ECAI 2016
G.A. Kaminka et al. (Eds.)

doi:10.3233/978-1-61499-672-9-1714

Secure Multi-Agent Planning Algorithms

Michal Štolba[1] and **Jan Tožička**[1] and **Antonín Komenda**[1]

Abstract. Multi-agent planning (MAP) is often motivated by the preservation of private information. Such motivation is not only natural for multi-agent systems, but is one of the main reasons, why MAP problems cannot be solved centrally.

In this paper, we analyze privacy leakage of the most common MAP paradigms. Then, we propose a new class SECMAP of secure MAP algorithms and show how the existing techniques can be modified to fall in the proposed class.

1 Introduction

Cooperative multi-agent planning models the problems in which multiple agents need to find a plan fulfilling a common goal. The reason the agents cannot simply feed their problem descriptions into a centralized planner typically lies in that although the agents cooperate, they want to share only the information necessary for their cooperation, but not the information about their inner processes.

A number of planners solving Multi-Agent Planning (MAP) has been proposed in recent years, such as MAFS [4], FMAP [6], PSM [7] and GPPP [3]. Although all of the mentioned planners claim to be privacy-preserving, thorough formal treatment of such claims is rather scarce. The privacy of MAFS is discussed in [4] and expanded upon in [1], proposing Secure-MAFS, a version of MAFS with stronger privacy guarantees.

We propose a new class of MAP algorithms, SECMAP and show that it preserves more privacy than the existing algorithms and how the existing algorithms can be modified to be SECMAP.

1.1 Multi-Agent Planning

In this contribution we use the MA-STRIPS [2] formalism to describe MAP. Formally, for a set of agents $\mathcal{A}$, a MAP problem $\mathcal{M} = \{\Pi_i\}_{i=1}^{|\mathcal{A}|}$ is a set of agents' local STRIPS problems. An agent problem of agent $\alpha_i \in \mathcal{A}$ is defined as $\Pi_i = \langle \mathcal{F}_i, \mathcal{O}_i, s_I, s_\star \rangle$, where $\mathcal{F}_i \subseteq \mathcal{F}$ is a set of facts partitioned into the set $\mathcal{F}^{\mathsf{pub}}$ and $\mathcal{F}_i^{\mathsf{priv}}$ of public (common to all agents) and private (of agent α_i) facts. The state $s_I \subseteq \mathcal{F}$ is the initial state and $s_\star \subseteq \mathcal{F}$ represents the goal condition. The set $\mathcal{O}_i$ of actions comprises of three pairwise disjoint sets: a set $\mathcal{O}_i^{\mathsf{priv}}$ of private actions of α_i, a set $\mathcal{O}_i^{\mathsf{pub}}$ of public actions of α_i and a set $\mathcal{O}^{\mathsf{proj}}$ of public projections of other agents' actions. *Public projections, e.g.* $\pi^{\triangleright}$, of actions / (partial) plans / problem which are shared with other agents are restrictions to public facts and actions. *Local solution* is a solution of Π_i and *global solution* is a solution of the whole $\mathcal{M}$.

A public plan is α_i-extensible, if by adding $a_k \in \mathcal{O}^{\mathsf{priv}}$ to the plan we can obtain a local solution to Π_i. According to [7], a public plan α_i-extensible by all $\alpha_i \in \mathcal{A}$ is a global solution to $\mathcal{M}$.

[1] {stolba,tozicka,komenda}@agents.fel.cvut.cz, Department of Computer Science, Faculty of Electrical Engineering, Czech Technical University in Prague, Czech Republic

2 Analysis of MAP Algorithms

To analyze the worst-cases of different planning paradigms, we alter the multi-agent planning problem $\mathcal{M}$. Let $\mathcal{M}^*$ be the problem of finding all solutions of $\mathcal{M}$. This modified problem corresponds to the worst-case execution of state-space search algorithms (explore complete search space), partial-order planning algorithms (explore all possible partial plans) and coordination-space search algorithms (explore all possible combinations of local plans).

Let $T(\Pi_i)$ denote a structure containing all solutions of Π_i and $T^*(\mathcal{M})$ denote a structure containing all solutions of $\mathcal{M}^*$. Obviously, $T^*(\mathcal{M})$ represents the minimal knowledge that is revealed by the solution of $\mathcal{M}^*$ and thus also by the worst case scenario in $\mathcal{M}$. It is not tractable to achieve $T^*(\mathcal{M})$ leakage as it is at least as difficult to solve the $\mathcal{M}^*$ problem.

2.1 Privacy Leakage of MAP Algorithms

There are two dominating MAP paradigms: FS is a forward-chaining (or analogously backward-chaining) state-space search. In the multi-agent version, each state expanded by a public action is sent to all relevant agents. Examples of such planners are MAFS [4], SECURE-MAFS [1], or forward-chaining Partial Order Planning (POP). In POP (e.g. FMAP [6]), the public projections of plans are shared in order to coordinate the exploration. CS is a coordination-space search, a paradigm specific for multi-agent planning, where agents attempt to agree on a coordination scheme (public projections of local solutions) which is then extended by private actions of all agents. Examples of such are the PSM [7] and GPPP [3] planners.

We analyze the leakage of private information of the described planning paradigms and particular planners in the worst-case scenario, that is when solving $\mathcal{M}^*$. We will focus on three types of private knowledge leakages. *Superfluous plans* are (partial) plans revealed by the algorithm without being the actual solution (or its prefix). *Superfluous distinct states* are publicly equivalent states s, s' revealed that $s \neq s'$ and either s or s' is not part of the solution. The most common situation where the superfluous distinct state information leaks is the use of unique state labels. And finally *superfluous action applicability* is an information of applicability of an action a on two distinct publicly equivalent states s, s' s.t. $a^{\triangleright}$ is applicable in both $s^{\triangleright}, s'^{\triangleright}$ known to the adversary that s, s' are distinct states and either s or s' is not part of the solution, in other words, the states s, s' are superfluous distinct states.

Forward/Backward State-Space Search The most significant source of leakage in state-space search algorithms is the use of unique IDs representing the private parts of the states, thus distinguishing publicly equivalent states even when it is not necessary.

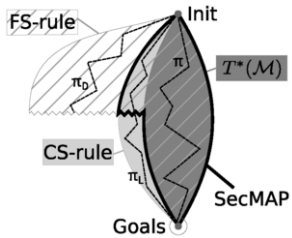

Figure 1. Portions of the state space leaked by application of the CS-RULE, FS-RULE and their combination, where π_D is a sequence of actions leading to a dead-end, π_L is a local plan for Π which cannot be extended to form a global plan for $\mathcal{M}$ and π is a global plan for $\mathcal{M}$.

A multi-agent forward search algorithm jointly explores the state-spaces of all agents, therefore only globally reachable states are explored. A source of superfluous distinct states is that dead-end states are also explored, communicated with other actions and subsequently action applicability is revealed. There are no superfluous plans in FS.

The SECURE-MAFS [1] algorithm reduces privacy leakage by not communicating a state with equal public and other agents' private parts twice. While this approach reduces the number of revealed states and actions, it does not prevent the exploration of dead-end states.

Coordination-Space Search Only states and actions which appear in some local plan are explored in the coordination-space search, that is states which are locally reachable and are not local dead-ends. On the contrary, parts of the state-space which are not globally reachable may be explored as well. In CS, only the necessary publicly equivalent states need to be distinguished, which also results in less superfluous action applicability revealed.

2.2 Designing a Secure Multi-Agent Planner

Based on the above analysis, we can attempt to improve the existing algorithms to reduce leaked private information when solving $\mathcal{M}^*$. Let us state three rules preventing privacy leakage based on the techniques used in the existing algorithms:

CS-RULE: Before communicating a state s, make sure it is part of a local solution to the agent's problem Π_i.

FS-RULE: Before communicating a state s, make sure it is reachable in $\mathcal{M}$.

$\triangleright$-RULE: Do not communicate a state with equivalent public and other agent's private parts more than once.

Figure 1 illustrates portions of the state space leaked by application of the CS-RULE, FS-RULE and their intersection. The $\triangleright$-RULE is not shown in the figure as it does not directly influence the search space, but rather make the leaked information less dense. Obviously, the best algorithm would expand and communicate only states of $T^*(\mathcal{M})$, thus resulting in zero leakage, but that would require to check whether a state is part of a global solution (i.e. solving $\mathcal{M}^*$).

We propose a class of algorithms called SECMAP, containing algorithms which follow all three proposed rules when communicating about states, actions and plans.

2.3 SECMAP Algorithms

The rules defining SECMAP are constructive and thus they can help us modify each of algorithms to fall in the SECMAP class.

MAFS and SECURE-MAFS already satisfy the FS-RULE as all reached states during the search are globally reachable. To satisfy the CS-RULE, the agents need to verify that the extracted state s is part of some local solution, before sending it to other agents. Since MAFS assures that s is (globally) reachable, it is enough to check that also the goal is reachable from this state using $\mathcal{O}_i$ actions. Such check requires to solve new local planning task and if it is unsolvable the state s can to be ignored. The $\triangleright$-RULE can be satisfied by the same way as in SECURE-MAFS, that is never sending a state s which differs only in the private part of the sending agent.

PSM builds local plans in parallel by all agents. These plans are exchanged among all agents, therefore all agents have (in the end) public projections of all agents' local solutions. A non-empty intersection of these solutions represent global solutions (a public plan α_i-extensible by all $\alpha_i \in \mathcal{A}$ is a global solution [7]).

PSM naturally fulfills the CS-RULE. To satisfy the FS-RULE, the secure variant SECMAP-PSM must not send the whole local solutions π at once, but each agent has to check prefixes of the generated plans whether they are all globally reachable. Again, the $\triangleright$-RULE can be satisfied by never sending a state s which differs only in the agent's own private part from some already sent state s'.

Theorem 1. *The* SECMAP-MAFS *and* SECMAP-PSM *algorithms do not leak more information than* SECMAP.

Proof. The proof can be found in [5]. □

3 Conclusions and Future Work

We have identified which cases of information leakage are presented in the most common multi-agent planning paradigms and a new class SECMAP of privacy preserving algorithms has been proposed. This class is guaranteed to leak less information than any currently known algorithm for a certain classes of problems. We proposed how to change the existing planners to belong to SECMAP for the price of increased computational complexity as all SECMAP algorithms require another (albeit local) planning process. It allows to decrease the need for communicating information which can be used to deduce private parts of the problem. Proposing a practically efficient SECMAP algorithm is left for future work.

Acknowledgments

This research was supported by the Czech Science Foundation (grant no. 15-20433Y) and by the Grant Agency of the CTU in Prague (grant no. SGS14/202/OHK3/3T/13).

References

[1] Ronen I. Brafman, 'A privacy preserving algorithm for multi-agent planning and search', in *Procs. of the IJCAI'15*, pp. 1530–1536, (2015).
[2] Ronen I. Brafman and Carmel Domshlak, 'From one to many: Planning for loosely coupled multi-agent systems', in *Procs. of the ICAPS'08*, pp. 28–35, (2008).
[3] Shlomi Maliah, Guy Shani, and Roni Stern, 'Collaborative privacy preserving multi-agent planning', *Procs. of the AAMAS'16*, 1–38, (2016).
[4] Raz Nissim and Ronen I. Brafman, 'Distributed heuristic forward search for multi-agent planning', *JAIR*, **51**, 293–332, (2014).
[5] Michal Štolba, Jan Tožička, and Antonín Komenda, 'Secure multi-agent planning', in *Proc. of the Intl. Workshop on PrAISe*, (2016).
[6] Alejandro Torreño, Eva Onaindia, and Oscar Sapena, 'FMAP: distributed cooperative multi-agent planning', *AI*, **41**(2), 606–626, (2014).
[7] Jan Tožička, Jan Jakubův, Antonín Komenda, and Michal Pěchouček, 'Privacy-concerned multiagent planning', *KAIS*, 1–38, (2015).