ABSTRACT

In order to be useful in the real world, an AI agent needs to plan and act in the presence of other agents, who may be helpful or disruptive. In this paper, we consider the problem where an autonomous agent needs to act in a manner that clarifies its objectives to cooperative agents while simultaneously preventing adversarial agents from inferring those objectives. We call it Mixed-Observer Controlled Observability Planning Problem (mo-copp). We develop two new solution approaches: one provides an optimal solution to the problem given a fixed time horizon by using an integer programming solver, the other provides a satisficing solution using heuristic-guided forward search to achieve prespecified amount of obfuscation and legibility for adversarial and cooperative agents respectively.

KEYWORDS

obfuscation, legibility, integer program, planning

ACM Reference Format:


1 MO-COPP

In a multi-agent environment, the activities performed by an agent may be observed by other agents. In such an environment, an agent should perform its tasks while taking into account the observers' sensing capabilities and its relationship with the observers. Several prior works have explored the generation of legible behavior to convey necessary information to a cooperative observer [1, 5, 8] and obfuscating behavior to hide sensitive information from an adversarial observer [3, 4, 6, 7]. However, in real-world settings of strategic importance, an agent might encounter both types of observers simultaneously. This would necessitate synthesizing a behavior that is simultaneously legible to friendly entities and obfuscatory to adversarial ones. For instance, in soccer, a player may perform a feinting trick to confuse an opponent while signaling a teammate. This problem gives rise to a novel optimization space that involves trading-off the amount of obfuscation desired for adversaries with the amount of legibility desired for friends.

A MO-COPP setting involves an actor (A) and two observers, where one is adversarial observer (X) while the other is cooperative (C). The actor has full observability of its own activities and knows the sensor models used by the observers. The observers have different sensor models. When the actor takes an action and reaches a new state, an observation is emitted. After obtaining the observations, the observers update their belief. The actor leverages the known limits in the observers’ sensors to control the observability of multiple observers in the environment simultaneously. Given a set of candidate goals, the objective of the actor is to convey information about its goal to C and to hide it from X.

Formally, a mixed-observer controlled observability planning problem is a tuple, mo-copp = (Λ, P, G, {Ωi}i∈Λ, {Oi}i∈Λ, {B_0^i}i∈{XC}). Λ = {A, C, X} is the set of agents. P = (T, Op, I, G_A) is A’s task captured as a planning problem [2], where T is the set of fluents, Op is the set of actions, I is the initial state and goal G_A is a subset of fluents. Also, for a ∈ Op, pre(a), add(a), delete(a) are each a subset of fluents representing preconditions, add effects and delete effects of a. G = {G_1, G_2, …, G_{n−1}, G_A} is the set of candidate goals, where G_A is the true goal of A, which is not known to either C or X. Ω_i is the set of observation symbols for agent i, which are emitted when A takes an action and reaches a new state. Further, Ω_A = {o_{A}^i}a ∈ Op, s ∈ S}. O_i : Op × S → Ω_i is agent i’s deterministic sensor model. S is the set of states, where each state is an instantiation of all fluents. Further, O_A maps each action-state pair to a unique observation, giving A full observability. While, O_X and O_C are noisy sensor models that map multiple action-state pairs to the same observation symbol, giving observers partial observability. B_0^i is the initial belief of an observer, i ∈ {X, C}. The initial belief is a set of states inclusive of I.

Although the observers are aware of the planning domain of the actor and of the candidate goals, they do not know which candidate goal is the actor’s true goal, G_A. The observers’ partial observability is due to mapping of multiple ⟨a, s⟩ pairs to an observation, i.e., ∀i ∈ {X, C}, ∃a, a’ ∈ Op, s, s’ ∈ S, a ≠ a’ ∧ s ≠ s’ : O_i(a, s) = O_i(a’, s’). Each observer i ∈ {X, C} maintains its belief, which is a set of states. Γ(·) is a transition function, such that, Γ(s, a) = s if s ∈ pre(a); else Γ(s, a) = s ∪ add(a) \ delete(a). Now we can define the belief update: (1) at time step t = 0, the initial belief of i is given by B_0^i, (2) at time step t ∈ {1, …, T}, B^i_t = {s | ∀a, s_i = Γ(s, a) = s; s ∈ B^i_{t−1}; O_i(a, s) = o_{A}^i}, where T is the time horizon. That is, the belief is updated using the previous belief and the observation received.

A plan π (or sequence of actions) is associated with a sequence of observations, ObsSeq(π) = {o_{A}^1, …, o_{A}^n} for each observer i.

A plan π is a valid solution to a mo-copp = (Λ, P, G, {Ωi}i∈Λ, {Oi}i∈Λ, {B_0^i}i∈{XC}), iff: Γ(I, π) ⊨ G_A. In other words, any
solution to \( P \) is a solution to mo-copp. We measure the quality of a valid mo-copp solution in terms of the extent to which the plan is legible to C and obfuscating to X. Given a solution plan, \( \pi \), \( G_X^\pi = \{ G \in G : \exists s \in B_{TP}^G, s \models G \} \) denotes the goals present in X’s final belief and \( G_C^\pi = \{ G \in G : \exists s \in B_{TP}^C, s \models G \} \) denotes the goals present in C’s final belief. \( |G_X^\pi| \) and \( |G_C^\pi| \) represent the amount of goal obfuscation for X and the amount of goal legibility for C. By increasing (or decreasing) the number of goals in \( G_X^\pi \) (or \( G_C^\pi \)), we can improve the goal obfuscation (or goal legibility).

Given a solution plan \( \pi \) that satisfies mo-copp, where \( |G| = n \), the goal difference, of \( \pi \) is given by: \( GD(\pi) = \frac{|G_X^\pi| - |G_C^\pi|}{|G|} \) where the denominator represents the difference between ideal values of \( G_X^\pi \) and \( G_C^\pi \). An optimal solution to mo-copp maximizes the trade-off between amount of goal obfuscation and goal legibility. That is, it maximizes the difference between the number of goals in \( G_X^\pi \) and \( G_C^\pi \). Equivalently, closer the \( GD(\pi) \) value to 1, better is the plan quality. A solution plan with \( GD(\pi) = 1 \) is an optimal plan.

2 PLAN SYNTHESIS

We now present two solution synthesis approaches: (1) we formulate mo-copp as a constraint optimization problem and provide an IP encoding to solve it in \( T \) steps, (2) we use heuristic-guided search to achieve preset levels of goal obfuscation and legibility simultaneously.

**mo-copp as Integer Program** The IP encoding provides an optimal solution for the given horizon by maximizing the trade-off between the amount of obfuscation and legibility. Let \( x_{a,t}, y_{a,t}, w_{a,t} \) be indicator variables for action \( a \), state \( s \) and observation \( o \) at time \( t \) respectively, \( b^t_{a,s}, h^t_{a,s,t} \) for state \( s \) and action \( a \) being applicable in state \( s \) in observer \( i \)’s belief at time \( t \) respectively, \( g^i_{G,T} \) for a goal \( G \) present in observer \( i \)’s final belief. The objective function is essentially the granularity of \( GD(\cdot) \) metric, i.e.,

\[
\max \sum_{G \in G} g^i_{G,T} - \sum_{G \in G} g^i_{G,T}
\]

The IP constraints are as follows:

1. \( \forall s \in S, s = I : y_{s,0} = 1; s \notin I : y_{s,0} = 0; \sum_{s \in S} x_{s,t} = 1 \)
2. \( \forall i \in \{X, C\}, \exists s \in B_{TP}^i : b^t_{i,s,0} = 1; \exists s \notin B_{TP}^i : b^t_{i,s,0} = 0 \)
3. \( \forall i \in \{X, C\}, G \in G, m > |\{ s \in G \} | : m \times g^i_{G,T} - \sum_{G \in G} h^t_{G,T} \geq 0 \)
4. \( \forall a \in Op, t \in \{1, \ldots, T\}, \forall s, s' \in S, s \models \{ s \} G_{a,t} \models \exists_{s \models \{ s \} G_{a,t-1}} \text{add}_a \text{add}_a = \{ \text{add}_a( s \text{add}_a) \text{add}_a( s \text{add}_a) \text{add}_a( s \text{add}_a) \}
\)
5. \( \forall s, s', t \in \{1, \ldots, T\}, \exists_{s \models \{ s \} G_{a,t}} \text{add}_a = \{ \text{add}_a( s \text{add}_a) \text{add}_a( s \text{add}_a) \text{add}_a( s \text{add}_a) \}
\)
6. \( \forall a \in Op, t \in \{1, \ldots, T\}, \exists_{s \models \{ s \} G_{a,t}} \text{add}_a = \{ \text{add}_a( s \text{add}_a) \text{add}_a( s \text{add}_a) \text{add}_a( s \text{add}_a) \}
\)

**Empirical Evaluation** The average and standard deviation \( GD \) for 3 domains are reported in Figure 1. The IP has higher \( GD \) for all 3 domains. While, the search algorithm generates satisfying solutions with lower \( GD \) that meet the goal constraints. The baseline planner that achieves satisfying solution to a single goal produces worst quality solutions (lowest \( GD \)).

**Acknowledgements.** This work is supported in part by ONR grants N00014-16-1-2892, N00014-18-1-2442, N00014-18-1-2840, N00014-19-1-2119, AFSOR grant FA9550-18-1-0067, DARPA SAIL-ON grant W911NF-19-2-0006, NSF grants 1936997 (C-ACCEL), 1844325, NASA grant NNX17AD06G, and a JP Morgan AI Faculty Research grant.
REFERENCES