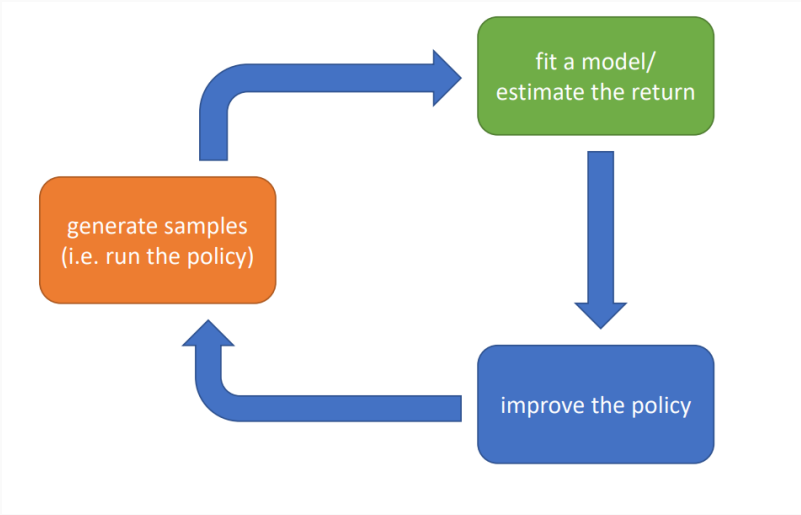
Planning vs. Monte Carlo ? Model-free Model-Based ?



Which approach is best ? **The one that is most suitable to the problem, and not just the one you are used to.**

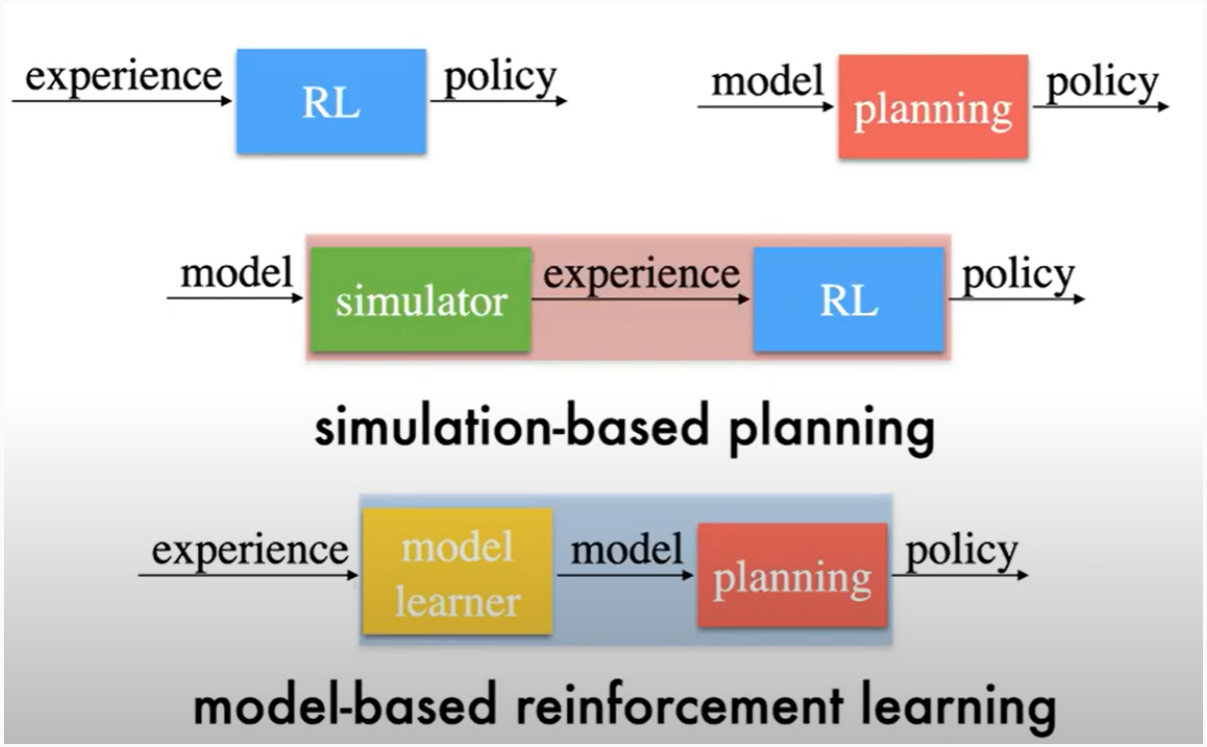
Control Approaches Skeleton





Left image by Sutton and Barto. Right image By Sergey Levine

Models in RL

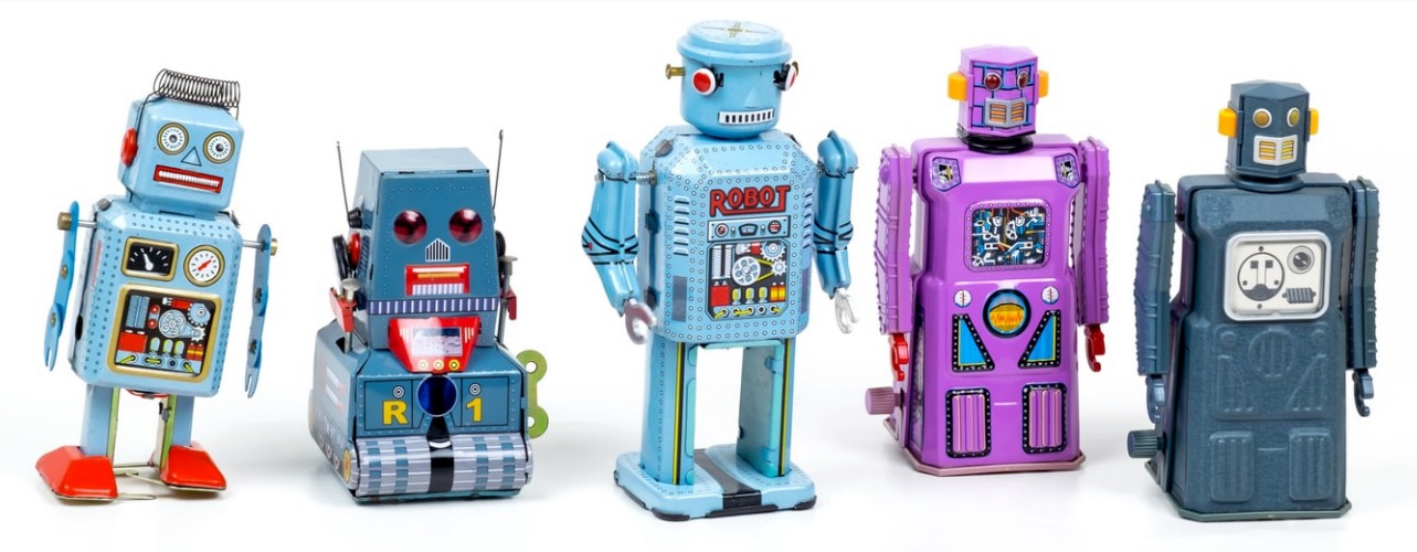


Talk by Michael Littman (the funniest AI researcher in the world!)

https://youtu.be/45FKxa3qPHo?t=265

Dimensions of Multi-Agent Systems

- **Control:** from centralized to decentralized.
- **Decision-making approach:** planning, learning, RL
- **Communication:** from no communication to limitless communication
- **Observability:** no sensing (conformant) to full information
- **Shared resources/interaction:** space, energy, etc.
- **Objectives/utilities:** from collaborative to adversarial



What Next ?

Other courses at the Technion

- Many advanced AI courses
- Many robotics related courses *link*
- AI project at the CLAIR lab