Notes Week 50

Paper: Solving Transition Independent Decentralized Markov Decision Processes

Abstract: Formal treatment of collaborative multi-agent systems has been lagging behind the rapid progress in sequential decision making by individual agents. Recent work in the area of decentralized Markov Decision Processes (MDPs) has contributed to closing this gap, but the computational complexity of these models remains a serious obstacle. To overcome this complexity barrier, we identify a specific class of decentralized MDPs in which the agents' transitions are independent. The class consists of independent collaborating agents that are tied together through a structured global reward function that depends on all of their histories of states and actions. We present a novel algorithm for solving this class of problems and examine its properties, both as an optimal algorithm and as an anytime algorithm. To the best of our knowledge, this is the first algorithm to optimally solve a non-trivial subclass of decentralized MDPs. It lays the foundation for further work in this area on both exact and approximate algorithms.

Summary:

Introduction

- The Multi-agent Markov Decision Process (**MMDP**) is a straightforward extension of the Markov Decision Process (**MDP**) to multiple agents by **factoring the action space** into actions for each of the agents.
- **Centralized** approach is easier to solve optimally than a general **decentralized** approach.
- Other researchers have focussed similarly to the authors on the **interaction between agents through the reward function**.
- Differently than the authors, these researchers use an observation model that resembles **full observability** of the MMDP rather than the **partial observability** of the authors model.
- Case of study: Situations where each agent has a **partial** and different

view of the global state.
- MMDP assumption: every agent has the same **complete** world state view.
- **Zero cost communication** can satisfy this assumption.
- A paper (Xuan and Lesser (2002)) has agents only communicate when there is **ambiguity**.
- ***The problem the authors examine, is one where communication has a very high cost or is not possible***.
- For the general **DEC-POMDP**, **the only known optimal algorithm** is a new dynamic programming algorithm developed by Hansen, Bernstein, and Zilberstein (2004).
- The authors approach computes the **optimal policy** goal as part of an **optimal joint policy**.

Problem description:

- The class of problems studied in this paper is characterized by two or more cooperative agents solving (mostly) independent local problems. **The actions taken by one agent can not affect any other agents' observation or local state**.
- ***An agent can not observe the other agents' states and actions and can not communicate with them***
- **The interaction between the agents happens through a global value function that is not simply a sum of the values obtained through each of the agents' local problems**
- The non-linear rewards combined with the decentralized view of the agents make the problem more difficult to solve than the MMDP, while the independence of the local problems make it easier to solve than the general DEC-MDP.
- This class of problem is called Transition Independent Dec-MDPs.
- Problems where the value of a single action performed by one agent may depend on the actions of other agents.
- Here the example is given where complementary and redundant actions are possible in a two agent setting. The global *utility* function (quantifies the preference of agents action in a given state (not a value function as it is not involved with a policy)) is no longer additive over the agents.

- Redundancy = global value is subadditive, as there is little additional value to completing both local problems.
- Complementary = global value is superadditive, as completing one of the local problems may have little value.
- The problem is motivated by a NASA rover problem where multiple rovers have to act (semi) autonomously, but have overlapping areas to explore or act within.
- Here, doing the same tasks in overlapping sites is redundant.

Formal Problem Description

- An n-agent **Dec-MDP** is defined as the tuple $\langle S, A, P, R, \Omega, O \rangle$, where:
    - S is a finite set of world states, with a distinguished initial state s^0.
    - $A = A_1, A_2, ..., A_n$ is a finite set of joint actions. A_i indicates the set of actions that can be taken by agent i.
    - $P : S \times A \times S \rightarrow \Re$ is the transition function. $P(s'|s, (a_1, ..., a_n))$ is the probability of the outcome state s' when the joint action $(a_1, ..., a_n)$ is taken in state s.
    - $R : S \times A \times S \rightarrow \Re$ is the reward function. $R(s, (a_1, ..., a_n), s')$ is the reward obtained from taking joint action $(a_1, ..., a_n)$ in state s and transitioning to state s'.
    - $\Omega = \Omega_1 \times \Omega_2 \times .... \times \Omega_n$ is a finite set of joint observations. Ω_i is the set of observations for agent i.
    - $O : S \times A \times S \times \Omega \rightarrow \Re$ is the observation function. $O(s, (a_1, ..., a_n), s', (o_1, ..., o_n))$ is the probability of agents 1 through n seeing observations o_1 through o_n, where agent i sees o_i, after the sequence $s, (a_1, ..., a_n), s'$ occurs.
    - There is joint full observability: the n-tuple of observations made by the agents together fully determine the current state.
- A **factored**, n-agent Dec-MDP is a Dec-MDP such that the world state can be factored into $n + 1$ components, $S = S_0 \times S_1 \times ... \times S_n$. The intention of this factorization is a separation of features of the world state that belong to one agent from those of the others (S_i) and from the external features (S_0). This separation is strict, where an observation

can not belong to two groups. The local state is the observations one agent can observe: $\hat{s}_i = S_i \times S_0$.

- A factored, n-agent Dec-MDP is said to be **transition independent** if there exist P_0 through P_n such that

$$P(s'_i|(s_0...s_n),(a_0...a_n),(s'_0...s'_{i-1},s'_{i+1}...s'_n)) = \begin{cases} P_0(s'_0|s_0) & \text{if } i = 0 \\ P_i(s'_i|\hat{s}_i, a_i, s'_0) & \text{for } 1 \leq i \leq n \end{cases}$$

  That is, the new local state of each agent depends only on its previous local state, the action of that agent, and the current external features. Here, the external features change based only on the previous external features. Implies:

$$P((s'_0...s'_n)|(s_0...s_n),(a_0...a_n))) = \prod_{i=0}^{n} P_i$$

- A factored, n-agent Dec-MDP is said to be **reward independent** if there exist R_1 and R_2 such that:

$$R((s_1...s_n),(a_1...a_n),(s'_1...s'_n)) = \sum_{i=0}^{n} R_i(s_1, a_1, s'_1)$$

  That is, the overall reward is composed of the sum of two local reward functions, each of which depends only on the local state and action of one of the agents.
- A factored, n-agent Dec-MDP is said to be **observation independent** if the observation an agent sees depends only on that agent's current and next local state and current action.
- A factored, n-agent Dec-MDP is said to be **locally fully observable** if each agent fully observes its own local state at each step.

Joint Reward Structure

- A **history**, $\Phi = [s_0, a_0, s_1, a_1, ...]$ is a sequence that records all of the local states and actions for one agent, starting with the local initial state for that agent.
- A **primitive event**, $e = (s_i, a_i, s'_i)$ is a triplet that includes a state, an

action, and an outcome state. An **event** $E = \{e_1, e_2, ..., e_n\}$ is a set of primitive events.

- A **primitive event** occurs in history, denoted $\Phi \models e$, if the triple e appears as a subsequence of Φ. An event occurs in history if every primitive event in the set occurs in history.
  Events are used to capture the fact that an agent accomplished some task.
- A **primitive event** is said to be **proper** if it can occur at most once in any possible history of a given MDP.
- An **event** is said to be **proper** if it consists of mutually exclusive proper primitive event with respect to some given MDP.

Given two histories for two agents Φ_1, Φ_2, a **joint reward structure** $\rho = [(E_1^1, E_1^2, c_1), ..., E_n^1, E_n^2, c_n]$ specifies the reward (or penalty) c_k that is added to the global value function.

Due to the **transition independence** of the MDP, we can define underlying MDPs for each agent, even though the problem is not reward independent.

Given a joint policy $(\pi_1 ... \pi_n)$ and a joint reward structure ρ, the **joint value** is:

$$JV(\rho | \pi_1 ... \pi_n) = \sum_{i=0}^{|\rho|} P(E_i^1 | \pi_1) \cdot ... \cdot P(E_i^n | \pi_n)$$

The **global value function** of a transition-independent Dec-MDP with respect to a joint policy is:

$$GV(\pi_1 ... \pi_n) = V_{\pi_1}(s_0^1) + ... + V_{\pi_n}(s_0^n) + JV(\rho | \pi_1 ... \pi_n),$$

where the standard value of the underlying MDP for the agents is summed up and added to the joint value.

The **optimal joint policy**, denoted $(\pi_1 ... \pi_n)^*$, is a set of policies that maximize the global value function:

$$(\pi_1 ... \pi_n)^* = argmax_{\pi'_1 ... \pi'_n} GV(\pi'_1 ... \pi'_n)$$

To summarize, a problem in our transition-independent decentralized MDP framework is defined by underlying MDPs per agent $\langle S_i, A_i, P_i, R_i \rangle$ and a joint reward structure ρ.

How does the CityLearn MDP satisfy these definition constraints?

- ☑ Definition **Dec-MDP**
- □ Definition **n-factored Dec-MDP**
    - This constraints are satisfied before cost-free communication among agents. After, the factored agent states have observations that overlap, violating the strict separation constraint.
- ☑ Definition **transition independent**
    - The new local state of each agent does only depend on the local state, the action the agent takes, and the current external features. The external features only depend on the previous external features. In the CityLearn environment, all shared observations are precalculated and independent of agent actions.
- ☑ Definition **observation independent**
    - The observation an agent sees depends only on that agent's current and next local state and current action. Observations that are not pre-measured (and independent of agent actions) are: energy storage state of charge (heating, cooling, dhw, electrical) and energy consumption (heating, cooling, dhw, electrical), which are all dependent on each buildings own actions.
- ☑ Definition **locally fully observable**
- □ Definition **reward independent**
    - Neighborhood level electricity consumption involves other agent local state features (sum of building net electricity consumption).

Paper: Scaling Up Decentralized MDPs Through Heuristic Search

Abstract: Decentralized partially observable Markov decision processes (Dec-POMDPs) are rich models for cooperative decision-making under uncertainty, but are often intractable to solve optimally (NEXP-complete). The transition and observation independent Dec-MDP is a general subclass that has been shown to have complexity in NP, but optimal algorithms for this subclass are still inefficient in practice. In this paper, we first provide an updated proof that an optimal policy does not depend on the histories of the agents, but only the local observations. We then present a new algorithm based on heuristic search that is able to expand search nodes by using constraint optimization. We show experimental results comparing our approach with the state-of-the-art DecMDP and Dec-POMDP solvers. These results show a reduction in computation time and an increase in scalability by multiple orders of magnitude in a number of benchmarks.

Summary

The authors propose an algorithm that casts and Dec-MDP with independent transitions and observations as a continuous deterministic MDP where states are probability distributions over states in the original Dec-MDP, which they call occupancy distributions. Then, continuous MDP techniques can be used to solve the MDP. The algorithm performes state occupancy exploration similarly to learning real-time A* while the policy selection is in accordance with decentralized POMDP techniques.

A **history-dependent decision rule** σ_i^τ at time τ maps from τ-step local action-observation histories h_τ^i to local actions: $\sigma_i^\tau(z_\tau^i) = a_\tau^i$ for all τ-steps.

A **markov decision rule** maps from local observations to local actions.

The τ-th **state occupancy** of a system under the control of a decentralized Markov policy $\langle\sigma_0 ... \sigma_{\tau-1}\rangle$ and starting at η_0 is given by:

$$\eta_\tau(s) = P(s|\sigma_{0:\tau-1}, \eta_0), \text{for all } \tau \geq 1$$

which are the probability distributions of ending up at a state given the history of

Markov decision rules up to that point given a certain starting state, for each step of the τ-steps

The current state occupancy depends on the past decentralized Markov policy only though previous state occupancy and the previous Markov decision rule. In other words, the state occupancy summarized all possible joint action-observation histories, decentralized Markov policy produced at horizon τ for the estimate of the joint decision rule.

The state occupancy is a **sufficient statistic** for decentralized Markov decision rules.