Cooperative Epistemic Multi-Agent Planning with Implicit Coordination
Thorsten Engesser1, Thomas Bolander2, Robert Mattmüller3, Bernhard Nebel1

1University of Freiburg, 2Technical University of Denmark

Why Epistemic Planning?

- Implicit Coordination by projecting into the other agents’ state of knowledge
- Collaborative, decentralized setting: Agents individually plan and decide, when and how to act under incomplete/distributed knowledge.
- Idea: Make observational and communicative actions part of the plan!

Dynamic Epistemic Logic

- $\varphi := p | \neg\varphi | \varphi \land \varphi | K_i\varphi | C_i\varphi | (a)\varphi$
- Allows reasoning about knowledge and action consequences
- $K_i(a)\varphi$: Agent $i$ knows $a$ is applicable and leads to a situation where $\varphi$ holds
- Interpretation over standard SS, Kripke models

- $M = \langle W, R, K, L, a, e, s \rangle$

- $M, w_i \models p$ and $M, w_2 \not\models \neg p$
- $M, w_1 \models \neg K_i p \land K_1 \neg p$

Epistemic States and Actions

- **Epistemic States** $(M, W_i)$ correspond to the belief states $(\{M, w \mid w \in W_i\})$
- Global states $(M, \{w\})$ contain complete information about the situation
- Local states (with $W_i$ closed under $R_i$) match the perspective of an agent $i$
- Shifting perspective to the associated local state of another agent possible
- **Epistemic Actions** $(\mathcal{L}, E_\mathcal{L})$ as multi-pointed Kripke structures on possible events with additional per-event pre- and post-conditions
- Successor states via the **Product Update** $(M', W'_i) = (M, W_i) \otimes (\mathcal{L}, E_\mathcal{L})$

Example: Observational and Communicative Actions

- Sensing action $\text{sense}$ for agent 1, checking whether $p$ or $\neg p$:

  $\begin{array}{c|c|c|c}
  w_1 & w_2 & e_1 & e_2 \\
  \hline
  p & \neg p & (p, T) & \neg(p, T) \\
  \hline
  w_1 & w_2 \\
  \end{array}$

- From the global perspective $s_0$ (given $w_1$ is the actual world):

  $\begin{array}{c|c|c|c|c}
  w_1 & w_2 & e_1 & e_2 & e_3 \\
  \hline
  p & \neg p & (p, T) & \neg(p, T) & 2,3,1 \\
  \hline
  w_1 & w_2 \\
  \end{array}$

- Agent 1 will be able to identify the actual world at run-time.

  $s_0 \models (\langle \text{sense}\rangle K_p) \land \neg (\langle \text{sense}\rangle \neg p)$

- For agent 3, $\text{inform}_2p$ is indistinguishable to agent 1 informing agent 2 of $\neg p$.

  $s_0 \otimes \langle \text{sense}\rangle \models K_3(\text{inform}_2p) \land \neg \langle \text{inform}_2p\rangle K_p$

Solution Concepts

- **Centralized Plan** $a_1, a_2, \ldots, a_n \in A$, iff (concept from [3])

  $s_0 \models (a_1)\models (a_2) \ldots (a_n)\models 

  \text{Issue: Owner of an action may not even know that the action is applicable, let alone that it makes progress toward the goal}.$

- **Implicitly-Coordinated Plan** $a_1, a_2, \ldots, a_n \in A$, iff

  $s_0 \models K_{a_1}(a_1) \land K_{a_2}(a_2) \ldots K_{a_n}(a_n) \models 

  \text{The owner of the first action $a_1$ knows that $a_1$ is initially applicable and will lead to a situation where the owner of the second action $a_2$ knows that $a_2$ is applicable and will lead to a situation where... the owner of the $n$th action $a_n$ knows that $a_n$ is applicable and will lead to the goal being satisfied}.$

- Allows the agents to self-coordinate implicitly during the plan execution
- Forward search using product updates, shifting perspective in each step
- We generalized this concept also for conditional plans

Example: Russian Card Game [2]

- Problem: Seven cards randomly dealt to Alice, Bob & Eve
- Solution Concepts
  - **Centralized Plan** $a_1, a_2, \ldots, a_n \in A$, iff (concept from [3])
  - **Implicitly-Coordinated Plan** $a_1, a_2, \ldots, a_n \in A$, iff
  - Allows the agents to self-coordinate implicitly during the plan execution
  - Forward search using product updates, shifting perspective in each step
  - We generalized this concept also for conditional plans

Conclusion and Future Work

- Synthesis of epistemic plans, with coordination as part of the plan
- Practical Issues: Scheduling & livelock avoidance
- Applications, e.g., in Human-Robot Collaboration
