MULTIAGENT PLANNING

Video segments: coordinating activities of multiple robots

- Reassembly an object with two cooperating manipulators, from a group at University of Zagreb, Croatia
  - [http://www.youtube.com/watch?v=DzDzwG2qFKc](http://www.youtube.com/watch?v=DzDzwG2qFKc)

- Avoiding collisions between aerial robots, from the Aerospace Controls Lab at MIT
  - [https://www.youtube.com/watch?v=Zkxc4PRGvC4](https://www.youtube.com/watch?v=Zkxc4PRGvC4)

- How can agents build distributed plans and coordinate their activities?
Planning

- Necessary when near-term choices of actions can enable, or prevent, later action choices required to achieve goals
- Possible when agent possesses a sufficiently detailed and correct model of the environment, and of how actions affect the environment
- Challenging because the space of possible plans grows exponentially with the plan duration
Multiagent planning

- Now the near-term choices of actions can enable, or prevent, later action choices of others required to achieve goals, and others’ near-term actions can affect the agent’s later choices too.

- Possible when agents can explicitly or implicitly model others’ plans, and predict outcomes in the environment of executing the plans jointly.

- Challenging because the space of possible individual plans grows exponentially with the plan duration, and of multiagent plans grows exponentially in the number of agents.
What aspects are multiagent? (1)

- Multiagent planning could refer to just the *product* of the planning process
  - A centralized process builds a plan representation that specifies how each of multiple agents should behave

- Multiagent planning could refer to the *process* of formulating plan decisions
  - Multiple agents participate in the construction of a single plan or policy
**What aspects are multiagent? (2)**

- *Both* the product and the process are multiagent
  - Each agent applies its local expertise and awareness to construct its local plan
  - Agents use communication, and/or shared knowledge and biases, to shape their local plans to conform better to others’ plans, in order to more effectively achieve collective objectives
Flavors of multiagent planning

- Coordination prior to local planning
  - Committing to how to work together, and then making suitable local planning decisions
- Local planning prior to coordination
  - Formulating local plan decisions separately, then adjusting them for coordination
- Decision-theoretic multiagent planning
  - Multiagent planning in the face of non-determinism and partial observability
- Dynamic multiagent planning
  - Monitoring and replanning during execution
Decision-theoretic multiagent planning

- A group of agents interact in a stochastic environment
- Each “episode” involves a sequence of decisions over some finite or infinite horizon
- The change in the environment is determined stochastically by the current state and the set of actions taken by the agents
- Each decision maker obtains different partial observations of the overall situation
Markov Decision Process (MDP)

- Expressive model for stochastic planning
- Originated in operations research in the 1950s
- Adopted by the AI community as a framework for planning and learning under uncertainty
- Can be solved efficiently by DP algorithms and a range of search and abstraction methods
- Everything is an MDP – just keep adding states!
Multiagent MDPs

- Full states vs. local states
- Joint actions vs. independent actions
- Team rewards vs. local rewards

... and all the combinations and variants of the above, including ...
Cooperative repeated games or stochastic games

- $n$ agents
- Individual actions $A_i$
- Agents play repeated games
  - Each agent selects one of its actions $\rightarrow$ joint action
  - A joint action is associated to a (probability distribution over) reward
  - The same reward is given to all agents
- Each game is a state
- A transition function returns the probability of moving from state $s$ to state $s'$ given actions of the agents
Solution representation

- Each agent’s behavior is described by a local policy (also called strategy) $\delta_i$
- Policy can be represented as a mapping from local memory states to actions
- Actions can be selected deterministically or stochastically
- Goal is to maximize expected reward over a finite horizon or discounted infinite horizon
Joint action learning
[presented by Dominic Crippa]

Gradient ascent
[presented by Alessandro Artoni]

...
Partially Observable MDP

- Generalization formulated in the 1960s
- The agent receives noisy observations of the underlying world state
- Need to remember previous observations in order to act optimally
- More difficult, but there are DP algorithms
Decentralized POMDP

- Generalization of POMDP involving multiple cooperating decision makers, each receiving a different partial observation after a joint action is taken.
Example: Mobile robot planning

States: grid cell pairs

Actions: \(\uparrow, \downarrow, \leftarrow, \rightarrow\)

Transitions: noisy

Goal: meet quickly

Observations: red lines
Example: Cooperative box-pushing

**Goal:** push as many boxes as possible to goal area; larger box has higher reward, but requires two agents to be moved
**DEC-POMDPs**

**Definition** A decentralized partially observable Markov decision process (DEC-POMDP) is a tuple \( \langle I, S, \{A_i\}, P, \{\Omega_i\}, O, R, T \rangle \) where

- \( I \) is a finite set of agents indexed \( 1, \ldots, n \).
- \( S \) is a finite set of states, with distinguished initial state \( s_0 \) or belief state \( b_0 \).
- \( A_i \) is a finite set of actions available to agent \( i \) and \( \tilde{A} = \bigotimes_{i \in I} A_i \) is the set of joint actions, where \( \tilde{a} = \langle a_1, \ldots, a_n \rangle \) denotes a joint action.
- \( P : S \times \tilde{A} \rightarrow \Delta S \) is a Markovian transition function. \( P(s'|s, \tilde{a}) \) denotes the probability of a transition to state \( s' \) after taking joint action \( \tilde{a} \) in state \( s \).
- \( \Omega_i \) is a finite set of observations available to agent \( i \) and \( \tilde{\Omega} = \bigotimes_{i \in I} \Omega_i \) is the set of joint observation, where \( \tilde{\omega} = \langle o_1, \ldots, o_n \rangle \) denotes a joint observation.
- \( O : \tilde{A} \times S \rightarrow \Delta \tilde{\Omega} \) is an observation function. \( O(\tilde{\omega}|\tilde{a}, s') \) denotes the probability of observing joint observation \( \tilde{\omega} \) given that joint action \( \tilde{a} \) was taken and led to state \( s' \). Here \( s' \in S, \tilde{a} \in \tilde{A}, \tilde{\omega} \in \tilde{\Omega} \).
- \( R : \tilde{A} \times S \rightarrow \mathbb{R} \) is a reward function. \( R(\tilde{a}, s') \) denotes the reward obtained after joint action \( \tilde{a} \) was taken and a state transition to \( s' \) occurred.
Subclasses and related models

- **Decentralized MDP** (DEC-MDP): DEC-POMDP in which the combined observations of all the agents provide perfect information about the underlying world state.

- **Multiagent MDP** (MMDP): DEC-MDP in which each agent has perfect information about the underlying state.

- **Partially-Observeable Stochastic Game** (POSG): Generalization of DEC-POMDP in which each agent can have a different objective function.

- **Interactive POMDP** (I-POMDP): A model in which each agent explicitly represents beliefs about the other agents and about the world state.
Each node is labeled with an action and each edge with an observation that could be received.
Each controller state is labeled with an action and edges between states are labeled with observations.

Green arrow designates the initial state of the controller.
In each controller state, actions are selected stochastically; when an observation is obtained, the transition to a new state is also stochastic.
Evaluating solutions

- For a finite-horizon problem with initial state $s_0$ and $T$ time steps, the value of a joint policy $\delta$ is

\[
V^\delta(s_0) = E \left[ \sum_{t=0}^{T-1} R(\tilde{a}_t, s_t) | s_0, \delta \right].
\]

- For an infinite-horizon problem, with initial state $s_0$ and discount factor $\gamma$ in $[0;1)$, the value of a joint policy $\delta$ is

\[
V^\delta(s_0) = E \left[ \sum_{t=0}^{\infty} \gamma^t R(\tilde{a}_t, s_t) | s_0, \delta \right].
\]
Algorithms

- Exact dynamic programming
  \textit{[presented by Angelo Carlino]}
- Policy iteration for infinite-horizon DEC-POMDPs
- ...

...
Coordination prior to local planning

- Formulate interaction plans/rules beforehand, and commit to following them

- Main ideas
  - Core aspects about what coordination decisions will need to be made and how they will be resolved are known ahead of time
  - Details of agents’ plans specific to a particular problem instance can fit into the predefined coordination framework
Approaches

- Social laws
- Organizational structuring
- ...
Local planning prior to coordination

- Appeals to locality and decomposability arguments
  - The collective endeavor is composed of largely independent activities done by individuals
  - Interdependencies are local to small numbers of individuals

- This argues for a divide-and-conquer approach
  - Each individual plans as if it were completely independent
  - Then interdependencies are identified and resolved
Approaches

- State-space techniques
- Plan-space techniques
  [presented by Davide Azzalini]
- ...
Other issues on multiagent planning

- Control and execution
- Other approaches
  - Teamwork
  - ...